

POLITECNICO DI TORINO

Master's degree thesis in Computer Engineering



A Mixed Reality application to support TV Studio Production

Supervisor

Andrea Sanna

Candidate

Federico Ucchesu

Company supervisors – RAI CRITS

Ing. Roberto Iacoviello

Ing. Davide Zappia

Academic Year 2018/2019

Ringrazio il professor Andrea Sanna, l'ing. Roberto Iacoviello e l'ing. Davide Zappia per il supporto costante e la disponibilità durante questi mesi.

Ringrazio la mia famiglia e i miei amici per l'affetto che non mi hanno mai fatto mancare.

Contents

Abstract	4
Abbreviations Table	7
Chapter 1 Introduction and context	8
1.1 TV Production Context	9
1.2 Current solutions for set creation	12
1.3 An innovative approach	15
Chapter 2 State of the Art	16
2.1 Virtual, Augmented and Mixed Reality	16
2.2 Augmented & Mixed Reality Devices and SDKs	19
2.2.1 Microsoft HoloLens	19
2.2.2 Meta 2	20
2.2.3 Magic Leap One	20
2.2.4 Epson Moverio BT-300	21
2.2.5 AR SDKs	22
2.3 Augmented & Mixed Reality Application Fields	25
2.3.1 Medicine and surgery	25
2.3.2 Teaching	26
2.3.3 Cultural heritage	26
2.3.4 Tourism	27
2.3.5 Assembly and maintenance	28
2.3.6 Entertainment	30
2.3.7 Military applications	31
2.3.8 Architecture and Design	31
2.4 Previsualization techniques for TV and Cinema preproduction	33
2.4.1 Storyboarding	33
2.4.2 Animatics & CG Renderings	34
2.4.3 Real-time compositing	35
2.5 AR for Television	36

Chapter 3	<i>Technologies</i>	39
3.1	Microsoft HoloLens	39
3.1.1	Hardware: Sensors, Optics, Processors and Memory	40
3.1.2	Basic Usage	42
3.1.3	Developments: HoloLens 2	43
3.2	3D Scan and Modeling	45
3.3	Unity 3D	47
3.4	HoloToolkit	50
3.4.1	Premise	50
3.4.2	Introduction	50
3.4.3	Input: Gaze, Gestures, Voice	51
3.4.4	Spatial Mapping & Spatial Processing	53
3.4.5	Sharing	55
3.4.6	UX (User Experience)	58
Chapter 4	<i>A HoloLens Application to Support Television Set Design</i>	59
4.1	Project Requirements	59
4.2	System Architecture	61
4.2.1	Contents creation workflow	61
4.2.2	Application Manager and Interface	62
4.2.3	Environmental Awareness System	64
4.2.4	Placing and Editing System	67
4.2.5	Sharing System	69
4.2.6	Scene Capture	72
Chapter 5	<i>Test and Results</i>	73
5.1	Test	73
5.2	Results	74
5.3	Future developments	76
Conclusion		78
References		79
List of Figures		82

Abstract

New technologies, such as the Microsoft HoloLens, enables the user to visualize and interact with data in new ways. This thesis explores Augmented and Mixed Reality and presents their potentiality.

In Augmented Reality (AR), users see and interact with the real world while digital content is added to it. The real environment is overlaid with virtual objects, labels, various kinds of contents that can enhance the real world.

Mixed reality (MR) is the blending of real world with the virtual one. Virtual objects are not just overlaid on the real world, but they are interactive. In this sense, it can be seen as an evolution of the augmented reality. A user remains in the real-world environment while digital content is added to it and he can interact with virtual contents.

In these days AR and MR can be used in many application fields, from medicine to entertainment, architecture or teaching. Several examples of application field will be proposed, focusing on the applications in the field of cinema and television production.

The preparation of a TV set is a complex work: in the phase of previsualization some sketches or CG renderings of the design are created, with many alternatives and with views from different angles; during the actual set creation the crew manually place the objects in the environment but often the director does not like this first configuration and the crew has to move again the objects in the set.

Mixed Reality can be used in TV Studio Production to support scenographers and set directors in a collaborative working platform.

The project of this thesis has been realized in collaboration with RAI-CRITS (Centro Ricerche, Innovazione Tecnologica e Sperimentazione), for a real use case, Rob-O-Cod, a tv show broadcasted on Rai Gulp.

The aim of the thesis is been the creation of a Microsoft HoloLens application with a novel visual user interface. Using the available input methods on the HoloLens, a set designer and a set director can interact and collaborate to design a set with free hand gestures. Using this application, a set designer can feel himself immersed in a virtual studio, he can move in the environment and position, move, rotate and scale virtual objects. In this way, the previsualization is not mediated from a pc display, a user can set up the studio and immediately see the scene from different angles.

The main technologies used in the development of this work are Microsoft HoloLens HMD, the Unity game engine with the HoloToolkit API, that allows the developing of holographic applications, a 3D scanner and Blender as a CG modeling software. All these technologies had been analyzed in detail, especially the HoloToolkit, whose components had been used and combined to meet the application requirements, from the multimodal input system, to the sharing system, to the environmental awareness, to the spatial processing, to the editing UI.

The whole experience has been developed to be accessible by gestures or by voice, with a simple menu that explains users what they can do.

After a phase of connection in case of collaborative experience, users can scan the environment, seeing the creation of an environmental mesh.

Then the mesh is processed to find planes that the user can select to use them as working tables, where he can put virtual objects, moving, rotating and scaling them as he wants to create a set configuration. Finally, it's possible to take a picture of the set, with a compositing of the real and the virtual environment.

The application has been tested by a group of testers, composed by about 20 people, that has been asked to create an augmented reality set and to complete a standard SUS questionnaire to evaluate the usability. The results of the questionnaires show that the application has been appreciated by users in terms of usability, but some limitations of the application, mainly caused by the hardware specifications, were found evident.

The application will be merged with the one developed parallelly within the same project, with the features of loading 3D models and textures from a remote DB, and with the search system and the interface to find models through tags and properties of 3D models. Furthermore, in future, the project will be ported to different devices with improved features and performances, probably to the new version of HoloLens.

The thesis is divided in five chapters. In Chapter 1 the motivations for this project are presented. In Chapter 2 there is an explanation of what Augmented and Mixed Reality are and how they work. Furthermore, a survey of the state of the art of AR and MR is presented, with emphasis on the applications in the fields of cinema and television production. In Chapter 3 the technologies used in the project are described, and Chapter 4 presents the developed application, from the project's requirement to the system's architecture, describing the main features. Chapter 5 present the tests done and the results. Finally, conclusions and future possible developments are illustrated.

