Connecting Content to Space
The 4D Research Lab's experience with eXtended Reality, 2018-2020

Jitte Waagen, Markus Stoffer

University of Amsterdam

4D Research Lab

The 4D Research Lab Report Series has been established as an instrument to promote transparency regarding the 4DRL virtual visualisation projects, workflow and pipeline development and technical experiments. The aim is to maximize knowledge sharing, meta- and paradata communication and clarification of author- and ownership of 4D Research Lab products.

Jitte Waagen, Tijm Lanjouw

Connecting Content to Space - The 4D Research Lab's experience with eXtended Reality, 2018-2020

4D Research Lab Report Series 12

Author(s): Jitte Waagen, Markus Stoffer
Layout: Mikko Kriek
Editor(s): Tijm Lanjouw, Alicia Walsh

ISSN 2772-7734

January 2025

http://4dresearchlab.nl

Project datasheet

Project name	*Various:* Visualizing the process of facial reconstruction in AR (FR) Blending Past & Present (BP&P) Augmenting Artefacts (AA) Artificial Archaeological Intelligence (AAI) The Vlooienburg HoloLens Experiment (VB)
Date	01/2018 – 12/2020
Author of report	Jitte Waagen, Markus Stoffer, 4D Research Lab
Project initiators	J. Waagen
Project execution	J. Waagen (coordination/research/education/conceptual development), Markus Stoffer (research/technical development/coding, documentation), T.J.R. Lanjouw (conceptual/technical development), 4D Research Lab
Deliverable(s)	Various apps
Location and accessibility of project files	4D Research Lab archive, cloud storage. To access contact Tijm Lanjouw (t.j.r.lanjouw@uva.nl)
Project outcomes	Jeffra, C., Hilditch, J., Waagen, J., Lanjouw, T., Stoffer, M., de Gelder, L., & Kim, M. J. (2020). Blending the Material and the Digital: A Project at the Intersection of Museum Interpretation, Academic Research, and Experimental Archaeology. *The EXARC Journal*, *2020*(4). https://exarc.net/ark:/88735/10541 [details] Waagen, J., & Lanjouw, T. (2020). Augmented Blended Learning: In archeologisch onderwijs. *Archeologie in Nederland, 2020*(1), 48-49. Augmented Reality Good Practice at UvA video: https://youtu.be/wPNnm-FlnRs?si=D9fZ4N9WV1JUBF-0 AR/Vuforia/Unity video walkthroughs: https://youtu.be/ra8kVLVYAhM?si=at6jSukVAYNtSPf0 https://youtu.be/o0ybVu9SB7g?si=UwGD22iTqVL5IzA5

Abstract

From 2018 to 2020, the 4D Research Lab (4DRL) gained experience in the concept of Blended Learning, using object recognition software and historical reconstructions. Spanning four different projects, this report 1) explains the goals and content of the projects, and 2) focuses on the technical documentation of a Unity project making a 3D model explorable using a Microsoft HoloLens 2. This report addresses the need for sharing the actual technical solutions to application development challenges.

Introduction

In education and in museum environments, the ongoing trend of combining the strengths of physical and digital teaching methods includes the experimentation and use of extended reality technologies. This umbrella-term, 'extended reality' (xR or XR), includes virtual reality (VR), augmented reality (AR) and mixed reality (MR). The terms indicate the varying degree in which a digital application corresponds to, and interacts with, the real world making use of digital 3D objects and data retrieval methods. The application of these technologies in education is linked to the concept of Blended Learning. This is an innovation theory concerned with blending various forms of student-teacher interaction, with a strong emphasis on complementing traditional forms of teaching with digital methods. Common aims, among others, are to improve immersion and interaction, stimulate active learning attitudes, and increase the retention of knowledge. Due to the versatility of digital assets, e.g., multimedia and multisensory content, highly interactive learning experiences are possible, which can simulate the real world to various degrees (Chittaro & Ranon, 2007). In cultural heritage and archaeology, these technologies are also useful to give access to broken, lost, distant, or inaccessible material objects and worlds.

In the museum environment, the application of extended reality is often stimulated by the intention to creatively communicate knowledge and available material, utilizing the additional possibilities of interaction with the present collections. An important aspect of the creation of such applications is that an immersive and educational environment needs to be designed and created, fitting seamlessly into the existing physical configuration of the museum/exhibition, while avoiding or minimizing distraction from the original qualities of that museum/exhibition. To instrumentalize the pedagogical potential of extended reality, a workflow can be explored that incorporates communicative clarity and scientific grounding from the start (Meehan 2020). The educational potential of digital visualization may appear self-evident, as it can function as an extension of the limited possibilities to display everything a museum or archive has to offer, as well as make vulnerable objects explorable and resistant to degradation. However, this educational potential can be severely limited, for example when the digital visualization does not connect well with the physical context or lacks in interaction with museum content (Champion 2017, 17-19).

This report specifically deals with the initial explorations of development and integration of XR apps in various contexts, explaining backgrounds, motivations, solutions, and technical implementations. A critical assessment of the abovementioned issues is dealt with elsewhere (e.g. Jeffra et al, 2020), and thorough quantitative evaluations were not adopted at this early phase of experimental explorations, and therefore are both not addressed in this account.

The 4D Research Lab's first efforts using extended reality

In the fields of material and immaterial heritage, the 4D Research Lab (4DRL) continually experiments with, and evaluates conceptual approaches to, XR to find baselines for effective application. The endeavors with these techniques have been largely realized in the course of a series of projects funded by the University of Amsterdam in the context of education in the Faculty of Humanities, within the Amsterdam Centre for Ancient Studies and Archaeology (ACASA) specifically, and in close collaboration with the Allard Pierson archaeological museum, in what was dubbed 'Augmented Blended Learning' (Waagen and Lanjouw 2019). Before the projects described in this text, the 4DRL already had experience applying digital techniques to education. One initiative that was enriching reality with digital information layers was developed, used, and tested in ACASA education until 2019: a city walk with location-based historical information as the basis for student assignments, called Through the Looking Glass. Students could use a tablet and GPS to navigate the historical layers of Amsterdam and connect the information to the current urban environment. This combination of location and various layers of information allowed a better understanding of their relatedness. The engaging activity of navigating and 'puzzling together' information related to the real world, appeared to be an effective method in stimulating communication between students, as well as teaching skills of media navigation and critical observation of information (Moseley 2008).

