Simulating a Mixed Reality Memory Palace

Matthew Rossman
College of Information and Computer Sciences
mrossman@umass.edu

May 2020

Abstract—Memory palaces are a valuable mnemonic strategy for effectively memorizing many types of information. Spatial computing is in a unique position to reduce the barrier of entry for this method by providing users with immersive virtual memory palaces, turning the mental task of visualization into an experiential one. Prior works have developed virtual and augmented reality demos of virtual memory palaces, but few efforts have fully explored the potential for mixed reality implementations in which mnemonics have presence in the real world. The contributions of this work include the novel use of virtual reality hardware to prototype a mixed reality memory palace tool, as well as an outline for a psychological experiment that would guide the design of immersive learning tools. The prototype application shows promise for mixed reality as a platform for memory palaces, and highlights areas for improvement in future implementations.

Honors Thesis Committee

Narges Mahyar
College of Information and Computer Science

David Huber
Department of Psychological and Brain Sciences

Contents

Introduction

The memory palace technique, more formally known as the method of loci, is a proven method to retain large amounts of information. The technique involves associating each piece of information with a location along a route in a familiar space. To retrieve the information, one can simply mentally retrace their steps through the space and visualize the item assigned to each location. Spatial computing allows us to structure data and visualizations in physical space, so it's easy to see how this technology would benefit a spatial learning strategy. Packaging the method of loci as an unintimidating spatial application would enable average users to much more effectively learn all sorts of information, ranging from remembering telephone numbers to conceptualizing foreign languages. Prior works have largely focused on 2D or strictly virtual reality (VR) implementations of the virtual memory palace, while mixed reality implementations, which leverage our prior familiarity with the spaces around us, have not been studied as thoroughly.

In this thesis, I propose an experiment that addresses the potential benefits of a mixed reality memory palace through assessments of recall performance when using personally familiar environments in a virtual memory palaces. Additionally, I present a prototype memory palace application developed using a mixed reality framework. This application lets users search for 3D visualizations of objects from an online repository to spawn into their environment, which they can then position and label using natural hand gestures. To demonstrate the application, I take a novel approach of simulating mixed reality by displaying 3D reconstructions of real spaces on a standalone virtual reality headset. Based on my own experience with the prototype, I believe VR can create a convincing simulation of mixed reality (MR) while also providing some additional benefits with regards to immersion and accessibility. Building memory palaces with your hands is fun and easy, and I think there is promise in this kind of application. However, limitations of navigating the real world mean that there is still some work required to design accessible memory tools for MR.

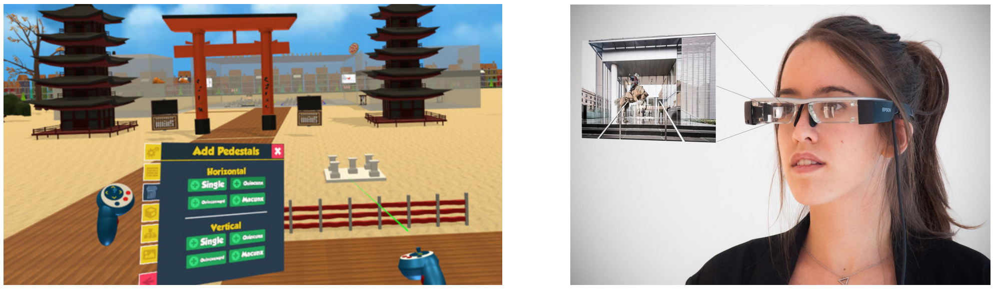
The topic of this paper is significant because memory palaces can help individuals retain more knowledge. The more people know, the more they can do with their knowledge. Throughout history, humans have devised numerous systems to organize information and support memory. The earliest cave paintings 40,000 years ago, the printing press from a few hundred years ago, and the computer developed within the past 50 years serve to offload memory and share it with others. These developments bring with them revolutions in thought. 2020 marks a turning point where the vision of spatial computing is rapidly entering our reach. Big companies like Apple, Facebook, Google, and Microsoft are investing heavily in the space. This new wave of technology will bring with it yet another way of organizing information—we just have to design effective tools to facilitate it.

Within the domain of psychology, the experiment proposed in this paper would clarify some widespread suspicions that memory palaces should be constructed from places that you have some personal attachment to, such as your home or route to work. This could shed more light on the way that humans encode and retrieve information in the brain. In the domain of human-computer interaction (HCI), the experiment would give useful insights into which XR platform is best suited for memory palace applications, since augmented reality (AR) naturally builds upon the real world while VR tends to substitute it for something else. Additionally, the prototype MR application demonstrates a potential interface for constructing memory palaces in MR and highlights areas for improvement.

Background

The Method of Loci (MoL) dates back to ancient Greek times, attributed to Simonides' recounting of the deceased attendees of a banquet by picturing how they were seated at the table. It more recently gained popularity following the 1966 publication of Yates' *The Art of Memory* [1], which formally defined the technique. Most superior memorizers report using the method [2], enabling them to accomplish feats such as Alex Mullen's 2017 record of memorizing more than 3000 random digits within an hour [3]. The method is most easily applied to information that has a straightforward visual representation (e.g. grocery products), but with the help of carefully crafted mnemonics it can be applied to nearly any content. For example, if you want to memorize a math equation, you can assign mnemonics to each symbol and position them within an environment as they appear in the equation. Perhaps you place one mnemonic on a shelf above another to represent a fraction, or under a roof to represent a square root. Another example is long sequences of numbers, as Mullen performed for his record. For this you can use something like the Major system to encode chunks of numbers as phonetic sounds, which you can translate into visual mnemonics. Memory gurus have devised tricks like these to encode just about anything into memory palace entities.

The method's success is generally attributed to humans' evolved capacity to remember the layout of spaces, a characteristic which helped our ancestors survive in the wild. Recent research has studied the method more critically to determine its factors for success. One of the most peculiar findings came from Legge et al. in 2012 [4]. In this study, participants used an unfamiliar virtual environment as the basis for a memory palace, which they were allowed to study for 5 minutes. Other participants used the conventional method, basing their palace off a highly familiar location. Legge found comparable performance between the two groups, both of which outperformed a control group which was not instructed to use the MoL. This suggested that familiarity of the environment is not critical to the success of the imagined MoL technique. It should be noted that subjects using the virtual environments were permitted to explore the environment prior to training, while



(a)

(b)

Figure 1: (a) Screenshot from MacunxVR, now known as MunxVR (b) Simulated imagery of the NeverMind system [10]

users of the conventional method had to rely exclusively on their imagination.