Abbreviations Table

API	Application Programming Interface
AR	Augmented Reality
AV	Augmented Virtuality
CG	Computer Graphics
FOV	Field of View
HMD	Head Mounted Display
HTK	HoloToolkit
MR	Mixed Reality
MRTK	Mixed Reality Toolkit
SDK	Software Development Kit
SUS	System Usability Scale
UI	User Interface
UX	User Experience

Chapter 1 Introduction and context

As technology has advanced, so the way of visualizing simulations and information is changed in the digital era.

Virtual Reality (*VR*) and Augmented Reality (*AR*) are great examples of such visualization methods which are booming in this period, either by being immersed in a simulated virtual environment or adding a new dimension of interaction between digital devices and the real world.

Both methods have something similar, equally significant in their ways providing experiences and interaction being detached or blending with the real world, making real and virtual alike. The process of replacing and supplementing the real world according to the needs is what makes these methods more desirable and increasingly popular.

From consumer application to manufacturers these technologies are used in different sectors providing huge benefits through several applications, even if has to be said that they are still immature technologies, in a state of evolution.

The purpose of the next chapter is to examine better some AR technologies and possible applications in many fields.

The project of this thesis has been developed in the context of a TV studio production, in collaboration with RAI-CRITS (Centro Ricerche, Innovazione Tecnologica e Sperimentazione), in the same context as the previous work "Augmented Reality to Assist Television Production" [1].

1.1 TV Production Context

A producer's brilliant idea could be certainly the first step toward the realization of a successful television show or film. However, the only idea is not enough. After being found, it has to be developed in a written or viewable form, to be analyzed, modified, reworked: a good intuition is not enough if it is not supported by an articulated and exhaustive project.

Television is a complex language that uses many expressive registers at the same time. Developing the project is one of the most complex phases: it is at this moment, in fact, that we move from the idea to the program, and it is through this process that we try to get as close as possible to the original idea.

The creation of media content for film or television is a creative process, something similar to art, like design or photography, but it is also a technical process.

Actually, cinema and television studios use lots of technical instruments (cameras, lights, special effects, etc.). Since it is a very composite work, it is normally divided in three steps:

- ◇ pre-production,
- ◇ production,
- ◇ post-production.

Some aspects of the pre-production will be discussed more in detail.

As explained in "Television Production Handbook" [2], preproduction includes all the preparations and activities before the troupe actually move into the studio or the field on the first day of production.

It involves idea generation, research, scripting, discussions with all the crew members and actors, arranging equipment, video/audio tapes, properties, costumes, sets designing and creation or location hunting and booking of editing shifts. Pre-production activities involve also the planning, budgeting, and preparation needed before shooting begins. In all it involves planning

everything in advance. This is very essential to get desired results. The pre-production period can last as long as a month or more for a movie, or just a week for a single episode of a situation comedy.

During the more creative part of this process, all the activities necessary to transform the basic idea into a workable concept take place. It is in this part that are used all those techniques that may be linked to the process of previsualization.

Previsualization (also known as “previs” or “previz”) is the process in pre-production of visually mapping out scenes in a movie, before principal photography, or the creation of studio design previews, before the actual setting up of the studio. It can utilize a combination of photography, storyboards or simple sketches, CG renderings and animatics.

The renderings can range from simple stick figures and modest drawings to very detailed, artistic illustrations and renderings that border on fine art. Figure 1 shows an example of a sketch of the set design of a RAI TV show, made by a set designer in the early stage of the production.

Previz modern techniques are based on computer generated images for visualizing scenes, camera angles, camera blockings, lighting conditions, and other situations and conditions before the actual shoot. Some of these techniques will be discussed better in what follows.

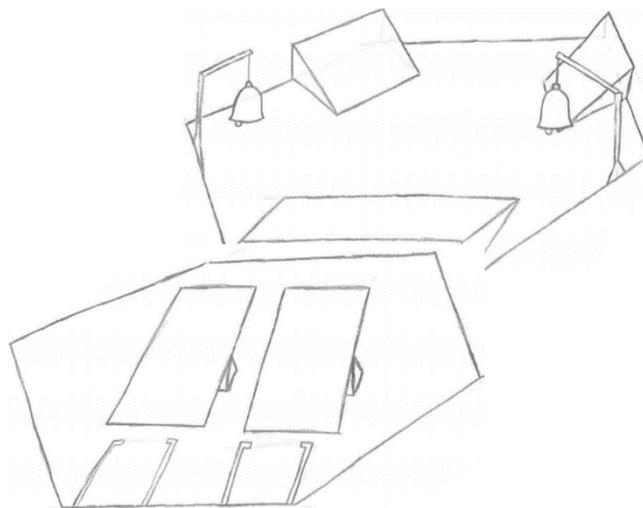


Figure 1 Sketch of a RAI TV set

The main advantage of previsualization is that it allows set director, scenographers, or other people involved in the TV Production, to experiment different options, such as lighting, camera placement and movement, camera angles, without having to incur the high costs of actual production.

In a TV production context, a very important and complex phase is the creation of the set. Sets can be at first divided in two categories: indoor and outdoor sets. During the development of this thesis work, the main context has been an indoor set.

1.2 Current solutions for set creation

An indoor set can be made in a real place, suitable for the scene you want to shoot, or in a studio, built and set up for this purpose. In both cases it is an expensive job in terms of resources and time.

Usually the set designer, perhaps in collaboration with an artistic director or another support figure, explains how the set should be set up, and the crew physically positions the objects in the environment. Unfortunately, various configurations have to be tried because the set usually has to be taken up by more points and angles of shooting, and the scenographer needs to try these configurations.

Relatively light and small objects may not represent a problem, but coping with large and heavy objects (e.g., a wardrobe or a piano) makes these operations more difficult and expensive.

Alongside the "traditional approach", there is the so-called Virtual Set, or Virtual Studio (VS). With the development of Computer Graphics (CG), producers soon realized the advantages of use modern technologies. A virtual studio is a television studio that allows the real-time combination of



Figure 2 A television set before/after the mixing.

people or other real objects and computer-generated environments and objects in a seamless manner.

The basic technology used is the green-screen, also known as Chroma-key, with which one can mix different images together. A scene is shot with the persons against a green background. The green color is afterwards removed from the background and replaced with a new image. Figure 2 shows this situation.