The 4D Research Lab continued to test various techniques afterward, ranging from roughly 2018 up to 2020. The lab's experience with extended reality has been based on four main teaching innovation studies, alongside a series of tests to explore possibilities and to gain more experience in techniques. As these were initial explorations without any previous experience, trial and error, as well as accidental discoveries characterized the early developments. As such, the experiments with the Vuforia Augmented Reality engine started as a consideration of an interesting tool, but when experimenting with its integration into a wider workflow, recognition of its larger potential was realized. The exploration and development of embedding Vuforia, Unity and hardware workflows into the scientific process forms the backbone of these projects. The experience could be described as similar to what Meehan states about the integration of the project team, communication, and scientific argumentation (Meehan 2020). In the 4DRL projects it was clearly observed that the connection between app development, the purpose in education, and the involvement of the direct stakeholders, is the foremost requirement to making a XR project in education a success (Jeffra et al 2020).

As mentioned, this report aims to share the process of adoption of the software, related techniques, and connected scripts, in order to share the solutions to the problems encountered. One of the issues with the development of XR applications is the specificity of the implementation, the exact scripts and tools used. It is hoped that the inclusion of the technical details will be of use to those interested in applying, and experimenting with, these approaches. First, the chronological sequence of the projects that provided a context for the application of XR will be dealt with to elucidate technical choices made in the development process. The different lessons gained from every project will be succinctly highlighted, illustrating how the experiments gradually finetuned the 4DRL's approaches to the issues such as immersion, development, educational embedding, practical optimization of content creation, etc. Afterwards, a technical overview of the production of the Vlooienburg HoloLens app (using Unity 2019.4.17f1) is provided, giving an overview of the solutions as well as documenting the custom code (see appendices).

Case studies

1 Visualizing the process of facial reconstruction in AR, 2018-2019

This case-study was the beginning of the 4DRL experimentation with augmented reality as a formal project. The aim was to visualize the process of reconstructing a human face from fragments of an excavated skull of a Russian soldier who died in the battle of Castricum, The Netherlands, in 1799. The reconstruction process was done by the physical anthropologist Maja d'Hollosy, and the lab kept in touch with her to evaluate the goals and application of the app. A virtually reconstructed version of the skull was printed as a physical 3D model and made into a keychain. The developed app uses the Vuforia engine (see resources) to recognize this physical model as an AR target (https://youtu.be/RoA6toUafS0?si=eP23obdJ1ThN5GWV). The object-oriented development of the Unity game engine made it possible to add multiple models of the different reconstruction phases to augment, which is not natively possible with Vuforia. When the keychain is recognized, the reconstructed face appears as an exact overlay. By swiping left and right on the smartphone or tablet, the various steps in the reconstruction process fade in and out. In this way it allows the user to navigate intuitively through the workflow, from archaeological material all the way to reconstruction, increasing the transparency of the process of facial reconstruction and its scientific process (fig. 1). The project resulted in some useful lessons learned on model optimization for such an application, as well as the information value of the added visualization layers.

Figure 1. Render of the 3D scan of the original bone fragments and 3D models of the facial reconstruction by physical anthropologist and facial reconstruction artist Maja d'Hollosy (image: Tijm Lanjouw).

2 Blending Past & Present (BP&P), 2019

The Blending Past & Present (BP&P) project was the 4DRL's first experiment with augmented reality in the context of Blended Learning. This project was linked to an Art History course at UvA focusing on 16th century artists residing in Amsterdam. Students were asked to visualize a part of historical Amsterdam, virtually reconstructing a building in 3D. The 4DRL assisted the students in creating AR experiences that placed the reconstructions in the current urban setting, allowing a true scale projection of the 3D model on the current façade of buildings in the same location (fig. 2). In doing so, the student was asked to evaluate the reconstruction in terms of its spatial attributes and context now that they had this true-scale and (to a degree) contextualized representation of it. In addition, they presented their reconstruction, and were asked to reflect on the potential of augmented reality for presentation and communication. This project set out to explore the process of teaching integration and possible affordances of AR for learning experiences, forming the starting point for later projects experimenting with blended learning. From the technical aspect, it provided a lot of insight into the possibilities of augmented reality applications in an urban environment, as opposed to easier to control indoor applications; especially light conditions in combination with lack of contrast in the targets (in this case building façades) made it quite challenging.

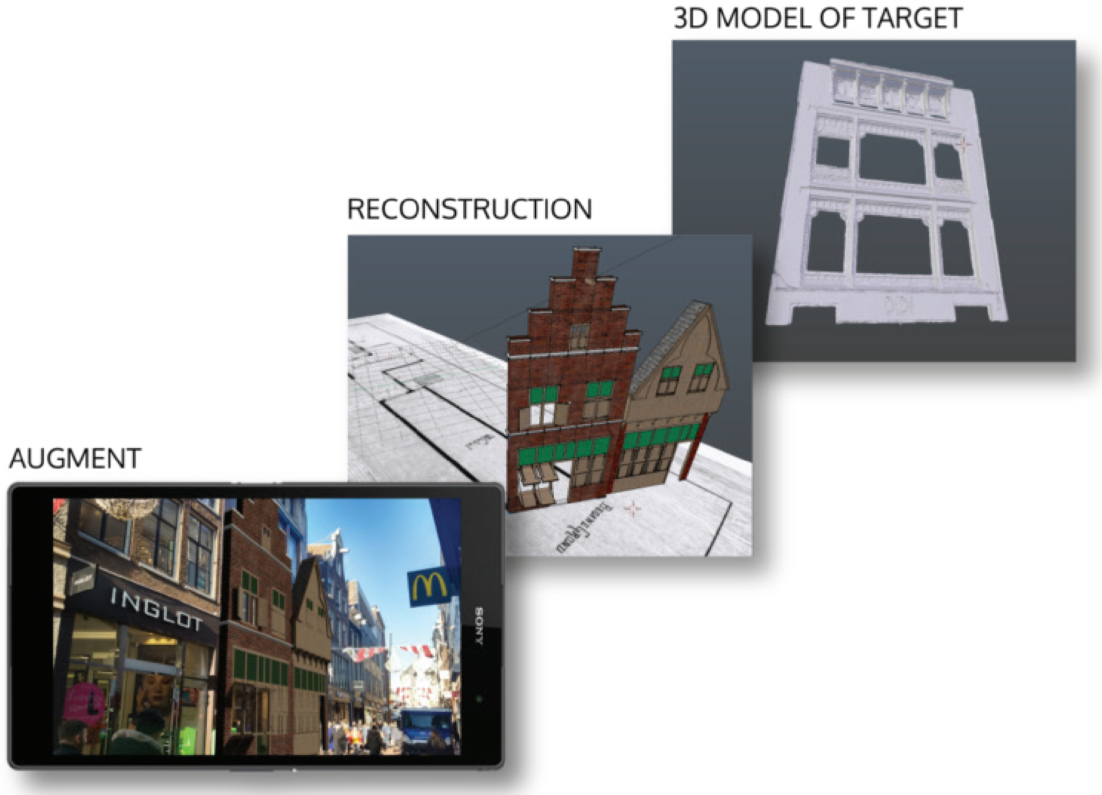

Figure 2. Steps in the process of making an augment: on the top right, a 3D representation of the target object based on a photogrammetrical surface scan, a real object in the world that can be recognized by the software as a target. In the middle a 3D reconstruction based on historical information such as maps, drawings, and descriptions. On the lower left the combined product in an app that can be installed on a mobile device displaying the reconstruction on the original location of the building (image: Tijm Lanjouw).

