VitruviAR: Interactive Augmented Reality for Early Design Stage Applications

Taeyong Kim, George Guida, Dongyun Kim

Harvard Graduate School of Design, US
taeyong_kim@gsd.harvard.edu
gguida@gsd.harvard.edu
dongyun_kim@gsd.harvard.edu

Abstract. The increasing development of Augmented Reality (AR) applications have found prevalence within construction stages of architectural projects. The workflows developed within digital fabrication and assembly processes provide insights on how the design cycle could be completed through mixed reality. In this paper we present VitruviAR, an AR prototype for handheld devices which focuses on the design ideation stages of a project through an intuitive user interface and multi-functional toolset. Three design methodologies relating to the act of sketching digitally are proposed: freeform 3D sketching in point-based meshes, additive 3D sketching with primitive and scanned objects, and computational 3D sketching via a User Datagram Protocol (UDP). These each demonstrate engaging ways of designing and visualizing new spaces and interacting with urban contexts in real-time.

Keywords: Augmented Reality, Concept Design, Accessible User Interface, Visualization, Generative Design

1 Introduction

Over the last few years, augmented reality (AR) has emerged as a valuable tool in architectural and design practices. It is increasingly commonplace to use this technology in the visualization and fabrication of standard and non-standard geometries through the uses of headsets and mobile devices. These applications typically adopt pre-scripted interfaces and pre-defined geometries as inputs, leaving limited space for users to design freely with real-time feedback. By providing improved control to users through an accessible user interface (UI) and mixed-platform tool kit, the design cycle of AR applications can be achieved, beginning from an initial 3D sketch.

In this paper, we present Vitruviar, an AR-based prototype that assists designers in the early design stages. By focusing primarily on the ideation process, we will introduce methods to rethink traditional design tools through three methods: *Freeform, Additive, and Computational 3D Sketching*. By explaining this framework, we will describe the accessible user interface used to tie mobile AR APP to generative workflows for experienced and inexperienced designers alike and conclude with limitations and future outlooks.

Figure 1. User-generated arch system with embedded computation. Source: Author of this paper, 2021.

2 Background

Over the past decade, AR technologies have developed significantly within the architectural discipline, particularly the assembly and digital fabrication workflows of projects (Song et al., 2021). In the same way 3D printers are decentralizing the manufacturing process, AR and VR technologies are creating new methods for humans to access and create information. While the seamless overlay between the digital and physical has provided clear benefits to on-site fabrication processes, for example by providing the unskilled user with intuitive instructions, the application of off-site processes relating to early architectural design stages remains in its early development.

When looking at the benefits within the construction stages of a project, this democratization of expertise through easy and straightforward holographic guidance demonstrates large shifts from traditional drawing-based workflows. This change has challenged the role of craftsmen, while giving them new and integrated possibilities. AR architectural applications range from component-based assemblies in iBrick (Hahm, 2019), to fabrication processes for unskilled workers in Augmented Construction (Goepel, 2019) and a new fabrication workflow for users to adjust assemblies in BloomShell (Song, 2020). These each demonstrate new ways to help people fabricate complex assemblies with the support of AR holograms and provide insight on the potential of AR technology within the architecture field. Through this process, complex or non-uniform assemblies could be achieved by this ease, giving untrained designers greater design freedoms.

With a predominance in AR research focused in the field of architectural fabrication, the field of immersive design-oriented AR applications is gradually emerging. These build upon the characteristics of on-site AR applications yet begin providing a more inclusive and collaborative interface for professionals and non-professionals. The use of gestures gives users the chance to design through gestures and interact with models in real-time. Examples of these include Gravity Sketch (Home - Gravity Sketch, 2019) and Google Tilt Brush for real-time drawing tools in VR. Mobile AR applications typically rely on devices such as Microsoft HoloLens, Facebook's Oculus or Microsoft Kinect, which are special AR and VR equipment intended for users to experience different forms of augmented environments. Of the AR applications available for 3D sketching include 3DBrush (llya Rimchikov, 2017), which permits users to generate geometry based on the curve data in the AR environment, and Mobi3DSketch (Kwan & Fu, 2019), which introduces creative curve based 3D sketching in smartphones. These VR and AR applications are limited to additive or curve-based methods of sketching. The project seeks to challenge these limitations, giving users access to multiple geometry based workflows for architectural design applications. Three different methods of sketching are proposed including Freeform 3D Sketching, Additive 3D Sketching, Computational 3D Sketching, all with the use of ubiquitous handheld devices, a smartphone or a tablet.