In a virtual studio, the real camera moves in the set environment, while the image of a virtual camera is rendered in real-time from the same perspective. The virtual camera adapts at any time its settings such as zoom, angle, traveling, etc.

This is the difference between a virtual studio and the “simple” technique of Chroma keying.

So, a proper VS should have at least these properties: people and real objects have to be integrated in the virtual set, 3D models must be rendered in real time during the shooting and there must be coherence between real and virtual, also when cameras are moving.

By using virtual studios, the problems of the traditional approach can be avoided, but the director works in a fictitious environment and it is not possible to walk through the environment before the shooting. As the consequence, the director can only imagine how the scene will appear on the screen.

With modern systems, such as “Viz Virtual Studio” solution [3], a set of software and technologies to create virtual studios, it’s possible to combine the physical and virtual objects and to display in real-time the results in a display.

This is a very good compromise, but it is clear that having full access to the complete set during the pre-visualization phase could be much more effective. The director can really see the entire set and he can have a good idea of what will appear on the screen. Unfortunately, this interaction is still mediated by the screen device, he cannot move for real around the 3D

environment, he is not really “there”, he can only see what is displayed on the device screen, and this could be limiting.

1.3 An innovative approach

Thanks to the philosophy of simple, instinctual interactions and the comfort of use of actual available augmented reality devices, it's possible to think to use this type of devices in the phase of previsualization.

In particular, modern smart glasses, such as HoloLens, Magic Leap or Meta 2, could be used as tool by which set up and visualize a studio in a tv pre-production context.

In the continuation of this thesis, all the aspects relative to AR will be analyzed, listing main technologies and application.

Then, the HoloLens application developed during this thesis work will be presented, including a description of the technologies used. This application aims to be a tool for set designers and artistic directors, used in TV Studio pre-production to support set creation in a collaborative working platform.

As will be seen, using this application a set designer can feel himself immersed in the virtual studio, he can move in the 3D environment and position virtual objects. In this way, the previsualization is not mediated from a pc display, the set director can set up the studio and immediately see the scene from different angles.

Furthermore, the application allows the collaboration of more people involved in the studio creation. Anyone wearing a HoloLens will participate, seeing what has been done by the others or giving its contribution.

The application has been developed for a real use case, a RAI TV show that will be better presented in the following, and the requirements has been identified thanks to meetings with RAI scenographers and set designers.

Chapter 2 State of the Art

This chapter starts with an introduction to augmented reality, and it explains the difference between similar concepts that are often associated with. The devices and the technologies currently available in this field will be presented. Augmented reality has been used in different ways in many fields; some example is presented in the following.

Then some technique of previsualization currently used in preproduction in a TV or Cinema context is presented. These techniques can be simple as well as technologically advanced.

Finally, it's possible to discover how some augmented reality technologies can be useful to develop a modern previsualization system for TV preproduction.

2.1 Virtual, Augmented and Mixed Reality

"Virtual" has had the meaning of "being something in essence or effect, though not actually or in fact" since the mid-1400s. The term "virtual" has been used in the computer sense of "not physically existing but made to appear by software" since 1959. [4]

Today Virtual Reality (*VR*) is commonly linked to all the technologies that immerse users in a completely virtual environment that is generated by a computer. The most advanced VR experiences even provide freedom of movement. Users can move in a digital environment and hear sounds. Moreover, special hand controllers can be used to enhance VR experiences.

In Augmented Reality (*AR*), users see and interact with the real world while digital content is added to it. The real environment is overlaid with virtual objects, labels, various kinds of contents that can enhance the real world for reasons of leisure or in more serious contexts.

The simplest example of augmented reality is offered by many smartphone applications, like Pokémon Go or some museum application.

The term Mixed Reality (*MR*) was originally introduced in 1994 in a paper by Paul Milgram and Fumio Kishino, "A Taxonomy of Mixed Reality Visual Displays" [5]. Their paper introduced the concept of the "virtuality continuum" outlined in Figure 3, and focused on how the categorization of displays. According to this taxonomy paper, "the most straightforward way to view a Mixed Reality environment, therefore, is one in which real world and virtual world objects are presented together within a single display, that is, anywhere between the extrema of the virtuality continuum", where the virtuality continuum extends from the completely real through to the completely virtual environment.



Figure 3 Virtuality Continuum

Following this scheme, two notable categories of mixed reality are identifiable: Augmented Virtuality (AV), when the virtual elements are dominant and are supplemented by elements from the real world, and Augmented Reality (AR), when the opposite is true.

From this original definition, the term Mixed Reality was reintroduced by Microsoft in 2015 with the Windows Mixed Reality platform, based on Windows 10. The platform was released together with his main device, Microsoft HoloLens.

In this context, to give an explanation can be said that *Mixed reality (MR)* starts with the real world, and virtual objects are not just overlaid on the real

world but can interact with it. In this sense, it can be seen as an evolution of the augmented reality.

A user remains in the real-world environment while digital content is added to it; moreover, a user can interact with virtual objects.

According to Microsoft given explanation [6], “Mixed reality is the result of blending the physical world with the digital world. Mixed reality is the next evolution in human, computer, and environment interaction and unlocks possibilities that before now were restricted to our imaginations. It is made possible by advancements in computer vision, graphical processing power, display technology, and input systems”.

2.2 Augmented & Mixed Reality Devices and SDKs

Nowadays augmented and mixed reality devices are very popular and there is a big expectation about their capabilities and features.

Everyone can easily try the experience of augmented reality thanks to the availability of tens of AR applications for smartphones. By now also the head mounted displays (HMDs) of virtual and mixed reality are available to anyone since they are often used in museums, exhibits or in a variety of events to propose that type of experience to the visitors.

The performances of these new technologies are growing day by day, and devices features are more and more similar to those visible in science fiction films in the last decades, nevertheless is important to know that augmented and mixed reality are still in their infancy, indeed developers and companies are working hard on it and there are updates every week.

In this section will be introduced some of the actual available head mounted displays, their platforms, and other AR SDKs.

The description of virtual reality devices like Oculus Rift, HTC Vive or Virtuix Omni will be omitted, because they are not strictly linked to the purposes of this thesis. Instead, the focus will be to present the devices and the technologies currently available for mixed reality.