Legge returned to assist Jeremy Caplan et al. in 2019 on another test of the MoL [5]. This time, they tested the effects of switching up the structure of the virtual environment. A variety of environments were tested, ranging from something easily navigable like an apartment setting to something more challenging to navigate like a radial arm maze. They did not find a significant effect in changing the structure of the environment on effectiveness of the technique, suggesting that the success of the method is not dependent on imagined navigation.

These studies have cast doubt on some of the core principles of the method, suggesting that it is no more effective than countless other peg methods. However; they test the methods only in its imagined form, so the relevance of these inferences to an immersive application of the MoL is not clear. The importance of this distinction is highlighted by Huttner et al.'s findings that subjects recall information better when presented with visual representations of loci during information encoding compared to mentally visualizing the loci from a word list [6] as was done in Legge's studies. Perhaps the significance of familiar memory palace environments only arises when you see those spaces with your own eyes.

In 2006, Fassbender and Heiden proposed a virtual memory palace (VMP) as means to support and preserve human memory [7]. Their prototype used a low fidelity 3D engine to display 2D menonics as paintings around a virtual palace environment, with preliminary results suggesting that the virtual tool improved information recall among users compared to a simple world list. Furthermore, Krokos et al. showed that VMPs presented through an immersive head mounted display (HMD) resulted in a 8.8% improvement in recall compared to a 2D desktop condition [8], giving merit to commercial VR memory palace applications like Ralby et al.'s Macunx VR [9].

Rosello took a slightly different approach in 2017 with NeverMind, an augmented reality memory palace application in which users could see mnemonics overlayed on their real-world surroundings through a pair of smart glasses coupled with an iPhone app [10]. However, the mnemonic overlays were simply 2D images, and they weren't spatially tracked to the user's environment, creating a barrier to immersion. Rosello suggested future work look into a true mixed reality implementation. This suggestion is supported by Reggente et al.'s recent findings that binding mnemonics to specific locations in an environment improved subject's recall by 28% compared to fixing the mnemonics in the center of their vision [11].

Rosellos's call for future work seemed to be answered when Yamada et al. prototyped a MR memory tool called HoloMol, however their system relied on subjects placing physical markers at each locus in the environment, and the overlay showed only text-based cues. Their prototype targeted the Microsoft HoloLens, which has a very small field of view (FOV) as I'll reiterate on later in the paper. Lin et al. found that higher FOV seems to improve spatial memory of virtual environments [12], which means that the display limitations of HoloLens pose an issue to its usefulness as a memory palace platform.

Thus, there is a lack of effective MR memory palace applications that live up to the vision Rosello proposed. There is also some uncertainty generated by noteworthy research about whether the familiar backdrops of MR provide any benefit for immersive memory palaces over existing VR tools that do away with reality altogether. This project seeks to fill some of those gaps in the literature.

Note on terminology

There is a great deal of confusion within the community about proper use of terms like AR, MR, XR, etc. It's my understanding that XR is the most appropriate umbrella term for spatial computing platforms. VR represents platforms that fully replace reality with a virtual world, such as the Oculus Rift of Valve Index. It is my personal belief that AR refers to any device that overlays virtual content on real imagery, such as the HoloLens or Magic Leap. However, for the purpose of this paper I will use terminology consistent with what Rosello has used in his NeverMind thesis, since that is the main inspiration for this work. According to Rosello's paper, AR includes simple overlays of virtual content, as achieved by lower-end smart glasses, while MR specifically refers to systems that can merge virtual content with real imagery through techniques like spatial tracking and occlusion. Thus, I will rarely use the term AR in this paper since it is so narrowly defined by Rosello.

Methodology

3.1 Experiment intro

At the outset of this project, I wanted to conduct an experiment to answer the question: is there a benefit to using personally familiar environments as the basis for a virtual memory palace? With tools like MunxVR and NeverMind taking very different approaches to the design of virtual memory palaces, I was curious if one approach was superior over the other in helping users recall information. My suspicion was that feeling immersed in a personally familiar space would lead to stronger memories of the experience in the VMP, thus yielding higher recall rates. To test this, I'd need to give users a strong sense of presence in a space while still maintaining experimental control over the environment. This spawned the idea of simulating mixed reality, which I'll soon describe in more detail.

3.2 VR as a form of true mixed reality

Traditionally, the effect of mixing virtual content in with the real world is done through camera compositing, either by overlaying the virtual imagery on top of a mobile phone's camera feed, or by projecting it on the lenses of smart glasses. However, there are some major issues with the conventional approach. First, these methods require complex computer vision algorithms to not only track the motion of the camera with 6 degrees of freedom (DOF), but also analyze the depth of the scene in order to properly occlude the virtual entities. Most consumer devices don't yet have the efficiency or processing power to perform reliable depth occlusion, leading to the appearance that the virtual content is pasted on top of real imagery rather than blending into it. Secondly, traditional approaches leave the user with a very narrow viewport. Display technology for smart glasses can currently only cover a small area of the lens[1], as if looking at a distant television. Similarly, mobile AR limits your view to the phone's screen area.

A possible solution to this is MR on a VR headset. Michael Abrash, Chief Scientist at Oculus, claimed in 2018 that "Mixed reality in VR is inherently more powerful than AR because there's full control of every pixel rather than additive blending...The truth is that VR is not only where mixed reality will first be genuinely useful, it will also be the best-mixed reality for a long time" [13]. In addition to pixel-wise control, it's much easier to get a wide FOV on a VR display than on a see-through display. As a result, I support the argument that VR headsets are the most accessible and immersive mixed reality platform for the near future.

There are only a few experimental devices that take this "MR in VR" approach, such as the LYNX-R1 or the Varjo XR-1. Some rumors are circulating online that Oculus is moving this direction with their Quest headset, due to a series of recent software updates that enhance its pass-through camera capabilities. Despite the strengths of this approach, there is yet to emerge a strong platform for mixed reality experiences on VR hardware. However, following the proposal of Ragan et al., the effect of MR can be reasonably simulated in software [14].

[1] The $3000 HoloLens from Microsoft sports a 35°diagonal FOV, while its $3,500 successor HoloLens 2 upgrades to 52°diagonal FOV. Both pale in comparison to the 110°FOV of consumer VR headsets

3.3 VR as a simulation of mixed reality

To simulate mixed reality, you simply take what was real and make it virtual. That means digitally reconstructing the real-world backdrop on which the virtual entities are normally anchored, as well as other real entities in the scene. Using an assortment of 3D reconstruction techniques, environments can be represented as 3D models and imported into software to view them on a VR headset. The specifics of this process will be covered in the next section. Moving objects (e.g. cars and people) are not easy to digitize, but vision-based hand tracking offered on the Oculus Quest allows you to at least see a virtual version of your hands through the headset. Another factor is navigation: traditional VR hardware is tethered to a PC to handle the graphics processing. Over the past year, standalone wireless headsets have been released with inside-out spatial tracking, affording unrestricted navigation and natural locomotion. Although the system software tries to contain you to a defined playspace, you can turn this feature off to move around spaces larger than the recommended 25 $\times$ 25 ft area. The result is a convincing effect of presence in a place you aren't actually in. Unike most modern AR experiences, we can easily integrate new virtual elements into the scene with pixel-perfect object occlusion, steady spatial tracking, and high FOV.