3 Augmenting Artefacts (AA), 2019

During the Augmenting Artefacts (AA) project the lab tested target recognition on smaller objects, namely on ceramic artefacts on display in cabinets inside a museum environment. Augmented reality was used to render artefacts connected to the central theme of the ancient Greek *symposion*, into an interactive learning experiment for students of Archaeology courses (fig. 3). The app provided a navigable 3D representation of the ceramic objects by allowing students to take them virtually out of the cabinet and inspect them in detail (rotate, scale). It also provided a voice-over of one of the ACASA teachers providing information and context on the object and showed relevant external references for more elaborate study materials. The additional material provided through the app provided ways of interpreting, understanding, and reflecting on the objects and their functional and cultural context.

This app was created with a project group that included various stakeholders (museum education specialists, teachers, developers) that discussed the potential of creating an AR app and fine-tuning it towards the intended levels of

Figure 3. Action photo of Augmenting Artefacts app; an Android app is used to scan the cabinet, recognize a target artefact, allow interaction with it, and provide additional contextual information layers (photo: Tijm Lanjouw).

interactivity and immersion. This project also garnered experience in sustainable cross-platform development; the location-based augmented reality app for smartphones could at first not run on older Android versions, and therefore a lot of upgrading needed to be done to reach maximum compatibility with most Android versions, and finally a build route to iOS devices was explored.

Reflection on this project contribution to workflow experience in multiple ways. First, referencing external sources which are not related to the spatial context, either the museum or the object, can be counterproductive, and often not optimally accessed through a smartphone interface. Second, for tailor-made educational purposes just offering added layers of information can be confusing; the more effective approach is to prepare a script that stages a narrative that can then determine and incorporate the density and placement of all relevant content in the app. Third, to discuss more often with project partners about the function of the produced app, the relevant teaching goals, and important concepts.

4 Artificial Archaeological Intelligence (AAI), 2019

For the project, a bit tongue-in-cheek called, Artificial Archaeological Intelligence (AAI), the XR experiments were transferred to the Microsoft HoloLens, which has some distinct added qualities in comparison with common mobile devices. The HoloLens is a headset that projects an augment directly in front of one's eyes, on a transparent screen. Utilising gestures and hand/head tracking, and

Figure 4. Screencap of MS HoloLens experience using the app developed for Artificial Archaeological Intelligence.

a virtual map of the environment where it can spatially attach digital assets, one can then move with respect to those objects. Both features altogether create a much more immersive and embodied experience. The ACASA course in this project focused again on ceramic vessels in the Allard Pierson Museum in Amsterdam, highlighting the production techniques of three of those objects.

Building forth on our learned lessons from Augmenting Artefacts, the group of stakeholders (i.e., museum education specialists, university teachers, digital archaeology specialists, xR-developers) was again formed and now also included students, and the development had an iteration setup, allowing all relevant feedback to be gathered and incorporated in bug fixing and improvement, thereby creating a smooth-running application that was finetuned for its purpose. Also, the app content was coordinated from the start following a script and included no links to external content; instead, it consisted of recorded videos showing the different technical steps of the vessel's reproduction, a voice overlay narrating this process, a modifiable 3D model of the vessel, and a glossary of terms (fig. 4). It formed an environment where a student can discover knowledge at their own pace in a kind of 'virtual lab', without the actual presence of the teacher.

This project was also evaluated through a qualitative questionnaire. Students tested the app in two sessions; two HoloLenses were available for six to seven people per group, and two to three people were available for any required assistance. The app received quite positive reactions and incited the students to think about the possibilities of augmented reality use in education. It bolstered the experience that using a solid script to fine-tune content is paramount and provided some inspiring insight into how using a head mounted display (HMD) caters for additional immersion and control and thus student enjoyment and satisfaction. Points of improvement were partially connected to the device, as the HoloLens had a small area of display and controls through certain gestures that are not easily recognized. Things that could be improved on the development part of future projects included the explanation and instruction of the controls and navigation.

Experience with using the first HoloLens for Blended Learning proved its potential, but also some hurdles of using augmented reality for archaeological purposes (Jeffra et al. 2020). The counterintuitive controls, and limited available headsets increases the waiting time for people to enjoy the augmented reality experience. Furthermore, the level of comfort with the technology differs from person to person, and other researchers of virtual reality experiences note that 'external assistance for navigation or understanding of contents' counts as a main disruptive factor of an immersive experience; 19.1% of given reasons for a disrupted experience (Pujol-Tost 2019, 11-12). Finally, the experience in augmented reality also resulted in awareness about the requirements for a seamless and well-embedded digital educational experience in a context where it provides added value.

5 The Vlooienburg VR and HoloLens experiment, 2019

At the same moment of the AAI project, an app was produced for the Jewish Historical Museum in Amsterdam, in the context of a large research project where ACASA researchers, historians, municipal archaeologists and experts on Jewish history and culture of the museum were involved (Waagen and Lanjouw 2020). The 4DRL developed a large on-screen (without a headset) virtual reality scene based on historical, archaeological, and architectural research on the former Vlooienburg neighbourhood in Amsterdam (fig. 5). Rather than being strictly used in a controlled educational environment, this app aimed to be a complete stand-alone package, including in-app navigation, support, and being as intuitive as possible (see resources for a link to a screencast of the app). This had been developed as a desktop app running on Windows, intended to be used with a large multi-touch table. Technically and content-wise, it proved to be a valuable experience. The content consists of a central scene with a 3D model of historical Vlooienburg, of which a specific period can be selected, allowing for different virtual reconstructions according to the respective historical phase. The user can move to two other scenes, which are a nearby neighbourhood Valkenburgerstraat, and an interior scene with furniture and objects that one would encounter in a Jewish household. Besides camera movement, the user interface and information on time periods, the interior scene included more elaborate, object-focused functionality. Several objects on the walls and table in the scene could be highlighted and selected to get specific information, giving the opportunity to discover some historical facts in the 3D scene, rather than reading them without the visual context.