2.2.1 Microsoft HoloLens

Microsoft HoloLens [7] is the device that is been used in the develop of this project, so it will be deeply examined in the next chapter, but it is one of the best all-in-one mixed reality devices. It is fully standalone, it has some depth sensor to interact with the real environment and good see-through lenses. As will be seen later, there is no a separate SDK for Windows Mixed Reality development; it uses Visual Studio with the Windows 10 SDK. Like all the devices of this type, it supports the development with game engines like Unity or Unreal Engine.

2.2.2 Meta 2

Meta 2, the successor of Meta 1, is an augmented reality device, shown in Figure 4, developed by Meta company. This device has see-through lenses and a 3D camera that allow it to recognize easily hand gestures, and a hardware system that offers a field of view of 90°.

Meta 2 seems to be one of the best AR devices, because of his big field of view and his resolution, but it is tethered, so it needs a pc to work correctly. Currently Meta 2 is no longer for sale, but the SDK is already available.

In 2019 the original company declare itself insolvent and then a new company called Meta View announced it acquired the intellectual property assets of Meta.



Figure 4 Meta 2

2.2.3 Magic Leap One

In some ways, the Magic Leap One is a bridge between the HoloLens and the Meta 2, released from Magic Leap in 2018.

The headset is tethered, though not in the way the Meta 2 is. A cable connects the HMD, called Lightwear, to a small hip-mounted computer, called Lightpack, that handles the primary data and graphics processing. The hardware includes also a Control with touch, haptic feedback and

spatial tracking with six degrees of freedom. All the components are shown in Figure 5.

Magic Leap One is defined as a head-mounted virtual retinal display, which superimposes 3D computer-generated imagery over real world objects, by "projecting a digital light field into the user's eye" [8].

Magic Leap software includes a custom operating system, called Lumin OS, derived from open source components such as Linux and AOSP. It is designed specifically for the unique requirements of spatial computing.

His SDK, Lumin SDK, is supported and included in game engines like Unity and Unreal Engine.

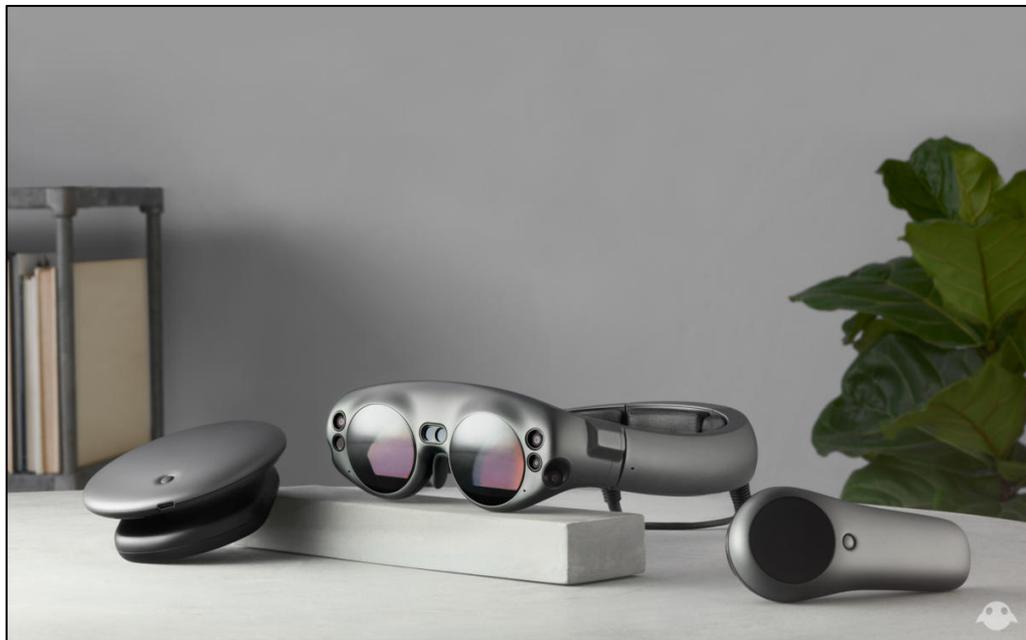


Figure 5 Magic Leap One

2.2.4 Epson Moverio BT-300

The Epson Moverio BT-300 (Figure 6) spots a transparent display that projects images on an 80 inches floating screen 5 meters in front of user's eyes. There is a 5 Megapixel camera located at the front of the glasses along with the motion and ambient light sensors. For the display it's used the Si-OLED technology. With a resolution of 1280x720, the Si-OLED

display provides sharp, bright and vibrant images. The display, when turned off, are totally invisible as it merges seamlessly to the glasses.

The Moverio BT-300 is powered by Intel Atom x5 processor and runs on Android 5.1 making it easier for developers to work on AR apps. It is connected to an Android controller that has a trackpad and basic Android buttons through a wire. This provides a familiar interface to existing Android user although an improvement would be welcome in the form of wireless or swipe/gesture control.

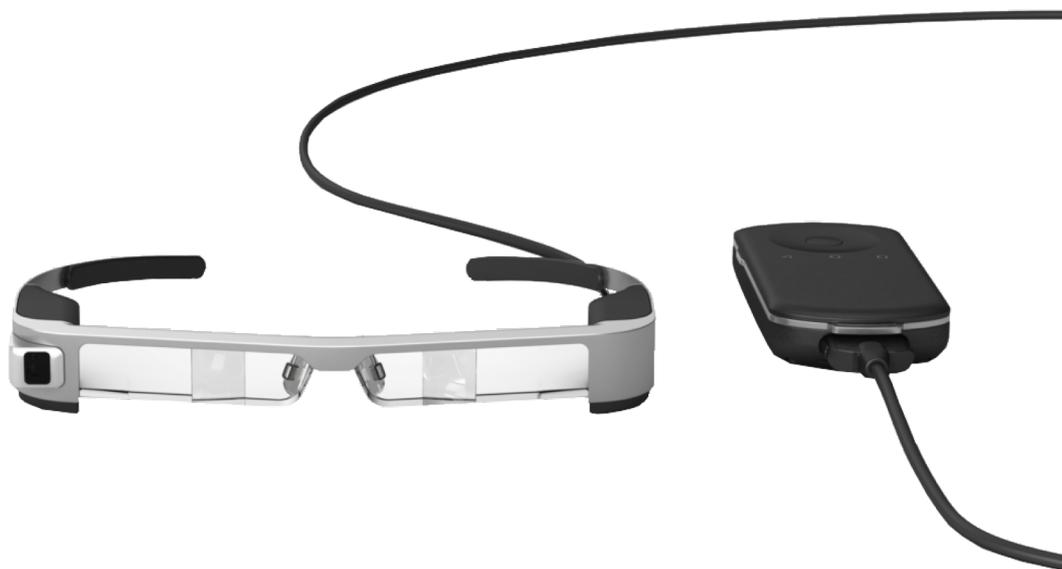


Figure 6 Epson Moverio BT-300

2.2.5 AR SDKs

Vuforia

Vuforia Engine [9] and SDK is one of the most popular system that help developers to build AR mobile applications for Android and iOS, and UWP Apps for Windows devices as HoloLens or PCs. Apps can be built with Unity, Android Studio, Xcode, and Visual Studio.