3.4 3D reconstruction methods

3.4.1 Refresher on 3D rendering

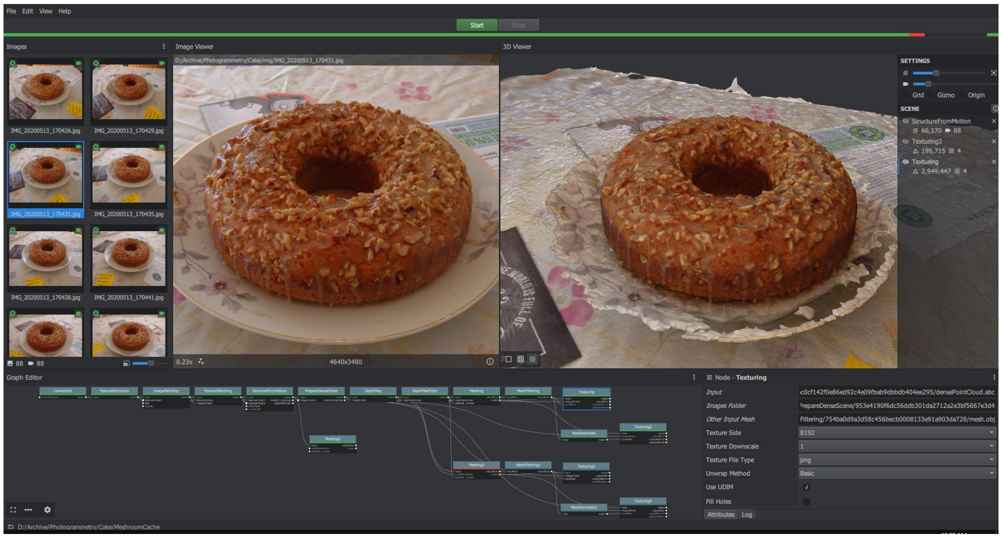
Before I get into the details of 3D reconstruction, I need to provide some background of how computers represent 3D models. Common 3D file formats include .obj, .ply, .dae, .fbx, .gltf, and many more. Each of these formats is at least capable of storing an object mesh, which consists of the positions of vertices and how they connect to form edges and faces. A mesh does not need to be connected, so two objects that appear completely disjoint could be part of the same mesh. In addition to a mesh, most formats provide some way to store materials. A material describes things like the roughness, color, and transparency of a surface. The color can be a fixed color like red or blue, or it can map to a multi-colored image texture saved in a regular .jpg or .png file. The resolution of a texture typically doesn't go beyond 4096x4096 pixels in order to maintain compatibility with popular game engines. In order to connect a 2D image to a 3D surface, models use a UV map, which describes where each vertex will lie in the two image coordinates (U and V). By carefully laying out faces in the UV map you can avoid wasting areas of the texture file. The last concept to keep in mind is "draw calls". A draw call is an instruction sent from the CPU to the GPU describing something to render. At a minimum, each mesh counts as a draw call, as does each material. Draw calls tend to be the cause of performance issues in 3D engines, as the CPU becomes a bottleneck. So to keep a model running smoothly it should use few meshes and few materials. Due the resolution limit of individual textures, it can take some effort to maintain detail with few materials. And simply combining all objects into a single mesh sounds like an appealing workaround to having too many mesh drawcalls, but this requires the entire mesh be rendered on each frame, even if most of it isn't visible. With those constraints in mind, I can get into the process of building a performant 3D scene.

3.4.2 Manual modeling

The field of 3D reconstruction is still quite young, and good-looking results are very challenging to produce. The most straightforward approach is to use 3D modeling software to build the scene by hand from reference images. By capturing pictures at orthogonal views, you can reasonably chunk out the shape of each object. To add materials to the mesh, you can take fronto-parallel photos of each surface with even lighting, paint out the edge seams, and tile it across the assigned faces. If you follow this approach of using isolated texture images, you will quickly find yourself with a lot of different materials (e.g. a room may have a wood material, a carpet material, a wallpaper material, and many more). This will incur a lot of draw calls and potentially drag down performance. A solution is to combine the various textured surfaces into a single image called a texture atlas, which can be achieved through a process called baking. The benefit of baking is that you can reduce the number of materials in your scene, but since our image still has a fixed size, the surfaces may lose detail. Another complication of baking is that tiled textures no longer work. Normally, when UV coordinates go beyond the (0, 1) domain they wrap around the other side, which is useful for simple repeating patterns like wood paneling. Tiled textures are important for large surfaces, which would otherwise require far more than a 4K resolution to cover. On a texture atlas you can't achieve this effect, because wrapping around the other side might place you on coordinates used by a completely different texture. You can see how designing a 3D scene is a complex balance between detail, performance, and ease of use. It's incredibly time consuming and often not feasible to get truly accurate results with the manual method.

3.4.3 Photogrammetry

Through a technique called photogrammetry, 3D reconstruction software can take in a large dataset of regular images and spit out a textured model. I'll outline the process as implemented by Meshroom, a popular open source photogrammetry application [15]. The photogrammetry pipeline has many steps, but it can be broken down into two main phases: structure from motion (SfM) and multi-view stereo (MVS). The SfM phase starts by selecting feature points of high contrast in each image, often using the SIFT (Scale-invariant feature transform) algorithm. Next, it matches feature points that are shared between images as "tie points". Once it knows how tie points are related and arranged within each image, the SfM algorithm can geometrically calculate the original view positions based on camera specifications (namely sensor size and focal length), and triangulate tie point positions in 3D. After this phase, the software has generated a sparse point cloud of the scene. The next phase is multi-view stereo, which takes the knowledge of camera positions and sparse depth to estimate pixel-wise depth values through a method such as Semi-Global Matching (SGM). Then, it merges the depth information from each view into a mesh. The mesh is automatically UV unwrapped to maximize texture space, and textures are assigned by sampling pixel colors from cameras that have the the best view of each face. After this phase, the software has produce the final dense, textured reconstruction of the object. The dense reconstruction often has millions of vertices, so some kind of mesh simplification and re-texturing is often added as a final step.