To capitalize the potentials of these mobile AR applications, this project seeks to provide both architectural assembly tools with meshes and integrate different computational design programs. While programs such as Grasshopper or Unity provide several unique design and UI functions, their interoperability remains limited. By integrating Grasshopper via UDP permits several design methods, which help overcome the computing challenges of smartphone devices.

The inclusivity of this tool intends to support new forms of spatial learning and interactions with the surrounding city. Considering these AR's characteristics, it could help users learn and develop skills and knowledge more effectively than other technology environments (Hedberg et al., 2018). As seen through the popularity and engagement in AR games such as Pokemon GO, users can in this case design, reimagine urban spaces and share these visions with urban stakeholders. These potentials can be seen in similar light to Vetruvius's De architettura, which provided a rationale for developing several branches of knowledge into built form or scientific discipline. This treatise on architecture additionally gave emphasis to the skills of the artisan, which can be reappropriated within what we will experience as an 'Augmented Age' (Brett et al., 2016). The project's ambition is therefore to speculate different uses of this real-time concept design tool within the architectural discipline.

3 Methodology

VitruviAR is an AR design tool conceived to provide multiple early design stage workflows more accessible and inclusive. By capitalizing on the use of AR on handheld devices, this prototype provides real-time design feedback for three design methodologies based on the concept of sketching: freeform, additive, and computational 3D sketching. This section will outline the main steps adopted in the development of this prototype including application architecture, user interface, and 3D sketching methods. Additionally, because the development of the prototype based on the Android OS, toolset from Android NDK is used (Android NDK, n.d.). And API from Unity is mainly used for the AR functions in the application (Unity-Manual, n.d.).

3.1 Application Architecture

The application development began with the decision of an appropriate device for users. With accessibility as a primary focus, we used Galaxy S7 with Android 8.0.0 OS, one of the lowest versions of the OSes for AR application development. The front and back-end were then developed on Windows 10 based PC.

On the front-end, we used the gaming engine Unity with the plugin AR Foundation. This permitted an AR pipeline where images could be processed in real-time by common smartphones or tablets with user orientation and world tracking. The plugin ARCore XR was used to enable the camera detection of the plane in the physical world and select objects in the AR environment by a simple touch of the smartphone screen through its raycast function. The touch screen is used as an interface between the physical and digital environment for the application to register spatial coordinates.

On the back-end, data is first transferred wirelessly from the device to the PC via a User Datagram Protocol (UDP). This data is received by Unity and depending on the selected form of sketching, is then transferred to Grasshopper via custom C# scripts. The use of UDP permits faster streaming capacity compared to Transmission Control Protocols. This additionally enables low-latency internet connections, allowing for real-time feedback between Grasshopper, Unity and the device.

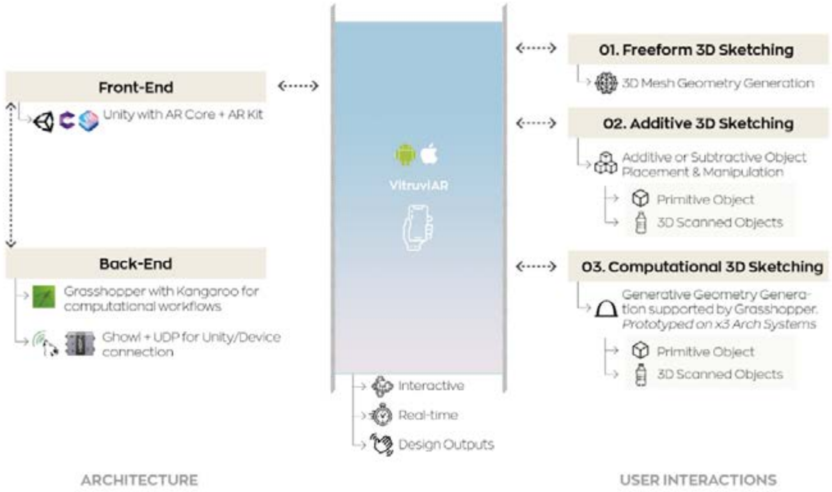

Figure 2. VitruviAR Architecture. Source: Author of this paper, 2021.

3.2 User Interface

The UI was designed in Unity with a focus on accessibility towards all types of users. A series of buttons, sliders, pulldown and text inputs are used to create an intuitive UI while providing a wide range of design possibilities for each of the three design methods. The lower set of icons focus on the sending and receiving of models to and from Unity and Grasshopper, together with the object manipulation sliders. These include the scaling, rotating, coloring, adding, and deleting of models, as well as the placement and adjustment of a base plane upon which all geometry can be built. Users can check the number of objects placed and the number of points used for generating a mesh from the text located at the left bottom of the screen.