Its main features are the following:

- ◇ Object Recognition: it can recognize and track a broad set of 3D objects. The physical objects should be opaque, rigid and contain few moving parts. The objects should preferably be viewed indoors under diffuse and evenly distributed lighting [10];

- ◇ Image Targeting: any image with sufficient detail, including magazines, advertisements, and product packaging can be recognized and tracked. Instead of using the traditional fiducial markers or QR codes, which require special coded regions in the image, the SDK is able to detect and track the natural features found in the image by comparison with the features previously stored in a database. Data in the database can be created starting from JPG or PNG images in RGB or grayscale. Once the Image Target is detected, Vuforia Engine will track the image as long as it is at least partially in the camera's field of view.
- ◇ Cylinder Targeting: it is the to capability to detect and track images rolled into cylindrical and conical shapes. Cylinder targets can be used to recognize and track product packaging that are cylindrical or conical in shape. Soda cans, coffee cups, mugs and beverage bottles are good examples.
- ◇ Extended Tracking: it is a capability that delivers a continuous visual experience even when the target is no longer in the field of view of the camera, so that users have the freedom to move over large areas or to point the device away from the target. When recognizing a target, Vuforia builds also a map of the surrounding environment, assuming it to be mostly static: when the target goes out of view, Vuforia then uses these data to keep on tracking;

EasyAR

EasyAR [11] is another easy to use AR library, it is an alternative of Vuforia. The free version does not include all the features, but it offers planar image recognition and tracking, and surface tracking with the SLAM technology. With the paid licence it is available also the recognition of 3D object tracking.

ARCore

ARCore [12] is the native Google's platform for building augmented reality experiences. Using different APIs, it allows devices to integrate virtual content with the real world. Its main features are:

- ◇ Motion tracking: it allows a smartphone to understand and track its position relative to the world;
- ◇ Environmental understanding: it allows a smartphone to detect the size and location of all type of surfaces: horizontal, vertical and angled surfaces like the ground, a coffee table or walls;
- ◇ Light estimation: it allows a smartphone to estimate the environment's current lighting conditions.

ARKit

ARKit [13] is the API offered by Apple in its native development platform, so it is compatible only with iOS devices and with the app development in Xcode.

It combines device motion tracking, camera scene capture, scene processing, using either the back camera or front camera of an iOS device. The last version released, ARKit 3, includes a feature called "People Occlusion" by which the devices know where people and AR objects are and properly occludes the scene. With Motion Capture, it tracks human movement as input to the AR scene.

2.3 Augmented & Mixed Reality Application Fields

Augmented and mixed reality applications have been developed to support many fields, like medicine, teaching, workers training, media and entertainment.

2.3.1 Medicine and surgery

How is explained in a paper published from Current Medical Science [14], Mixed Reality has attracted great attention of clinicians in recent years. MR will bring about revolutionary changes in medical education training, medical research, medical communication, and clinical treatment.

In student teaching, MR technology helps students to more intuitively understand the complex anatomy of the human body. In a similar way, MR headsets can be used in the preoperative communication between doctors and patients or in a discussion between doctors for the formulation of an operation plan. In all these situations it is possible to improve the clarity of speech through the vision and manipulation of virtual 3D anatomical models. HoloLens applications offers a feature of shared experience, as in the example shown in Figure 7, that fits perfectly in this type of application.



Figure 7 Application of MR technique in preoperative discussion between doctors

2.3.2 Teaching

An example of the power of Augmented Reality for childhood education is the Quiver Education App [15], developed by Quiver Vision. This is an AR coloring experience available on iOS and Android mobile devices, with the focus on educational topics. Using the application, users can point their mobile device to a coloring book page and see an animated 3D virtual scene come to life. However, unlike other AR applications, with this app users can color the pages and see the virtual objects textured with their own colors. Any color that they add to the coloring page will be textured mapped onto the 3D model and seen in the AR scene. Some of the models also have simple interactivity.

2.3.3 Cultural heritage

There are many possibilities for the use of AR in museums. The most straightforward way is to use it to provide an augmented guide. This type of application can give supplemental information about an exhibit or the museum itself. This means visitors will get more information when they view exhibitions using AR.

Museums could even use it to display digital versions of artists next to their work. Now some museums are experimenting with AR apps inside the gallery spaces. For museum visitors, AR apps on their mobile devices will be very easy to use.

In other museums certain rooms have been arranged as a VR or AR spaces, where people can wear headsets (VR or MR) and try an application with 3D contents.

The National Museum of Singapore is currently running an immersive installation called Story of the Forest [16]. It is an immersive installation that transforms 69 drawings from the William Farquhar Collection of Natural History Drawings into three-dimensional animations that visitors can interact with. Visitors download an app and can then use the camera on their phone or tablet to explore the paintings. In a similar way the Smithsonian Museum,

in Washington D.C., offers to their visitors an AR application called “Skins and Bones” [17] that is a great example of using AR in a museum. With this application in his smartphone, a user can scan some skeleton to see how skin and muscle would have looked over the bones, and how the animals would have moved.

Petersen Automotive Museum, in Los Angeles, offers in his exhibits immersive HoloLens experiences. In the exhibit called “Hollywood Dream Machines” [18], visitors can live an experience of mixed reality with the famous DeLorean time machine car from the “Back To The Future” film, as well as the Warthog war vehicle from the “Halo” video game universe.

2.3.4 Tourism

An MR application for HoloLens presented in [19] has the main aim to provide information about historical buildings, supporting cultural outdoor tourism. The system can recognize known monuments and show user some multimedia information.

It has been tested in the city of Turin, as it shown in Figure 8.



Figure 8 MR application for outdoor navigation

In 2019 Google Maps released a new AR feature, that places digital directions on top of the real world to show where to walk (Figure 9). This feature could help tourists in the navigation in the cities they visit, in an immediate way.