Node - **Texturing**

Input	c0cf142f0e86ed92c4e09fbab9dbbdb404ee295/densePointCloud.abc
Images Folder	PrepareDenseScene/953e4190f6dc56ddb301da2712a2e3bf5667e3d4
Other Input Mesh	Filtering/754ba0d9a3d58c456becb0008133e91a903da728/mesh.obj
Texture Side	8192
Texture Downscale	1
Texture File Type	png
Unwrap Method	Basic
Use UDIM	☑
Fill Holes	□

Figure 2: A screenshot from the photogrammetry software Meshroom. Each node in the bottom panel represents a step in the photogrammetry pipeline. In this example, I took 88 photos of a cake to produce the model on the right, which consists of nearly 3 million triangles. It took a little under 3 hours to process on my Nvidia GTX 1070.

Photogrammetry is a powerful tool for objects that are too detailed to model and texture by hand. However, it has significant caveats. First, it only works well for certain kinds of objects. It needs to be able to easily identify feature points around the object, which means surfaces of a flat color (such as a white wall) may not be meshed properly. The process also assumes that everything the camera sees is an opaque, rough surface. That means reflective and transparent surfaces will suffer lots of distortion (see the missing geometry on the plate in Figure 2. If you need to capture a surface with reflections or transparency, your only option is to temporarily cover the surface with something opaque such as spray-on powder. In many cases, this workaround isn't a realistic option. Another big drawback of photogrammetry is that it takes a lot of time, effort, and resources at every step. The photos you take have a huge impact on your results, and it takes extensive practice to learn the proper camera settings, framing, lighting, and scene setup in order for later steps to work. Once you have a clean photoset, there are still a number of parameters that may need tweaking in the software, and the entire process can take many hours (sometimes days) to run, even on powerful GPUs.

3.4.4 Laser Scanning

Typically used for taking building measurements, LiDAR (Light Detection and Ranging) scanners provide extremely precise points clouds of an entire space from a static perspective. They are incredibly expensive, with products like the Trimble TX8 currently selling for $66,000. LiDAR scanners function by sending out a beam of laser light and measuring the angle at which it is reflected back, which can be used to triangulate the scene depth at that point. As the device slowly rotates, it builds a complete point cloud of the scene. The TX8 can capture points that are just a few millimeters apart, yielding a scan with hundreds of millions of vertices. For large scenes or scenes with lots of occluding objects, multiple scans may be necessary to fill in the gaps. LiDAR point clouds can be processed alongside 2D imagery in photogrammetry software to generate textured meshes, but their high vertex resolution makes mesh simplification a necessity.

One shortcoming of this approach is that while mesh accuracy is high, image quality for texturing can be low. Color data comes from a panorama captured alongside the point cloud, but this camera sensor is not the priority of the machine. This is why the high accuracy scans might need to be combine with photogrammetry data, which can be captured from a higher resolution camera at more diverse angles. The more obvious drawback of LiDAR scanners is their astronomical price. However, mobile devices like the 2020 iPad Pro have recently started incorporating simple LiDAR sensors in their camera module. While not as accurate as the professional hardware, these sensors make laser scanning vastly more affordable and portable. Within the next year, there's likely to be an influx of mobile phones and tablets sporting LiDAR and time-of-flight sensors[2], in order to keep up with the rising interest in AR applications.

3.4.5 Structured Light Scanners

Structured light scanning is another approach involving dedicated hardware. The principle of this method is to project a known visual pattern onto the subject and measure how the pattern is deformed. By gathering enough of these measurements from different perspectives, we can geometrically conclude what mesh was present to

[2] Time-of-flight sensors send out lasers as well, but rather than triangulating the depth they estimate it by measuring how long it takes for the light to return.

create those deformations. This same principle is what powers the original Microsoft Kinect. The benefit of these sensors is how quickly they can measure the depth of an entire view, and they can come in stationary or portable form factors. However, they are sensitive to scene lighting, making them ineffective in harsh outdoor lighting conditions. It also doesn't work well with objects that don't deform light predictably, like those with transparent or reflective surfaces. This method is mainly used for scanning small objects for 3D printing.

3.4.6 Neural Radiance Fields

The methods I've described so far all see the world within the contraints of traditional 3D graphics, where everything is a mesh with materials assigned to its faces. However, there is an alternate view that the world is represented as a volume of light, with each lit surface shooting off rays into that volume. From this perspective, a camera image represents some intersection of that volume by a plane, characterised by the plane's position and orientation. As shown by Mildenhall et al.'s NeRF paper [16], you can train a neural network on a set of images and teach it how that light volume behaves. You can then query the network with a desired camera position and orientation to generate new views of the scene. This allows you to capture the 3D structure of a scene, and unlike photogrammetry, it works for all types of materials including reflective and transparent surfaces. This strategy is still very early in development, and there is no way to interact with such a model in traditional 3D rendering software without first estimating a mesh. However, it has the potential to produce much more photorealistic recreations of a space than any of the other methods.

3.5 Formalizing the proposed experiment

3.5.1 Targetting an environment

In order to decide which of these various 3D reconstruction techniques I should employ in my MR simulation, it would be helpful to identify the environment I'd need to reconstruct. I knew that it should be a location on the UMass campus, so that I could ensure I'd find subjects who were familiar with it. I decided on the dining halls, since students already understand their structure as a set of discrete loci (e.g. there is a salad station, a pasta station, etc.). Specifically, I picked Worcester dining hall since it is the closest to my apartment. In the initial experiment plan, I also wanted to reconstruct Valentine dining hall at the neighboring campus of Amherst college, so that I could conduct a counterbalanced within-subjects study involving participants from each school. However, I later simplified the experiment design to only require reconstructing Worcester dining hall, thus I would test between-subjects.

3.5.2 3D reconstruction strategy

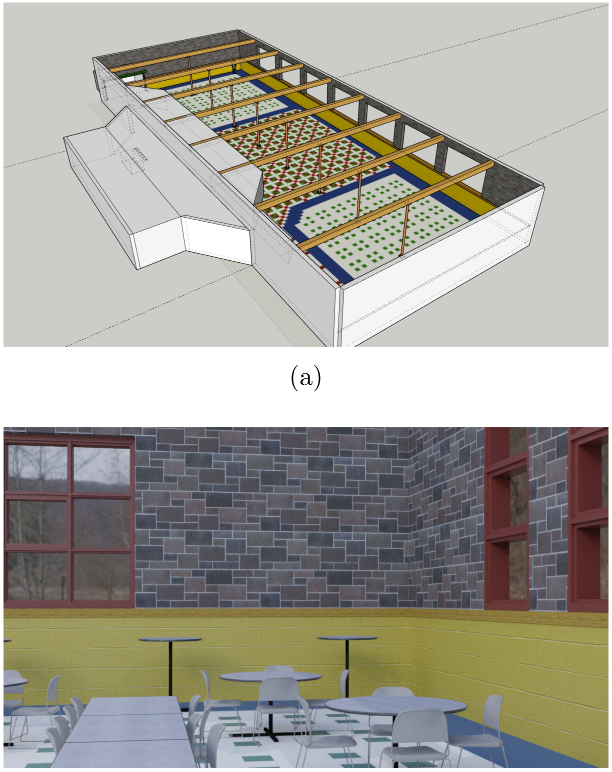
I spoke with staff members around campus to determine the best 3D reconstruction method to apply to my project. At the Media Lab, I was advised against photogrammetry due to the heavy computational requirements and limitations of surfaces that it can capture. I experimented with photogrammetry on my own, and found that reconstructing even small models was a tedious process that had to run overnight. I borrowed a handheld structured light scanner from the lab, but I found the model and texture quality were very low, and it struggled to maintain tracking while scanning objects. It also limited me to scan very small items, so scanning a whole room was out of the question. The university has a high end LiDAR scanner in the Building and Construction Technologies department, but the director there advised against using it for this task because it generates immense amounts of data. While LiDAR is very accurate, it would require many scans from various perspectives to cover the entire visible area, so my final result may end up with many missing patches. The general consensus was that although tedious, manual modeling would be the best approach for building a performant reconstruction for VR.