Figure 5. Screencap of the Vlooienburg on-screen VR experience, interior of a house on 16th century Vlooienburg.

Although this app was more of a practical product, again valuable lessons were learned to optimize workflow dealing with such elaborate virtual reconstructions. The size and required interactivity of the app forced optimization of functional design and resource management. In addition, and as a spin-off result, the size of the project with a huge number of 3D models set the stage to produce certain standards of modelling, with which the timeframe of production can be estimated in a more accurate way.

Ultimately, the substantial size of the virtual reconstructions prompted the 4DRL to consider the current device's capability of interactively displaying a large number of polygons. As an experimental set-up, the 3D Model of Vlooienburg from the last project, this time using a single period, was used in an XR app using the HoloLens 2. The goal was to explore possibilities and challenges in rendering an open space, to add some annotation and streaming or collaboration functions, and to reflect on usability for teaching and interdisciplinary discussion. However, the experiments were rather harshly interrupted by the Covid-19 epidemic, hindering laboratory work, and specifically the effective testing and deployment of head-mounted displays due to contamination risks.

Current avenues of research and development

Due to the Covid-19 epidemic, focus shifted to the potential implementation of online collaborative virtual reality platforms such as Hubs (https://github.com/Hubs-Foundation), where in the subsequent years pilots and finally a research project have been developed with (Virtual Past Places, see resources and Waagen et al. 2024).

Figure 6. Screencap of the Vlooienburg HoloLens 2 experience, 16th century Vlooienburg. On the left side a 'bird's eye view' together with the floating menu that follows the user, and on the right a streetview-like perspective with the lab in Amsterdam in the background.

Conclusion

The evolution of blended learning through the integration of extended reality technologies carries the potential to advance educational methodologies, particularly within the realms of cultural heritage and archaeology. During the projects from 2018 to 2020, the 4D Research Lab gained experience in the technical development and reflection on the use of technology in education.

The lessons learned from these projects highlight the necessity of documented development, well prepared content, and an intensive stakeholder collaboration that is not limited to sharing goals and individual progress but includes critical reflection and understanding of strengths and limits of media. Our experience culminated in the Artificial Archaeological Intelligence project from 2019, where network collaboration, thoughtful consideration of the various perspectives, and a balanced learning application led to a satisfying result that was very positively reviewed.

The 4D Research Lab continues to further develop efforts in the field of blended learning, aiming to enrich both the educational landscape as well as the field of research of archaeology and cultural heritage with digital applications and methods. For anyone who aims to create any educational apps, both the technical details in this report as well as further material on the 4D Research Lab YouTube hopefully provides a useful resource.

Bibliography

- Champion, E., 2017. The role of 3D models in virtual heritage infrastructures, in: A. Bernardou, E. Champion, D. Costis & L.M. Hughes (Eds.): *Cultural Heritage Infrastructures in Digital Humanities*. 15-35.

- Chittaro, L., Ranon, R., 2007. Web3D technologies in learning, education and training: Motivations, issues, opportunities, in: *Computers & Education,* 49(1). 3-18.https://web.archive.org/web/20181215224524/https://support.ingress.com/hc/en-us/articles/360002062447-Interacting-with-Portals

- Jeffra, C., Hilditch, J., Waagen, J., Lanjouw, T., Stoffer, M., de Gelder, L. & Kim, M.J., 2020. Blending the Material and the Digital: A Project at the Intersection of Museum Interpretation, Academic Research, and Experimental Archaeology, in: The EXARC Journal, 2020 (4). https://exarc.net/ark:/88735/10541.

- Meehan, N., 2020. Digital Museum Objects and Memory: Postdigital Materiality, Aura and Value, in: *Curator: The Museum Journal*. https://doi.org/10.1111/cura.12361.

- Moseley, A., 2008. An alternative reality for Higher Education? Lessons to be learned from online reality games, in: *International Journal of Game-Based Learning, 2(3).* 32-50.

- Pujol-Tost, L., 2019. Did We Just Travel to the Past? Building and Evaluating With Cultural Presence Different Modes of VR-Mediated Experiences in Virtual Archaeology, in: *J. Comput. Cult. Herit.* Vol. 12 (1). https://doi.org/10.1145/3230678.

- Waagen, J., & Lanjouw, T., 2019. Augmented Blended Learning: In archeologisch onderwijs. *Archeologie in Nederland*, 48-49.

- Waagen, J., & Lanjouw, T., 2020. De 3D-reconstructie van Vlooienburg: hoe wordt een buurt gereconstrueerd, in: Berg, H. & van Kempen, K. (eds.): *Waterlooplein: De buurt binnenste buiten*. 112-123.

- Waagen, J., Lanjouw, T., & Huurdeman, H. 2024. Virtual past places. Reinventing the classroom using virtual reality in humanities education. *AHM Conference 2024: "Heritage, Memory and Material Culture,"* 78–84. https://www.aup-online.com/content/papers/10.5117/9789048567638/AHM.2024.014

Resources

4D Research Lab – Augmented Blended Learning.
https://4dresearchlab.nl/education/augmented-blended-learning
(accessed 20-04-2023).

4Dresearchlab.nl blog.
https://4dresearchlab.nl/virtual-past-places-a-collaborative-vr-for-storytelling-and-education/

Mixed Reality Toolkit.
https://github.com/Microsoft/MixedRealityToolkit-Unity/releases (accessed 20-04-2023).

Documentation for the MRTK.
https://learn.microsoft.com/windows/mixed-reality/mrtk-unity/mrtk2/?view=mrtkunity-2022-05
(accessed 20-04-2023).

Unity documentation for the Camera class.
https://docs.unity3d.com/2019.4/Documentation/Manual/class-Camera.html
(accessed 20-04-2023).

Unity Real-Time Development Platform.
https://unity.com/
(accessed 20-04-2023).

Vuforia Engine Developer Portal.
https://developer.vuforia.com/
(accessed 20-04-2023).

Youtube - 4D Research Lab - Vlooienburg/Valkenburgerstraat historical VR.
https://www.youtube.com/watch?v=MADUWGyR-oQ
(accessed 20-04-2023).

Youtube - 4D Research Lab - Tutorial Vuforia in Unity for Smartphone (Android).
https://www.youtube.com/watch?v=ra8kVLVYAhM
(accessed 20-04-2023).

Youtube - 4D Research Lab - Tutorial Vuforia in Unity for Microsoft HoloLens.
https://www.youtube.com/watch?v=o0ybVu9SB7g
(accessed 20-04-2023).