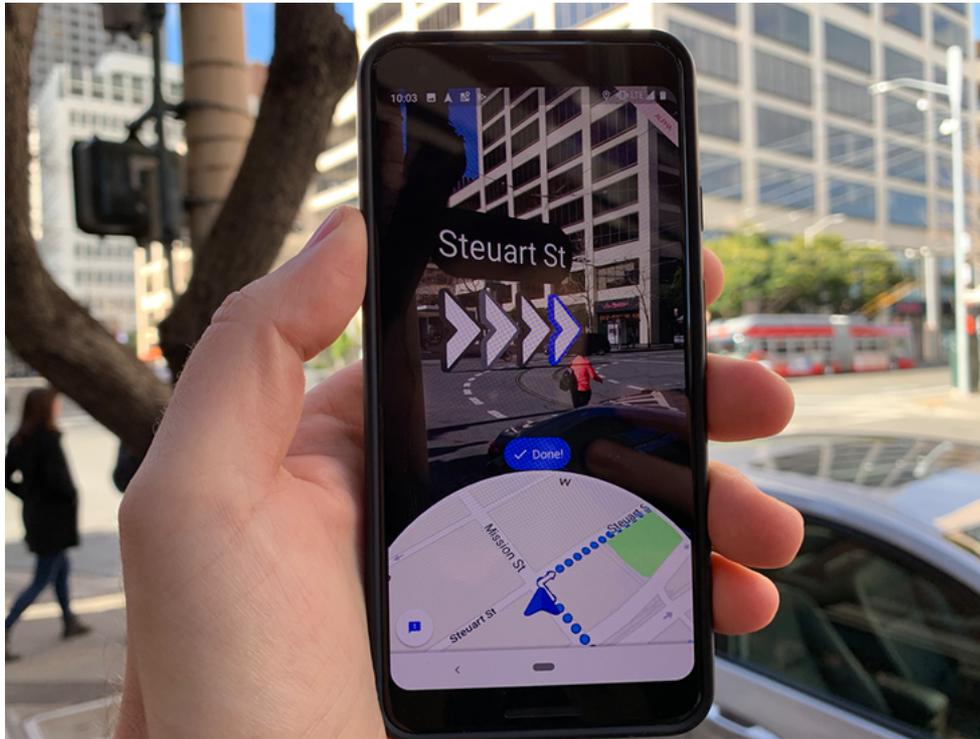


Figure 9 Google Maps AR

2.3.5 Assembly and maintenance

In the industrial field, and especially in the processes of production, assembly, maintenance, repair and operators training, AR can have an important application. These operations are very important, and operator are often required to learn complicated procedures.

It is not ever simple to memorize these procedures and AR provides a much more effective way of presenting instructions than traditional manuals by overlapping the procedures in the field of view of the worker through HMDs. Beyond providing faster learning, one also expects a reduction of errors due to the continuous switch of attention between device and manual.

A qualitative survey showed that mechanics found the augmented reality condition intuitive and satisfying for the tested sequence of tasks. [20]

Another advantage of AR in this field is the easy implementation of telepresence systems: in situations where the AR aid is not enough, a remote expert can be called for support.

ARMeisterbock [21] is a project in development whose main purpose is to design and implement an innovative training system and in-situ aid for operators in charge of controlling the dimensional quality of automotive components through the use of a template, or precision caliper, or, in its German name, a Meisterbock.

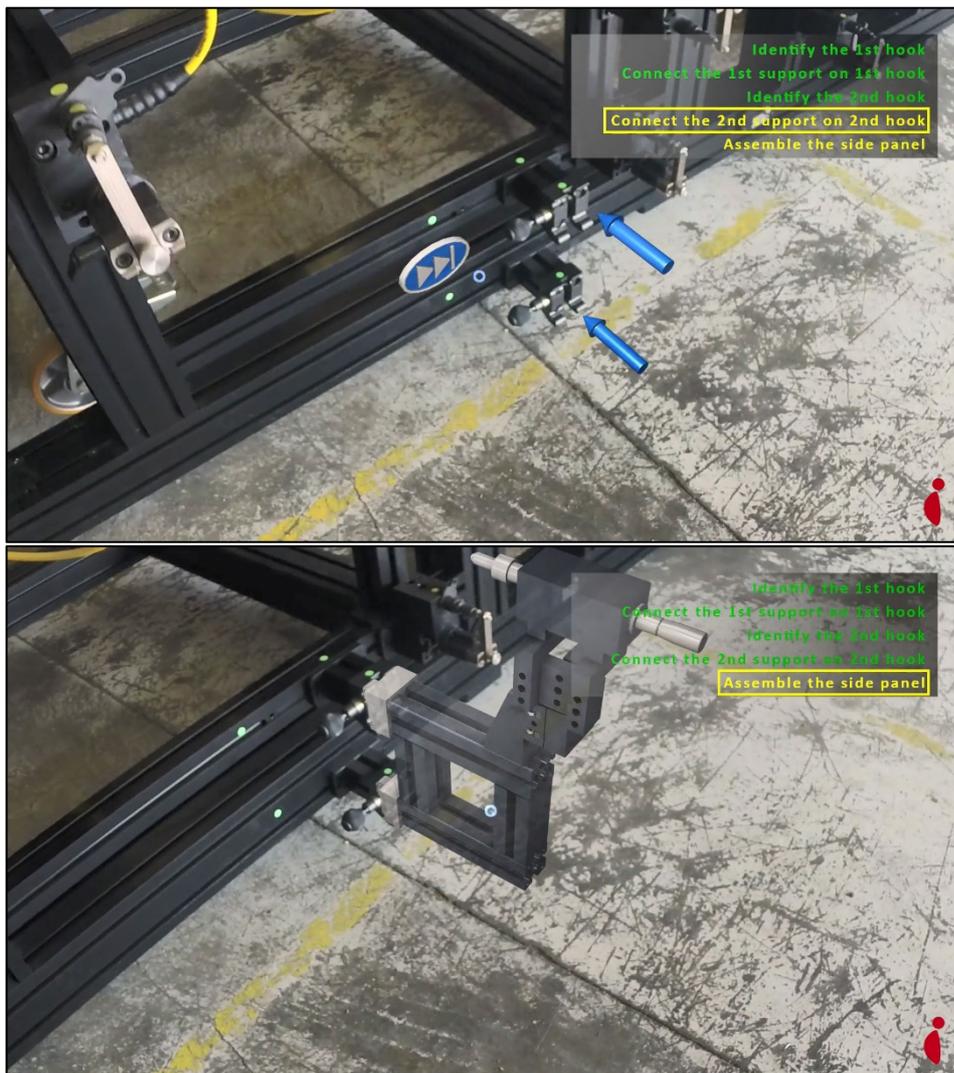


Figure 10 ARMeisterbock

This solution consists of an augmented reality system supported by an HMD (HoloLens in this case) which, thanks an AR application, will allow workers in industrial sites to:

- ◇ learning the peculiarities of the procedure during the training;
- ◇ practice operations and memorize the Meisterbock's set-up sequence;
- ◇ reduce the likelihood of error during the actual process thanks to continuous reminders and assistance.