I waited until winter break when the building was unoccupied, and got permission from dining hall management to survey the space. I brought with me a DSLR camera, a 360 camera, tripod, and Sense 3D scanner. My plan was to scan small items like decorations with the 3D scanner, use the 360 camera to capture the room layout from multiple perspectives, and use the DSLR for chunking out the shape of large objects as well as fronto-parallel shots of surfaces for texturing. I captured 178 standard photos and 16 360-degree shots of the main dining space. My attempts to use the 3D scanner were fruitless, as it often lost tracking of the subject and produced low quality models. I attained floor plans of the building from maintenance, which gave the general measurements of the space. I repeated this process at Valentine dining hall, though these materials were not needed after the change in my experiment design.

The floor of Worcester is grid of 1'x1' square tiles, which made it very easy to map the precise position of objects in the room. I chunked out the room in SketchUp, a paid 3D design software that I got access to through the Building and Construction Technologies department. SketchUp made it easy to create to-scale models with precise measurements, however it had trouble interfacing with other 3D softwares. I ultimately made cleaner models in Blender, an open source 3D modeling software. Blender gave me more precise control over the textures of objects in my scene, and gave tools for baking textures.

3.5.3 Experiment procedure

In addition to identifying a target environment, I needed to layout exactly what subjects would do with this environment. Past work, such as that of NeverMind



(a)

(b)

Figure 3: (a) The basic layout of the space was blocked out in SketchUp, which excels at measurement-based modeling. (b) Engine-friendly assets were modeled in Blender, which gave more control over mesh topology and texture mapping

and the the experiments of Legge and Caplan tend to assess subjects by giving them small wordlists and testing their ability to recall those words some time later. I went this same route to keep my results consistent with related works. As controls, I'd first have subjects memorize a wordlist without any memorization strategy, and then another with the conventional MoL. Then, they would memorize a wordlist in the simulated MR condition. The wordlists would consist of simple objects like "dice" or "apple", which have an obvious visual representation. Words would be presented to subjects one at a time, and each word would only be shown for about 5 seconds. To keep things consistent, each word would be paired with a visual representation in all conditions, which may be a 2D image or 3D model. Each wordlist would consist of 11 words, which is the same amount used by Legge et al., motivated by a desire to discourage subjects from using natural chunking methods [4]. However, since designing the experiment I came across Francis Belleza's guidelines for MoL studies, which suggest that much longer wordlists should be used [17].

I aimed to recruit around 40 participants, 20 being UMass students who are familiar with the dining hall, and the other 20 being students either on or off campus who reported no familiarity with that space. To incentivize participation, I planned to award participants with a handful of Amazon gift cards. I designed advertisements for the study with a link to a recruitment survey, which asked about their familiarity with various dining locations on campus. I'd aim to select participants who reported the highest and lowest familiarity with Worcester dining hall.

3.5.4 Human subjects preparation

In order to run an experiment on students, I'd need to get IRB approval for experiments involving human subjects. To start, I completed group 2 CITI training which went over guidelines for ethical human subjects studies. Then, I had to complete a lengthy IRB application detailing the exact experiment procedure, recruitment materials, compensation schedule, advertisements, and more. Each of these steps also required me to get permission from some department on campus, such as getting my advertisements approved with dining hall management.

3.5.5 Space reservation

As I planned to incorporate physical locomotion in my MR simulation, subjects would need a large open space to walk around in while wearing the VR headset. Outdoor spaces weren't an option, both due to the cold temperatures and the fact that sunlight can damage the VR headset's cameras and displays. I contacted staff in nearly every large building on campus to find a space indoors that I could reserve. I prioritized gymnasiums since they were the largest options, but faced heavy competitions will all of the sports teams that practice in those spaces. Large event halls and auditoriums were also tightly booked. Eventually, I found a lesser known location on campus, the Dick Rossi Room. The room is about 40'x50', which is plenty of space to walk around without fear of hitting a wall.

Results

4.1 Hypothetical outcomes

Due to the unique circumstances involving global health concerns, the in-person experiment could not take place. I want to briefly discuss some of the hypothetical outcomes that may have arisen from the experiment and what they would mean.

The data I would attain from the experiment would be a spreadsheet with the words recalled by each participant from each condition and delay phase, which I could compare against the ground truth lists they had been assigned to. For each word a participant recalled, I would query the ground truth list with some margin of error (e.g. 2 characters to account for mispellings) to find the word they intended to write. With this typo-removed data I could compute the lenient score trivially by counting the overlap between the ground truth and recalled words. For strict scoring, I would run a sequence alignment algorithm to find the number of words correcly recalled in order. From this I could make some plots comparing the forgetting curves of each condition (no strategy, conventional MoL, VMP unfamiliar, VMP familiar) to identify visual trends. I could also make graphs comparing performance at each specific degree of familiarity as reported on the recruitment survey, as well as performance against reported memorization ability. More importantly, I'd run a series of formal statistical tests on the data.

The main tool I'd use is two independent sample t-tests, which allows me to test whether the difference between sample means from two independent populations is statistically significant. Specifically, I want to test if the average score (strict or lenient) at a given delay phase significantly differs between the population who had personal familiarity with the virtual environment compared to those who didn't. I can run this test for each delay phase, as it might be the case that a significant effect only arises in the long term. If the personally familiar population has a significantly higher performance at some phase, then the results suggest that personally familiar environments (and therefore MR platforms) may be better suited for immersive memory palaces. If the difference is not significantly different, then the results would expand on Legge's findings by showing that novel virtual environments work just as well for VMPs, even when subjects are immersed in the palace during encoding. I would not expect the familiar population to perform worse than the unfamiliar one, but if it did then it could suggest that the subjects were distracted by the experience of virtually visiting a place they know already.