Technical documentation

The projects 1-3 described above made use of the Vuforia software with some custom scripts; the workflows with the former are explained in online tutorials by the 4DRL (see resources) and the latter are available on reasonable request. Project 5 described above has been a fully customized Unity application, also available on reasonable request. The HoloLens apps of project 4 and 5 described above used the Mixed Reality Toolkit; the tutorial below illustrates the technical setup of the Vlooienburg visualisation with custom functionality.

Visualizing the process of facial reconstruction in AR (FR)	Vuforia: see online tutorial (resources) and custom scripts
Blending Past & Present (BP&P)	Vuforia: see online tutorial (resources) and custom scripts
Augmenting Artefacts (AA)	Vuforia: see online tutorial (resources) and custom scripts
Artificial Archaeological Intelligence (AAI)	HoloLens; Mixed Reality Toolkit/Unity
The Vlooienburg HoloLens Experiment (VB)	HoloLens2; Mixed Reality Toolkit/Unity: Technical workflow below

Technical workflow for the Vlooienburg HoloLens 2 experiment

The following hardware and software were used to produce the augmented reality application.

Unity version	Unity 2019.4.17f1 Personal (released 22 December 2020)
Unity modules	Universal Windows Platform Build Support, Windows Build Support (IL2CPP)
External plugins	Mixed Reality Toolkit (MRTK) version 2.5.3 (Foundation package, Tools package, Examples package) (see resources)
Code editor	Visual Studio 2019 (required to update to version 16.8 or later)
Windows version	Windows 10 SDK 10.0.19041.0
HoloLens hardware	HoloLens 2, firmware 11973.01.01.200917.F
HoloLens operating system	Universal Windows Platform 10.0.19041.1131
Additional material	A USB cable to configure the HoloLens via the Windows Device Portal (Chrome browser), using a local internet connection

Setting Build Settings and preferences
First, when the new project is made, go to File (menu) -> Build Settings and change the build platform to Universal Windows Platform. The build settings used here look like this (fig. 7):

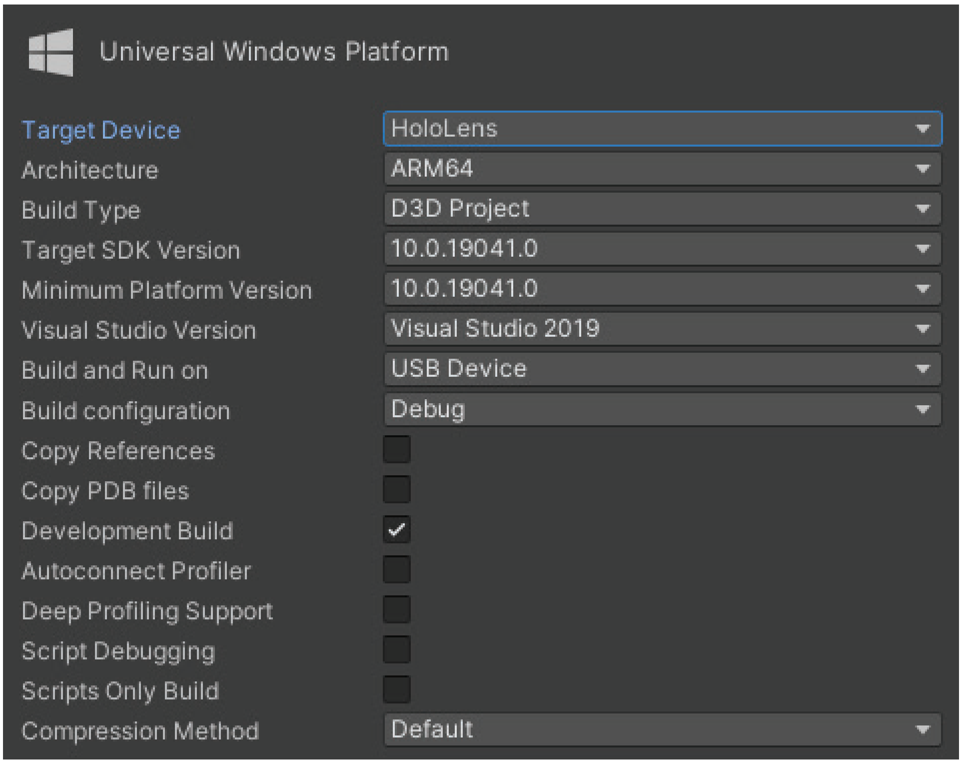

Figure 7. The Unity build settings screen for the Universal Windows Platform, necessary to build to the HoloLens.

The ‘Development Build’ option is useful to see whether there are framerate drops or heavy loading times, visualised as a bar showing a red or green status of the frames per second. Turn this option off to remove the diagnostic functions when the app is ready to be built.

There are a few other things to add:

- Once you have downloaded the Mixed Reality Toolkit (MRTK), import it through Assets (menu) -> Import Package (submenu) -> Custom Package. Then navigate to the location where you downloaded the .UnityPackage files of the MRTK.
- Import the TMP Essential Resources through Window (menu) -> TextMeshPro (submenu) -> Import TMP Essential Resources. Since normal text components of Unity are less suited for 3D, the MRTK and the HoloLens depend on the TextMeshPro’s components and work more fluently with them.

It is also useful to know that at any point with the MRTK toolkit imported, you can look at Mixed Reality Toolkit -> Utilities -> Configure Unity Project to reopen the configuration screen that pops up at import. The project and UWP settings recommended are generally the optimal settings for Mixed Reality building. Check whether 'Audio Spatializer' is then set to MS HRTF Spatializer.

The new and old Unity XR pipelines

For the MRTK as it is now, the Legacy XR pipeline from before Unity 2019.3 is used. There is a single MRTK profile that does work with the new XR Plug-in, but it is not optimised and for now limits developing options (see resources). The normal XR pipeline is found in Project Settings -> Player -> XR Settings (Deprecated Settings). This is disabled if the new SDK (Windows XR Plugin) is installed, in which case it first needs to be removed in Package Manager. After removing SDKs related to the new pipeline, it is possible to enable [Virtual Reality Supported] in the deprecated settings. The warnings about this being deprecated or unusable can be ignored, and the MRTK tutorials themselves encourage to do so. The XR Settings option may ask you to turn off 'Built-in VR' and other Mixed Reality components which are already initiated.

Once the 'Virtual Reality SDKs' screen appears in this sub-menu, add the 'Windows Mixed Reality SDK' via the plus button. It is also recommended that rendering is set to "single-pass instanced rendering path", because this synchronises the rendering for both eyes. It is useful for the performance of most apps to set "Depth Format" to 16-bit depth (apps that require 24-bit depth are those with masks which are not a 'RectMask2D', which use a stencil buffer).