The top inputs of the UI relate to the UDP connection essential for real-time interactions specifically for the Computational 3D Sketching design method. While all other design modalities can be done offline, the computational opportunities of Grasshopper and the computing power of a PC require one to be near the user. To implement this, users simply select the UDP address of their local PC and select the Port number or preset. Other options include the selection of what design modality they would like to design with, from mesh generation from points, predefined primitive objects such as spheres and cubes, or with user-uploaded 3D scanned objects. Finally, these can be placed above or near the base plane to point the mobile device and touch the screen to add or remove objects within physical space.

Figure 3. Vitruviar user interface. Source: Author of this paper, 2021.

Table 1. Vitruviar user interface description

Operation	Input (UI Type)	Description
1. UDP Address	Text (Input Field)	A keyboard appears after a single-finger tap on the field. IPv4 address should be typed in here, which PC and mobile phone are both using.
2. UDP Port Number	Touch (Dropdown)	Users can select the port number from 5001 to 5009. The port number will be applied instantly after selecting it.
3. Object	Touch (Dropdown)	Users can select different types of objects; mesh, box, cylinder, and scanned object. It also provides the selection of design scenarios. If users choose mesh, they can use the Freedom 3D Sketching method. However, if they choose other primitive objects or scanned object, users can do *Additive or Computational 3D Sketching methods.*
4. Number of Objects	N/A	Indicates the number of total objects calculated.

5. Color	Touch (Dropdown)	Different colors can be chosen by users. The color is only applied to objects and scanned objects, not to meshes from *Freedom 3D Sketching.*
6. Export	Touch (Button)	The location and geometric data of objects are sent from the mobile phone to Grasshopper via UDP communication. The data will be sent to each frame if the toggle is on.
7. Import	Touch (Button)	The scanned objects or modified objects should be downloaded in the local folder of the smartphone in advance. Users can view the downloaded objects from the "Object" operation after touching the "Import" button.
8. Show/Hide Planes	Touch (Toggle)	The base plane disappears when the toggle is on, and vice versa.
9. Delete selected object	Touch (Toggle)	If the toggle is on, the user can delete an object with a single-touch on the object.
10. Delete Everything	Touch (Button)	All objects will disappear with a single push of this button.
11. Scaling	Touch & Drag (Slider)	The user can change the scale of object by dragging the handle of the slider. The scale range is from 0.2 to 2.
12. Rotating	Touch & Drag (Slider)	The users can rotate the object from 0 to 360 degrees.

3.3 Design Methods

Three design methods were created to provide users with tools intended to facilitate early architecture design stages. While these are not intended for fabrication and assembly purposes, these do give users several design inputs to sketch freely in three dimensions.

3.3.1 Freeform 3D Sketching

This first method permits users to freely design and manipulate mesh-based geometry in space. This adopts a higher level of fabrication abstraction appropriate for conceptual 3D sketches of an idea. The process of making meshes has three steps. First, the user places points on the AR base surface. Then, the placed vertex points can be located using the coordinate data from the touch screen. If the point count is above three, a mesh is automatically generated, using these as mesh vertices. The users then can add, delete, and modify the spatial coordinate of each mesh vertex by clicking and dragging the point gumball on the handheld device *(Figure 4).*