I'd also want to run a dependent sample t-test on the simulated MR versus conventional MoL conditions as a sanity check to verify that the VMP yielded higher recall performance than the conventional word list strategies. I say dependent because a subject's performance in the VMP may be effected by "warming up" with the conventional MoL. All prior studies indicated that VMPs improve recall performance, so if my results don't reflect that then there could be a flaw in the design of the experiment or application.

4.2 Simpler reconstruction

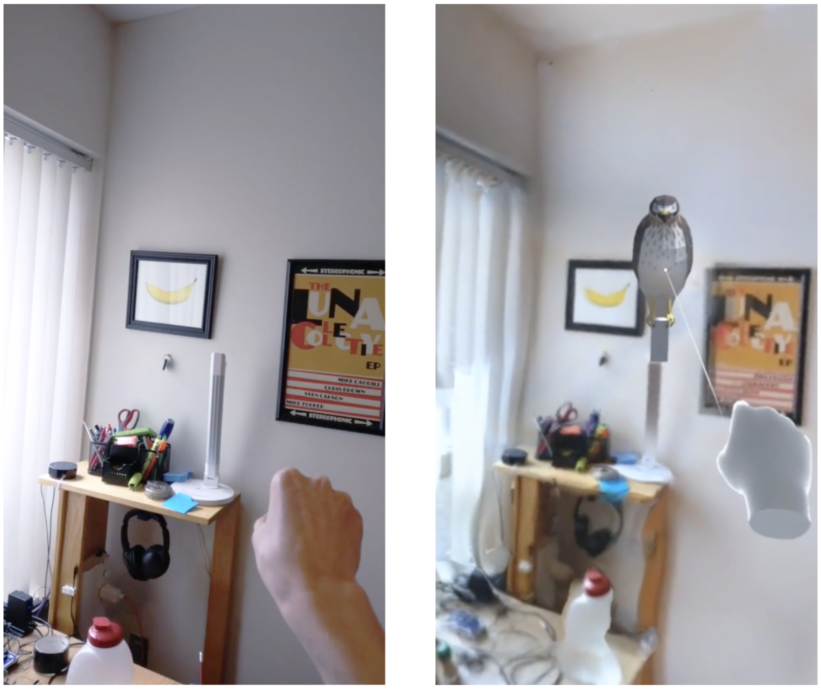
Most of my efforts so far had been with organizing the study, which was going to a controlled experiment that wouldn't require much UI for the subjects. Now, my focus shifted to simply prototyping a mixed reality memory application with tools for users to build their own memory palaces, giving a taste for how these applications might look in practice. I no longer needed to finalize the campus dining hall environment, which still required extensive optimization to run on mobile hardware. Instead, I could prototype the application on a dedicated PC connected to the headset, enabling much more flexibility with optimization. The fastest method I found to generate a 3D reconstruction was using a semi-open-source Android application, 3D Scanner for AR Core [18]. This app uses the ARCore Android SDK to interface with Google Tango's deprecated CHISEL 3D reconstruction algorithm [19] for realtime dense reconstruction on a mobile device. With this, I created a rough scan of my room, consisting of approximately 250,000 faces and two 2K resolution textures. Since the room is much smaller than the dining hall, this dense scan is actually lightweight enough to run on the mobile hardware. This scan would serve as the backdrop for my demo memory palace.

4.3 Proof of concept

The prototype application was built in the Unity game engine, with scripts written in C#. While NeverMind was designed as a HMD application with an accompanying mobile app, I wanted to design a system that ran entirely on the headset. This requires a difference in user input, specifically a shift to spatial hand tracking. Users should be able to physically move mnemonics around the environment by grabbing them. The best way to accomplish this was to use a framework designed specifically for MR interfaces, specifically Microsoft's Mixed Reality Toolkit (MRTK). The framework is a package for Unity that support development for Microsoft's MR hardware lineup, including HoloLens. It doesn't officially support Oculus Quest, but Eric Prvncher developed an open source extension bridge to enabled MRTK development on the Oculus platform [20], enabling standard MR hand interactions.

I first decided what user tasks I wanted to support. For this proof of concept I targeted three tasks: creating 3D mnemonic representations, organizing mnemonics in the scene, and managing virtual objects. This led to the development of several features to support those tasks. For creating mnemonics, I added a feature to make queries to Google Poly, a free repository of curated and user-submitted 3D assets for VR. For instance, if the user wants to remember the Falcons football team, they can search for "falcon" and see an assortment of relevant models. Upon selecting an asset from the search results panel, it will be spawned in the world in front of the user. In order to make text queries, I had to implement a virtual keyboard that the user can type on with their hands. For this demo, it supports only alphabet characters. The keyboard floats in front of the user, following their movements, and it can be dismissed with a button in the corner. After the user taps the enter button, a similar floating panel will appear with search results. Each panel in this MR UI will tend to float in front of the user, but its position can also be fixed in the environment by de-selecting the "follow me" button. The user can drag the panel around much like traditional windows on a desktop by either directly grabbing the border with their hands, or from a distance with selection rays extending from their palms.

To organize mnemonics, the user can grab their spawned 3D models directly or from a distance and move them around the scene. They can use to hands to scale and rotate the objects, or they can perform these actions with one hand by using handles that appear around the object's bounding box. Sometimes, a mnemonic may be composed of multiple 3D objects (e.g. a dinosaur waving a flag). The user