Note that for this version of the MRTK, the new XR Pipeline does not yet work correctly. Disable the 'XR Plug-in Management' menu in project settings, as well as the 'Windows XR Plugin' package (which is now at version 2.4.1 or so). Do not enable the Windows Mixed Reality option in XR Plug-in Management, or you will need to remove the package manually and enable VR Support again.

Direct Emulation of the HoloLens 2

Before starting the development of any HoloLens app, it is useful to have a workflow where you can test directly from Unity to the HoloLens. First, you need to download the "Holographic Remoting Player" on the HoloLens 2. Then in Unity, open the settings in Window -> XR -> Holographic Emulation (Deprecated). If there is still a message that implies there are packages from the new pipeline installed, look for 'XR plug-in management' and 'MockHMD XR plugin' or other packages implying the use of the XR Plugin system, and remove them; do not remove the Windows Mixed Reality package (see the previous paragraph on the XR pipelines).

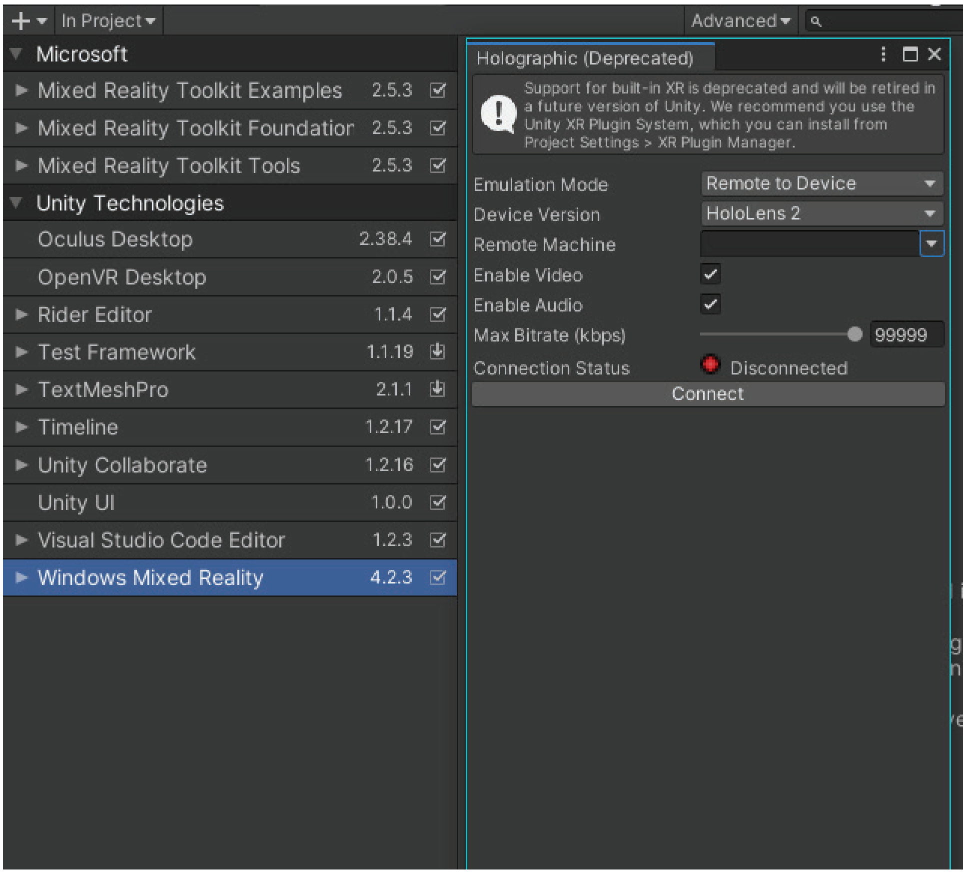

Figure 8. When Unity does not see any of the new pipeline components, the configuration for Holographic Emulation is shown.

For “remote machine”, type the IP-address in the same way you use it to reach the Device Portal on the PC. Unity might call the error ‘HandshakeUnreachable’ in this situation, as this means the HoloLens went to sleep mode, or the Holographic Remoting app is not turned on.

Step 2 – Implementing the Mixed Reality Toolkit

To add the MRTK to any scene, go to the scene that you want to make a MR scene, and click the “Mixed Reality Toolkit” menu, and then “Add to Scene” and “Configure”. There are many things to configure in the ‘MixedRealityToolkit’ option. Before you can change up a profile or sub profile here, you need to clone it. For the HoloLens 2, clone the ‘DefaultHololens2Configuration’ profile.
It is also recommended to have a separate user layer for Spatial Awareness: go to the Spatial Awareness sub-menu in MixedRealityToolkit, set “Enable Spatial Awareness System” and choose the default system type. Finally, clone the “Spatial Observer” profile, scroll to “Display Settings” at the bottom of the properties, and set “Display Option” to none. This causes the HoloLens to look for the environment to interact with, but not constantly visualise it as a mesh (which demands processing power).

As long as you are developing the app, you can also now enable diagnostics, which helps you keep an eye on the used processing power during development. When adding 3D objects to the scene, it is useful to run your app using diagnostics to see if they are not too heavy. On the HoloLens 2, it will create a status bar (visible in fig. 4) through which you can keep an eye on the frame rate and rendering load. If this changes too much whenever your 3D model is in the view frustum, the model is too heavy to be used in the app in its current form. If your 3D model is either small or is resizable, it is possible to get your entire model into view: when viewed from the side with the most complex 3D topology, the app will return the lowest possible frame rate. For augmented reality, we noted a minimal frame rate of 20, as the refresh rate will interfere heavily with spatial anchoring below this rate. This causes shaking that interferes with the user experience, so it is recommended to stay above it. Note that this also depends on how big your augment is in the real world. A low frame rate that is not problematic in augmented reality will be more noticeable in virtual reality, as the glitches in graphics can cause nausea.

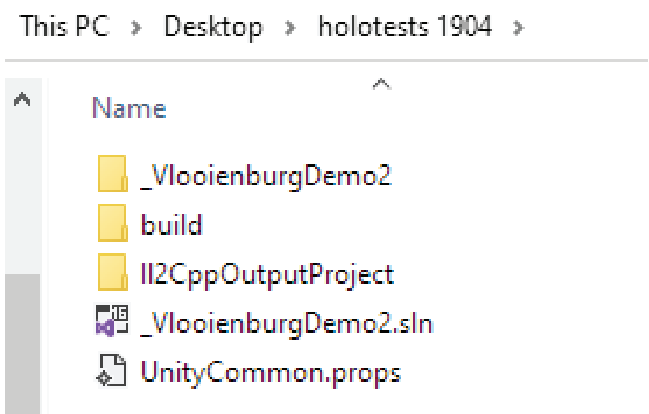

Figure 9. An example of the files produced by Unity when building for the HoloLens.