Figure 10 shows some steps of a guided configuration with *ARMeisterbock*.

2.3.6 Entertainment

In the field of entertainment, AR had a of great impact. In AR mobile games, virtual information is superimposed on to the real world through smartphones displays. The most popular example of this type of application is Pokémon Go [22], shown in Figure 11, a mobile location-based social game involving the physical activity of gamer in the real world with potential and documented health benefit.

Unlike VR gaming which requires a separate room or confined area to create an immersive environment, AR gaming expands the playing field,



Figure 11 Pokémon Go

making the game more interesting taking advantage of the diversity in the real world.

2.3.7 Military applications

Military applications of AR include devices for different service and force characteristics. One of the earliest examples is the Battlefield Augmented Reality System (BARS) developed in the U.S. Naval Research Laboratory, a decade-long research program that whose aim was to prototype some of the early wearable systems for dismounted soldier operating in urban environment for situation awareness and training. [23]

In this field, AR can bring innovation also in the teaching.

A recent project, at a military academy in Taiwan, investigates the introduction of a tactical teaching system based on augmented reality. It tries to use interactional flow and visual aesthetics in order to produce more effective learning and heightened learning interest among commanders and staff trainees under low cost and high efficiency. [24]

2.3.8 Architecture and Design

Traditionally architects and designers used a range of different tools to show their clients what their works (buildings, furniture, etc.) will be like, including 2D plans, physical models, 3D renderings, interactive walkthroughs and animated fly throughs. However, there are several shortcomings with these methods. For example, architect's 2D plans can show in great detail the layout of a building, but sometimes the client has trouble imagining what the building would look like from the 2D drawing. Physical models, 3D renderings and interactive fly-throughs help the client understand what the final products looks like, but they are a reduced scale version of the final space.

It will be better to convey a clear understanding of what works will look like full-size on the final location.

Mobile AR can be used to view full-sized 3D virtual models on the local where they will be built or posed. This allows clients to clearly understand what it will look like. The application could also be used as a planning tool, placing several versions of the same work on site and allowing the client to provide feedback about the different design options.

An example could be SketchUp Viewer [25], a viewer with AR capabilities of SketchUp, a very popular software used for a wide range of 3D modeling projects like architectural, interior design, landscape architecture, and video game design.

Kubity Go [26] is another mobile app that allow to display SketchUp and Revit models in AR.

Better results can be obtained using MR devices, such as HoloLens. An article published in Autodesk University [27] presents case studies of mixed reality, using Microsoft HoloLens, applied for different phases of architectural project. By projecting the Building Information Modeling (BIM) model directly over the physical site in MR, architects can communicate design ideas to the team and clients in an immersive and interactive way.

2.4 Previsualization techniques for TV and Cinema preproduction

Previsualization (also known as previs or previz), as anticipated in the introduction, is the process in pre-production of define scenes in a movie, or, in a TV context, is the visualization of a preview of the set before or during its realization, through something written or visual.

2.4.1 Storyboarding

Previz has existed since the start of filmmaking, with simple handmade storyboards and replica models being used to plan a shoot.

The storyboard is essentially a series of frames, with drawings of scenes that should appear in the movie or in a TV show, much like a comic book of the film and it is used in pre-production to visualize the ideas of authors or directors. An example of a sketch of a storyboard of a TV show is shown in Figure 12. This tool can be of great help to communicate the visual imagined by the director to the crew as well as to the actors.

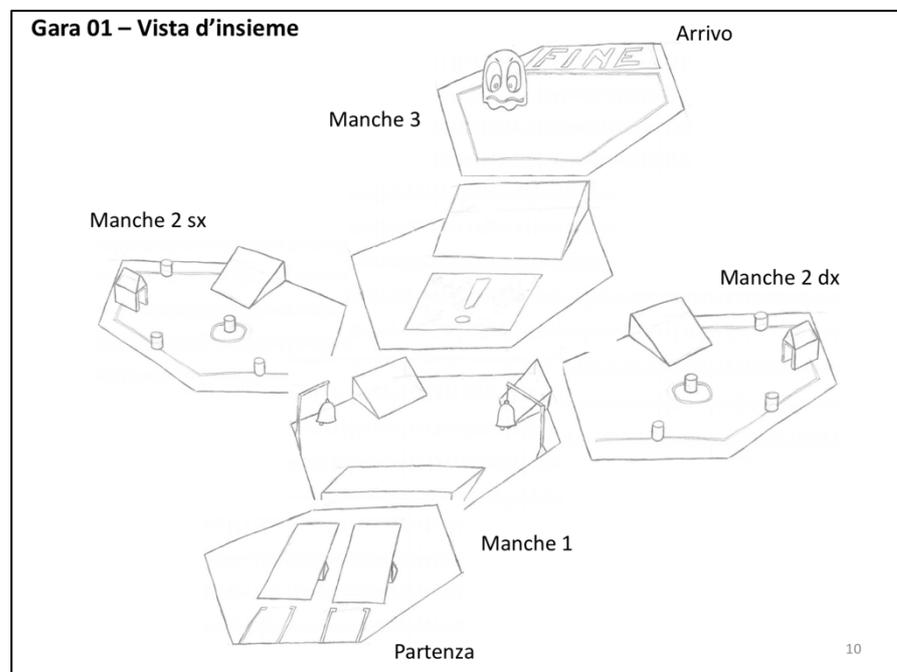


Figure 12 Storyboard sketch

Modern previz uses digital technology, and enables a director to explore, devise, develop and express their vision for sequences of their film. Previs companies work closely with the director, production designer and VFX supervisor to create the template for complicated sequences.

2.4.2 Animatics & CG Renderings

From the “simple” storyboard, it is possible to talk about animatics when a CG software is used to create a virtual version of what the director has to pre-visualize. For example, it is possible to create an optically correct Virtual Film on CG that helps a filmmaker to pre-visualize his work.