(a) Real-world view

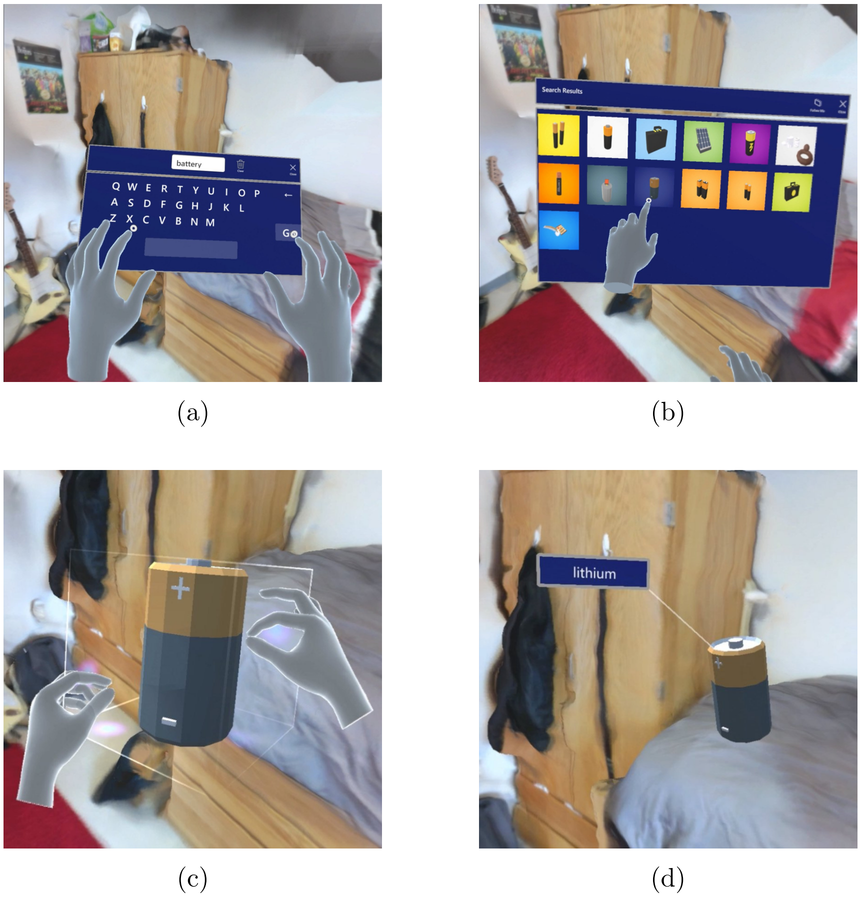
(b) Headset view

Figure 4: Alignment of real and virtual worlds enhances the effect of MR simulation. Real-world footage was recorded on a cell phone camera taped to the front of the Oculus Quest headset.

can overlap 3D assets with ease and re-position them as necessary. To make it clear that these objects represent one mnemonic, I add a labeling feature. The user can press a button to spawn a label, which consists of an anchor and a flag. The anchor and flag can be positioned independently by the user, such that the anchor points to a given mnemonic and the flag floats somewhere above it. Users can customize the text in the flag by tapping an edit button beside it, prompting them with the floating keyboard. Here, the user can write the source word that they intend to memorize. This makes it clear where each locus is in the scene. Lastly, the user can lock adjustments from being made which hides the text input buttons beside each flag and prevents any virtual content from being moved.

To manage objects, the user can pull up a panel with each of the 3D objects in the scene labeled by name and press a button to delete a desired item. In the future I would like to further develop this feature by allowing the user to manage and reorder labels, so they can quickly view the ordered wordlist. I would also like to add features to manage multiple word lists that may be shared within the same physical space.

In order to access all these features I include a menu that attaches to your hand



(a)

(b)

(c)

(d)

Figure 5: The flow for adding mnemonics to a memory palace. (a) Typing a 3D object query. (b) Selecting from search results. (c) Near manipulation of a mnemonic. (d) Labeling the mnemonic.

and appears when your palm is facing up. Along this menu are buttons to start a 3D object query, spawn a label, show the object manager panel, and lock/unlock the scene.

The application runs at a smooth framerate when run on my PC, and while I haven't tried building it standalone for the Quest, it should run smoothly there as well. The hand tracking is the most unstable part. As hand tracking is an experimental developer option on the Quest, sometimes it doesn't continuously detect a pinch motion, causing objects to be left behind when you try dragging them. This will likely be mitigated on future headsets that are designed with hand tracking in mind. It could also be addressed in software by adding a short delay to the release of interactable objects. I shared an early demo of the application on Twitter, where it caught the attention of MRTK-Quest's maintainer and his following, indicating that

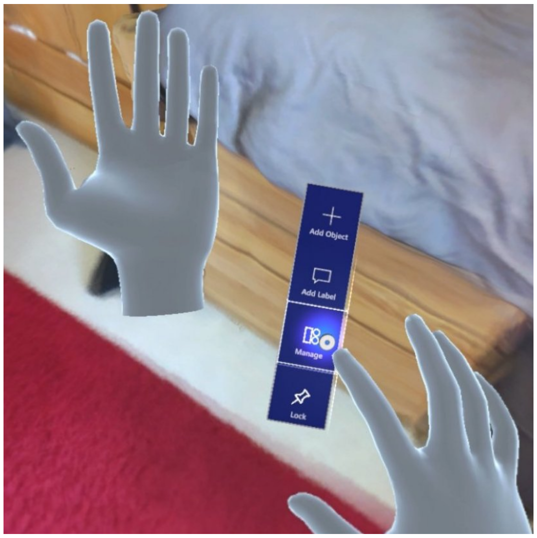


Figure 6: Initiating actions from the hand menu

there is a real interest in using VR to simulate MR. Since the app is designed on the broader MRTK framework, it can theoretically run on any supported device. For instance, by removing the reconstructed environment, this prototype could function as a true MR app on HoloLens.

Conclusion

5.1 Mixed reality simulation

There's a big difference between pitching an immersive application on paper and experiencing it for yourself. The 3D reconstruction task ended up being a big focus of this project for me, because I wanted to ensure the simulation was convincing enough to evoke a feeling of real presence in the space. However, I ended up using a low fidelity 3D scan of my own room in the eventual prototype. Even though this model was ridden with holes and blurry textures, it served its purpose just as well. My sense of presence was elevated by my ability to closely align the 3D reconstruction with my actual room, allowing me to reach out and feel the environment. Thus, I believed that the room I saw in the headset was my real room, despite the visual artifacts. Sometimes, if the virtual world was misaligned with the real world, I would become disoriented after taking off the headset, believing that I was actually standing in a different part of the room. This leads me to believe that convincing MR simulation does not require you to be in a physical space that aligns with the virtual one. Or perhaps haptic alignment is a key factor, but it only needs to occur a few times during onboarding to teach subjects to accept the virtual world as real.

I'm imagining a training exercise where subjects can be in a bare room, but have a real table in front of them aligned with a virtual table. After being granted freedom to touch the virtual table, they may be more likely to accept the virtual world as real, and the table can be removed to proceed with the main experiment. If I had the opportunity to run an experiment with human subjects, I'd be curious how their reaction compares to my own.

5.2 Mixed reality memory palaces

This experience has affirmed my belief that XR is a valuable tool for memory and learning. VR is already used by big companies like Walmart for employee training programs, and AR is used to provide on-site repair information at NASA. I'm sure that as XR devices become more widespread, some technique will rise above the others as a preferred information management system. Perhaps it will look like a virtual memory palace, or perhaps it will simply be 3D flashcards. Since the method of loci is so popular among the top memorizers, I think it deserves more attention, and the convenience of immersive memory palaces could make it more popular to practice.