Step 3 – Adding the game objects to the Unity Hierarchy

In this sub-chapter, the Unity hierarchy of "game objects" and components used for building the augmented reality experience are summarised and explained (fig. 10). In general, there are game objects that are 'prefabs' (prepared objects) part of the MRTK, and our additions to these objects, as well as the 3D objects of Vlooienburg. Below, the layout of the most important game objects is shown. They will be explained from top to bottom.

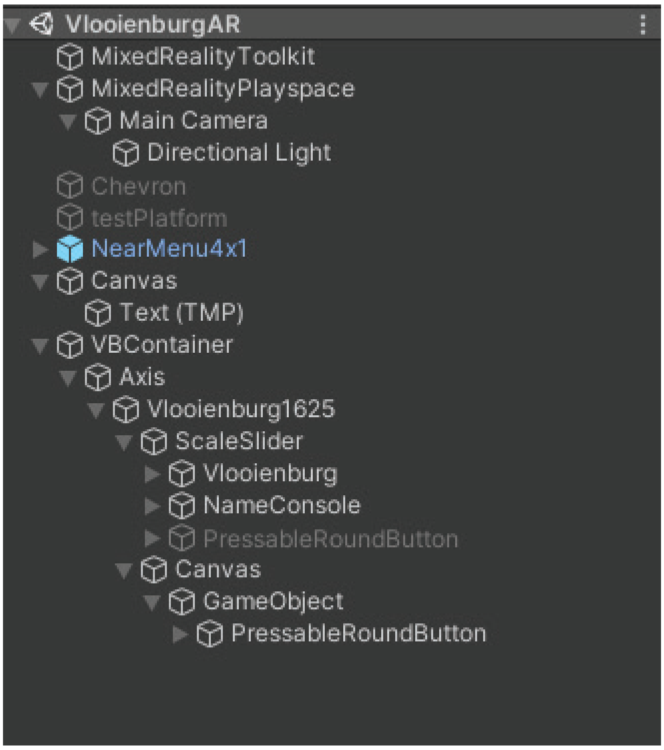

Figure 10. The Unity hierarchy of the finished app, where blue names indicate a prefab game object (in this case produced by the MRTK), and grey names indicate a disabled game object to be enabled during playtime. Individual objects are described in the text.

Objects controlling the MixedRealityToolKit

First, the 'MixedRealityToolkit' and 'MixedRealityPlayspace' objects are standard objects part of the MRTK. The former does not have any representation in world space, and only contains a single component which provides all the settings for the MRTK. Before edits can be made to these settings, a standard profile (in this case the DefaultHoloLens2ConfigurationProfile) is cloned. This opens up options in the component. Some of the separate tabs (Camera, Input, etc.) also has a standard profile, which can once again be cloned. In this case, the Camera profile has been cloned to edit the display settings, and the Spatial Awareness profile has been cloned to set its display to 'none'.

The 'MixedRealityPlayspace' object is an empty object but contains a child object: the standard Unity camera (see resources) with an added Input Module component, which prepares to receive input when run on the HoloLens. In this case, a simple directional light source is placed as a child of the camera. The light sources in augmented reality respond differently than normal, but without the light the entire model turns dark.

The ‘Chevron’ object is a visual indicator that activates when the user loses the main model out of sight and presses the “Toggle indicator” button. It is disabled on start. A mesh renderer with the standard MRTK chevron mesh activates according to a solver handler (meaning the object tracks a certain target, in this case the headset itself), and according to a directional indicator pointing towards the Vlooienburg model.

The ‘testPlatform’ object is the only object never activated in the HoloLens version of the app: it is useful for there to be a 3D plane for testing purposes in Unity, as long as you do not have the spatial awareness functionality of the HoloLens to detect actual planes for you. This is what the model is placed on in the in-editor testing phase.

The ‘NearMenu4x1’ object is a standard MRTK prefab (fig. 11): it is a menu of a certain number of buttons (here 4 columns by 1 row). Normal buttons do not work well in augmented reality, so the standard components ‘PressableButtonHoloLens2’ and ‘Interactable’ make the buttons functional. From the standard button prefab, the name, icon, and the scripted event ‘OnClick’ has been changed. The Vlooienburg augment and its dimensions can be interacted with through this small menu, which moves along as the user moves around, thanks to the “Radial View” component on the parent object. This can be turned off by pinning the menu with the pin icon, visible on the right.

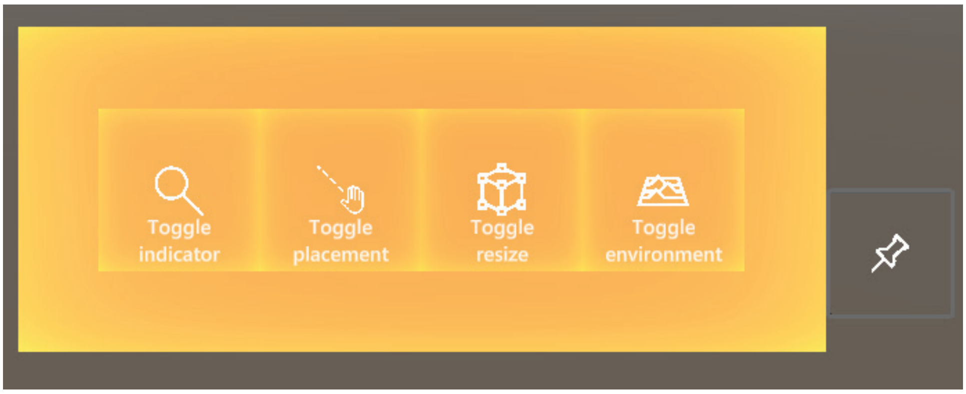

Figure 11. How the NearMenu4x1 object looks. When wearing the HoloLens, it follows the user in the bottom of the screen.

The first button toggles the Chevron object (either enables or disables it), using a custom Toggle script (see appendix I). The second button has a more complex function: it enables placing and rotating the Vlooienburg model on a plane detected by the sensors of the HoloLens, along with a button in world space to place the model. This is done by turning on the ‘TapToPlace’ and ‘ObjectManipulator’ components described below. These are not disabled with this button, instead being disabled on completing the placement. The third button uses the same Toggle script as the first button, to toggle the pinch slider object which controls the size of the 3D model. The fourth button uses the Toggle script to turn part of the 3D object on or off. This includes the water and larger environment, which becomes confusing when the model is sized up to study the houses of Vlooienburg.