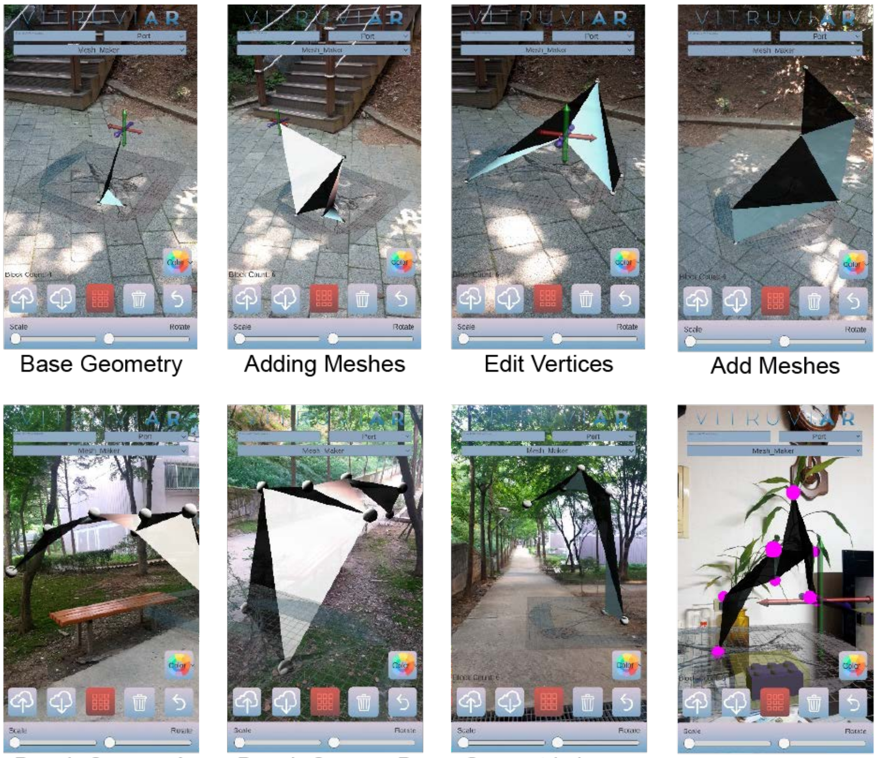

Figure 4. Images of Freedom 3D Sketching. Source: Author of this paper, 2021.

3.3.2 Additive 3D Sketching

The second method includes the use of both predefined primitive geometries, and user-uploaded 3D scans as building blocks for users to design architectural spaces within the AR environment. The UI gives users the possibility to manipulate objects including cubes, spheres, cylinders and digitally place these within their surrounding space. Geometry in this method remains unconstrained by heights or physics systems and can be stacked or placed irregularly *(Figure 5).*

For the second and third methods, users also can additionally import and manipulate pre-scanned 3D objects into the AR environment *(Figure 5)*. This gives users agency in defining their own geometries or building material manually or through Grasshopper as explained in the next step.

Additive Design A | Additive Design B | 3D Scanned Object in AR | Composite Design

Figure 5. Images of Additive 3D Sketching. Source: Author of this paper, 2021.

Figure 6. AR model in the real world. Source: Author of this paper, 2021.

3.3.3 Computational 3D Sketching

The last design method overcomes the limitations of remotely using handheld devices as seen in the previous 2 methods. By connecting the AR workflow with the computational capacities of Grasshopper, users can design parametric structures intuitively. To test this customizable workflow, we developed a system of self-generating arch structures that use the user-defined point data from the AR environment. These point systems can be manipulated by the user in a similar way as the first mesh editing method, and results can be refreshed via UDP.

In the case of these generated arch systems, the Grasshopper script subdivides each arch system into a series of points, permitting users to select an array of both the primitive objects or scanned 3D objects. Capitalizing on the tools of 3D object scanning, this tool permits users to upload and visualize scaled objects to scale and begin discussing structural, material, or fabrication properties *(Figure 7).*

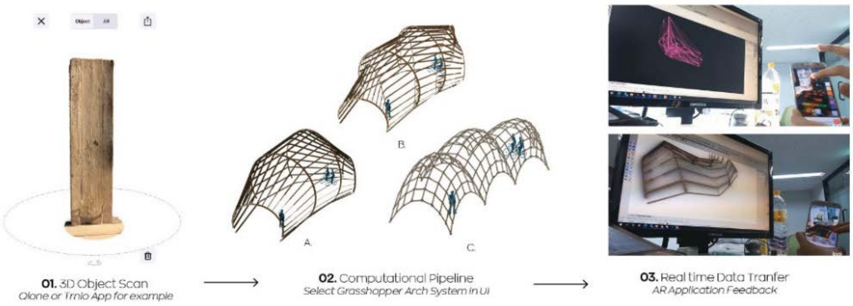

Figure 7. Computational Sketch Process. Source: Author of this paper, 2021.

4 Results and Discussions

VitruviAR is a tool that suggests the potentials of AR within early stages of the architectural design process. Ranging from freeform mesh generation and object manipulation, to integrated computational workflows, the possibilities within pre-fabrication stages are numerous. By developing an accessible UI, this prototype suggests a platform where experienced users can integrate their own computational workflows, and inexperienced users can use offline with pre-determined tools. Images of the results can be captured and shared between users and urban stakeholders to improve urban spaces, and 3D models can be exported for further development. Through an initial prototype testing at the Harvard Graduate School of Design, these three methods of sketching in AR demonstrate potential forms of engaging and interacting with the city, and new modes of education in learning and seeing spatially. This additionally offers the opportunity to design freely and evaluate solutions that are not necessarily optimized through the AR experience.