Despite my faith in spatial memorization tools, I don't think that popular mixed reality memory palaces will look the way that I've demonstrated it in this project. My prototype assumes that users would walk around a real space to lay out their palace, and revisit that space to review it. Users may construct palaces in lots of different locations, so it's unreasonable to expect them to physically navigate between and within spaces in order to use this tool. I reached out to Aaron Ralby, founder of Linguisticator and creator of the MunxVR memory application, about his thoughts on AR as a platform for memory palaces. He pointed out that, "A key value of VR is being able to build structures that mirror the structure of the subject you're learning", and "If you built a [memory palace] that represented a subject perfectly as a virtual space, you could shrink it down to the size of coin and place it in a real space with AR, then grab and expand it as needed. Like books on a shelf". This approach would grant users a sort of multi-tiered interface to the MoL. On one level, they are organizing information within a uniquely crafted virtual space, and on the next level they are organizing each virtual space (representing some isolated subject) within a real space. Users would not have to travel between locations in order to visit other palaces, rather they could position each palace as a miniature within the confines of their existing space, and perhaps carry these palaces along with them for review anywhere they go.

5.3 Future work

Since I was unable to run the proposed experiment on environment familiarity, an obvious direction for future work would be to run the experiment or one similar to it, to address whether there is any benefit to organizing information within the familiar spaces that surround us versus novel virtual environments. As I mentioned earlier, new consumer devices with more advanced spatial sensors like LiDAR and time-of-flight that will make medium fidelity 3D reconstruction much more accessible, so future efforts will not have such difficulties with digitizing real spaces.

Another area that should be investigated is web-based immersive memory tools. Over the past year, web-based XR applications have grown in popularity, particularly due to the release of an official WebXR API for web browsers. WebXR enables users to access AR and VR applications using standard web technologies, so no native download is required. A web-based VMP could therefore be easily supported on a plethora of devices in both 3D and 2D formats. Furthermore, a web-based memory palace could easily be shared between users, so after one user carefully designs a memory palace for a particular subject, other users can add the same palace to their collection to support their own understanding of the material. In this way, learning a new subject could be as simple as clicking a link and taking a literal walk in the park.

5.4 Closing remarks

Through this work, I've shed some light on the area of immersive learning, specifically through applications of mixed reality. Thanks to this experience I've learned a lot about the XR field as a designer and developer. At the start of this thesis I had never developed for a 3D platform, but over tha past year I've completed several 3D projects for both traditional and immersive platforms. I learned a lot about 3D graphics and technical artistry. I met industry leaders at a XR-themed hackathon at MIT, and found a passionate community of XR enthusiasts online. I explored the growing fringe between reality and virtuality. I even learned how to learn.

I think immersive experiences will play a huge role in future learning. As the world faces large scale closures of educational facilities, we're face to face with the difficulties of learning and teaching through a 2D display. We can expect a new suite of learning tools to arise, and that's where I think spatial platforms will get their time to shine. While the specific benefits of each respective XR platform are yet to be determined, my experience developing this prototype leaves me hoping that memory palaces will earn a spot in our future toolbelt as a shared space for immersive learning.

References

[1] F. A. Yates, *The art of memory.* Random House, 1992, vol. 64.

[2] E. A. Maguire, E. R. Valentine, J. M. Wilding, and N. Kapur, "Routes to remembering: the brains behind superior memory," *Nature neuroscience*, vol. 6, no. 1, pp. 90–95, 2003.

[3] "International association of memory statistics." [Online]. Available: https://iam-stats.org/discipline.php?id=NUM60

[4] E. L. Legge, C. R. Madan, E. T. Ng, and J. B. Caplan, "Building a memory palace in minutes: Equivalent memory performance using virtual versus conventional environments with the method of loci," *Acta psychologica*, vol. 141, no. 3, pp. 380–390, 2012.

[5] J. B. Caplan, E. L. Legge, B. Cheng, and C. R. Madan, "Effectiveness of the method of loci is only minimally related to factors that should influence imagined navigation," *Quarterly Journal of Experimental Psychology*, p. 1747021819858041, 2019.

[6] J.-P. Huttner, D. Pfeiffer, and S. Robra-Bissantz, "Imaginary versus virtual loci: evaluating the memorization accuracy in a virtual memory palace," in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.

[7] E. Fassbender and W. Heiden, "The virtual memory palace," *Journal of Computational Information Systems*, vol. 2, no. 1, pp. 457–464, 2006.

[8] E. Krokos, C. Plaisant, and A. Varshney, "Virtual memory palaces: immersion aids recall," *Virtual Reality*, vol. 23, no. 1, pp. 1–15, 2019.

[9] A. Ralby, M. Mentzelopoulos, and H. Cook, "Learning languages and complex subjects with memory palaces," in *International Conference on Immersive Learning.* Springer, 2017, pp. 217–228.

[10] O. R. G. Rosello, "Nevermind: an interface for human memory augmentation," Ph.D. dissertation, Massachusetts Institute of Technology, 2017.

[11] N. Reggente, J. K. Essoe, H. Y. Baek, and J. Rissman, "The method of loci in virtual reality: explicit binding of objects to spatial contexts enhances subsequent memory recall," *Journal of Cognitive Enhancement*, vol. 4, no. 1, pp. 12–30, 2020.

[12] J.-W. Lin, H. B.-L. Duh, D. E. Parker, H. Abi-Rached, and T. A. Furness, "Effects of field of view on presence, enjoyment, memory, and simulator sickness

in a virtual environment," in *Proceedings ieee virtual reality 2002.* IEEE, 2002, pp. 164–171.

[13] M. Boland, "Xr talks: Recalibrating vr's future, part ii," Oct 2018. [Online]. Available: https://arinsider.co/2018/10/26/xr-talks-recalibrating-vrs-future-part-ii/

[14] E. Ragan, C. Wilkes, D. A. Bowman, and T. Hollerer, "Simulation of augmented reality systems in purely virtual environments," in *2009 IEEE Virtual Reality Conference.* IEEE, 2009, pp. 287–288.

[15] AliceVision, "Meshroom: A 3D reconstruction software." 2018. [Online]. Available: https://github.com/alicevision/meshroom

[16] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *arXiv preprint arXiv:2003.08934*, 2020.

[17] F. S. Bellezza, "Mnemonic devices: Classification, characteristics, and criteria," *Review of Educational Research*, vol. 51, no. 2, pp. 247–275, 1981.

[18] L. Vonásek, "3D Scanner for ARCore," 2020. [Online]. Available: https://github.com/lvonasek/tango/wiki/3D-Scanner-for-ARcore

[19] M. Klingensmith, I. Dryanovski, S. Srinivasa, and J. Xiao, "Chisel: Real time large scale 3d reconstruction onboard a mobile device using spatially hashed signed distance fields." in *Robotics: science and systems*, vol. 4. Citeseer, 2015, p. 1.

[20] E. Prvncher, "MRTK-Quest: Mixed Reality Toolkit (MRTK) extension bridge for Oculus Quest + Rift / S ," 2020. [Online]. Available: https://github.com/provencher/MRTK-Quest