The Pinch Slider is a standard MRTK object (fig. 12), but the visuals have been changed slightly, as flat 3D objects are much harder to notice in an augment. We added an effect similar to volume sliders in most devices: some coloured bars increasing in size. The colour change in 3D is achieved through a custom cross-section script (Border Plane, see appendix I), as well as a material using a scripted custom clipping shader (see appendix). The clipping plane is connected to the slider 'button', which controls the size of the Vlooienburg model by manipulating the 'Scale Slider' script (see appendix I). The latter script was necessary to listen to the event data produced by the MRTK slider object and make it into a value that affects the three dimensions of the model.

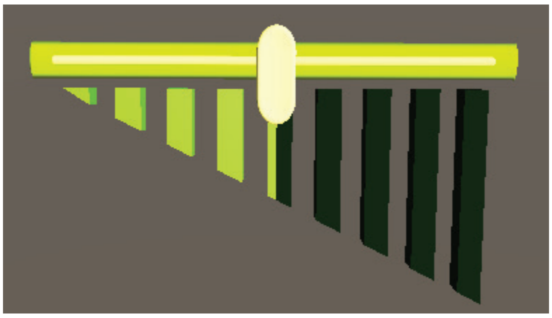

Figure 12. The Pinch Slider. Whereas a slider bar is much harder to find in augmented reality, including a visual indicator with a larger surface and a changing colour increases the visibility.

The next game object, a simple world space canvas called 'Canvas', just contains an introductory text (using TextMeshPro, called 'Text (TMP)') telling the user to place the model on a flat surface. It is only enabled when starting the app and is then disabled.

Objects that contain the Vlooienburg 3D Model

'VBContainer' is the most important game object, as it contains the Vlooienburg 3D model itself. Its own component is only a Toggle script that the interface buttons interact with. Furthermore, this object and its only child, 'Axis', have values for position, rotation and scale that centre the model correctly. The 'Axis' object offsets the rotation and position of the 3D model, putting the centre on the intersection in the centre of Vlooienburg. The next child object in the hierarchy controls the interactivity with the model. The standard MRTK component "Tap To Place" is used here, which uses the spatial awareness functionality of the HoloLens to place the 3D model on a detected plane. It has separate events for when the placement begins and when it ends: when the model is placed, the introductory message disappears and the Tap To Place component disables itself. Once again, a ‘solver handler’ is used to keep track of the head position. Finally, an ‘Object Manipulator’ component enables rotation of the 3D model as long as it is active. This component normally enables movement and rotation on all axes, but two constraining components disable movement and disable rotation on anything other than the vertical axis. To facilitate rotating the model, a separate button above the 3D model disables this function, rather than it being disabled whenever the model is placed or touched.

‘Vlooienburg1625’ has two child objects: an object called “Scale Slider” which receives the scale input from the slider, and a canvas with a single button (hovering over the centre of the model) which disables rotation. The Scale Slider object is parented to the actual model, as well as to a name tag for street names (for testing purposes). For the rest, there is only the Vlooienburg game object parented to the Scale Slider. This game object has parts of the model as child objects, which are either separate objects or have separate materials. Two kinds of objects have another parent: the wood stacks of Vlooienburg, which consist of many small game objects, and the partial models that are not part of the main island itself, grouped together under a ‘Toggle’ parent which is turned on or off by the ‘Toggle environment’ button.

Final notes

For any scripts not included in the standard Unity environment or in the MRTK, please see appendix I. Also, as describing the app in text might not give an adequate impression of how an augmented reality app is produced in practice, two videos on the 4D Research Lab YouTube channel tackle the process of developing a Vuforia App (see resources). The first is a general explanation of how to use Vuforia in Unity, and the second one applies this to the HoloLens specifically (using the AAI project as an example).

Appendix I.
Custom scripts for Vlooienburg

The custom C# scripts used in the Vlooienburg AR app are pasted here. Other scripts are used which are included in the MRTK, according to the description in the technical chapter in this text.

1. Toggle

```
using UnityEngine;
public class Toggle : MonoBehaviour
{
public void ToggleTTP(GameObject toggleObj)
{
toggleObj.SetActive (!toggleObj.activeSelf);
}
}
```

2. Border Plane

```
using UnityEngine;
public class BorderPlane : MonoBehaviour
{
public Material borderMat;
void Update(){
Plane plane = new Plane(transform.right, transform.position);
Vector4 planeVisualization = new Vector4(plane.normal.x, plane.normal.y, plane.normal.z, plane.distance);
borderMat.SetVector("_Plane", planeVisualization);
}
}
```

3. Scale Slider

```
using UnityEngine;
using Microsoft.MixedReality.Toolkit.UI;
public class ScaleSlider : MonoBehaviour
{
protected float ObjScale;
public float ScaleFactor;
public void OnSliderUpdated(SliderEventData eventData)
{
ObjScale = (eventData.NewValue+0.2f)*ScaleFactor;
transform.localScale = new Vector3(ObjScale, ObjScale, ObjScale);
}
}
```

4. Custom Clipping Shader

```
Shader "Custom/CustomClippingShader"
{
Properties
{
_Color ("Color", Color) = (1,1,1,1)
_MainTex ("Albedo (RGB)", 2D) = "white" {}
_Glossiness ("Smoothness", Range(0,1)) = 0.5
_Metallic ("Metallic", Range(0,1)) = 0.0
[HDR] _CutoffColor("Cutoff Color", Color) = (1,0,0,0)
}
SubShader
{
Tags { "RenderType"="Opaque" }
Cull Off
CGPROGRAM
#pragma surface surf Standard fullforwardshadows
#pragma target 3.0
sampler2D _MainTex;
half _Glossiness;
half _Metallic;
fixed4 _Color;
float4 _Plane;
float4 _CutoffColor;
struct Input
{
float2 uv_MainTex;
float3 worldPos;
float facing : VFACE;
};
void surf (Input IN, inout SurfaceOutputStandard o)
{
float distance = dot(IN.worldPos, _Plane.xyz);
distance = distance + _Plane.w;
clip(-distance);
float facing = IN.facing * 0.5 + 0.5;
fixed4 c = tex2D (_MainTex, IN.uv_MainTex) * _Color;
o.Albedo = c.rgb;
o.Metallic = _Metallic;
o.Smoothness = _Glossiness;
o.Alpha = c.a;
}
ENDCG
}
FallBack "Standard"
}
```