Two principal limitations relate to the real-time communication and accessibility of advanced workflows. In the case of the computational 3D sketching pipeline, the real-time communication is constrained to WIFI signal distances. Due to the implementation of the UDP method, it currently remains necessary for users to stay near an internet router at unobstructed distances less than 50m approximately, adjacent to the local PC and router. Further limitations in speed are found in the generation of parametric geometries while using grasshopper scripts. While these typically upload in real-time when users select the primitive objects, these can encounter lengthier upload speeds when using 3D-scanned objects. Transferring data, in this case with thousands of faces challenges with UDP, which we have successfully improved through a custom mesh count reduction C# script. The accessibility of this computational process, therefore, demonstrates a limitation for inexperienced users. Experienced designers can use their own processes, downsize 3D-scanned meshes or upload these directly to the application from their handheld device. Further user testing on these limitations for both users types will help improve this prototype, additionally providing insight on future integrated collaborative workflows.

5 Conclusions

VitruviAR demonstrates the potential of mobile AR in early design stages. It provides insights into new ways in which multiple users can sketch and ideate projects on and off-site. By expanding on the current development of digital fabrication AR workflows, this paper presents a consolidation of diverse architectural design methodologies appropriate for both specialists and inexperienced users. This establishes the potential for a continuous workflow in AR from initial design and visualization stages to fabrication and construction stages. Benefits include real-time design representation, different methods of 3D sketching, and improved accessibility through handheld devices.

This project provides an architectural design ideation platform in handheld devices with AR. It can be used to add new tools and freely imagine spaces that interact with the city. Therefore, the freedom provided through the three design methods becomes critical to the project, making structural or optimization processes secondary. Future work envisions the extension on both spectrums of the design process. On one side this suggests the creation of additional sketching tools as seen in VR applications such as pen drawing tools, while the other side encroaches on design development stages. By integrating further plugins from the Grasshopper ecosystem together with an improved UI, these could then be integrated with AR fabrication applications such as Fologram (Jahn, 2019), thus completing the design cycle in a native AR environment. Ultimately VitruviAR seeks to enable new conversations on architectural design ideation, through a mobile and accessible UI for experienced and non-experienced designers alike.

Acknowledgements. We would like to thank Jose Luis Garcia del Castillo Lopez, Area Head of the MDes Technology, for his continuous support and guidance at the Harvard Graduate School of Design.

References

Android NDK. (n.d.). Android Developers. Retrieved October 19, 2021, from https://developer.android.com/ndk/index.html

Goepel, G. (2019). *Augmented Construction*. Papers.cumincad.org; CUMINCAD. http://papers.cumincad.org/cgi-bin/works/paper/acadia19_430

Gravity Sketch | 3D design and modelling software. (n.d.). Gravity Sketch. Retrieved July 19, 2021, from https://www.gravitysketch.com

Hahm, S. (2019). *Augmented Craftsmanship*. Papers.cumincad.org; CUMINCAD. http://papers.cumincad.org/cgi-bin/works/paper/acadia19_448

Hedberg, H., Nouri, J., Hansen, P., & Rahmani, R. (2018). A Systematic Review of Learning Through Mobile Augmented Reality. *International Journal of Interactive Mobile Technologies (IJIM)*, *12*(3), 75. https://doi.org/10.3991/ijim.v12i3.8404

Jahn, G. (2019). Making in Mixed Reality Robotic Fabrication View project Design Futuring View project.

Kwan, K. C., & Fu, H. (2019). Mobi3DSketch. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. https://doi.org/10.1145/3290605.3300406

Kim, M. J. (2013). A framework for context immersion in mobile augmented reality. *Automation in Construction*, *33*, 79–85. https://doi.org/10.1016/j.autcon.2012.10.020

King, B., Lightman, A., Rangaswami, J. P., & Lark, A. (2016). *Augmented: life in the smart lane*. Marshall Cavendish Editions.

Rimchikov. (2017). 3DBrush — Augmented Reality Drawing. 3dbrush.com. http://3dbrush.com/

Song, Y. (2020). *BloomShell - Augmented Reality for the assembly and real-time modification of complex curved structure*. Papers.cumincad.org; CUMINCAD. http://papers.cumincad.org/cgi-bin/works/paper/ecaade2020_030

Song, Y., Koeck, R., & Luo, S. (2021). Review and analysis of augmented reality (AR) literature for digital fabrication in architecture. *Automation in Construction*, *128*, 103762. https://doi.org/10.1016/j.autcon.2021.103762

Szeliski, R. (2011). *Computer Vision: Algorithms and Applications* (1st ed. 2011. ed., Texts in Computer Science). London: Springer London : Imprint: Springer.

Technologies, U. (n.d.). *Unity - Manual: Unity User Manual (2019.3)*. Docs.unity3d.com. https://docs.unity3d.com/