FrameForge Storyboard Studio [28] is a previsualization storyboard software used by directors, cinematographers, VFX Supervisor and other creatives in the fields of filmmaking, television production, filmed advertising, industrial videos and other filmed or video content. The software comes with a library of 3D objects that allow for the construction of precise 3D spaces (sets and locations) using a drag and drop paradigm. These sets are viewed through virtual cameras which can exactly mimic the optical properties of real-world cameras. Figure 13 shows a view of the software interface.



Figure 13 FrameForge Storyboard Studio

A TV set can be designed by scenographers with CG rendering software to give previews to a set director and make decisions about the equipment or camera angles. Figure 14 shows an example of a rendering of the set of a TV show, and a photo of the actual TV show.



Figure 14 TV Show Rendering & real shooting

2.4.3 Real-time compositing

As anticipated, in these days, in tv production very often a virtual studio is used instead of a traditional one. In a context of a virtual studio, modern technologies allow to show in real-time the compositing of the real and the virtual environment.

In this way, a director can view the virtual environment during the recording, with also some VFX that is shown in real-time. The final result could be different because of post-production techniques, but he can have an exhaustive preview of the result. In the next section some examples of this type of technologies will be presented.

2.5 AR for Television

In a virtual studio context such as the one just outlined, but also in a traditional studio, sometimes some AR features are included.

This section presents examples of how AR is applied also in the fields of cinema and television.

The Viz Virtual Studio [3] solution is a set of software and technologies to create virtual studios. It should provide a perfect match between the real and the virtual scene elements. The reality is a compromise between the technical possibilities and the perceptual ability of the audience. The background (typically green) is masked away in real-time with a virtual software-created background. The physical objects in front (persons, desks, etc.) is merged with the virtual background creating the illusion of physical presence. The background will be created and prepared in advance, typically using simple images or 3D scenes. The Viz Engine, a real-time compositing engine, is responsible for combining the physical and virtual elements and for displaying the results. To display the results, it is possible to send the signal to a display or to an IP stream.

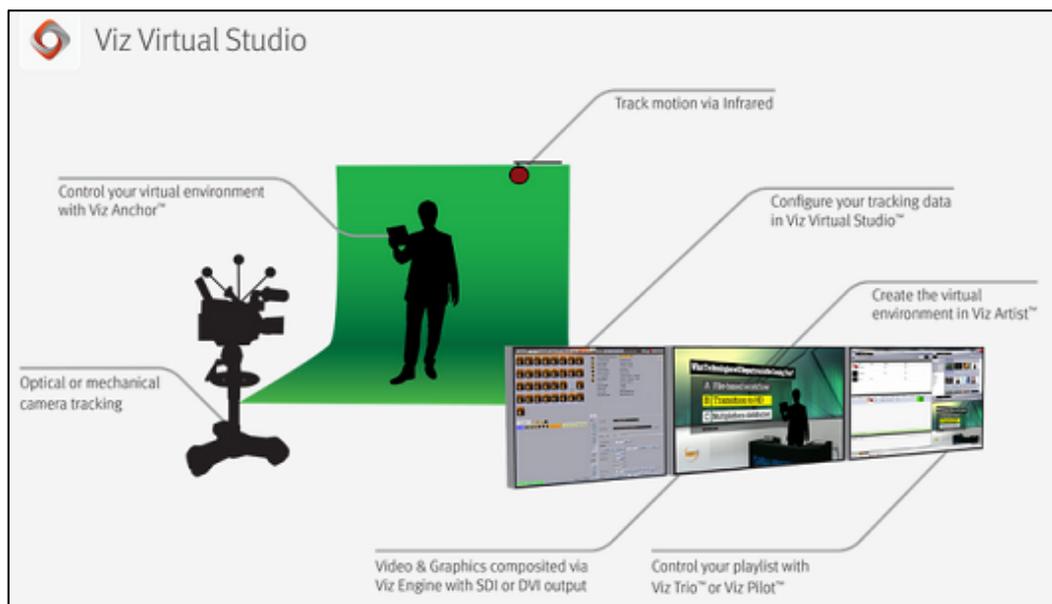


Figure 15 Viz Virtual Studio components.

An important part of the virtual studio is the camera tracking that uses either optical or mechanical measurements to create a live stream of data describing the perspective of the camera. Exact camera and sensor tracking with as little delay as possible is the key difficulty to solve in a virtual studio. Figure 15 shows the components of this system.

A similar system is used in RAI (Radiotelevisione italiana S.p.A.) virtual studios.

Cameras position and rotation is tracked with rotary encoders with IP transmission interfaces, and the virtual studio is managed with Brainstorm's eStudio [29].

Brainstorm's eStudio is a toolset that provides solutions for all kinds of 2D and 3D real-time graphics and it is also the render engine. With it is possible to design a virtual set using camera tracking and external chroma keys, or by using its own built-in chroma keyer.

With systems like the ones just described is possible to include features of AR. Traditional TV graphics are placed on top of camera images and do not react to changes in the images. With AR, the graphics can maintain the position when the camera moves, thanks to computer vision and camera tracking. This feature can be included in a virtual studio, but also in



Figure 16 AR in a tv studio

traditional one where no green screen is present. For example, it is possible to put a 3D graphical element on a table and the graphic keeps its position when the camera moves. Figure 16 shows a similar scenario.

Ncam [30] consists of a multi-sensor camera bar and a tethered server system. The camera bar attaches to any type of camera quickly and simply, whether hand-held, Steadicam, dolly or crane mounted. It takes minimal alignment time, and it is scalable to work simultaneously over multiple cameras. This system enables virtual production through real-time camera tracking, providing live previsualization of environments, set extensions and CGI elements directly in-camera while shooting.

SolidTrack [31] is a real-time previz and augmented reality solution developed by SolidAnim. This technology allows the real-time tracking of all kinds of camera as well as the overview of visual effects in real-time while shooting. With a minimal footprint, just a single survey camera linked to the film camera, SolidTrack captures in real-time the position and orientation of the film camera, including intrinsic parameters (zoom, focal, etc.).

Both of the last tracking solutions presented do not require any tracking markers as they automatically track the environment. This is important because it means there are no modifications required to the set or location to allow these systems to work.

On television, other examples of AR appeared as annotations of live camera footage during sports events. Notable examples of this concept are the virtual 1st and 10 lines in American football or the offside line in soccer, which are superimposed to help to understand the action. This type of technology is employed also to add virtual overlays in most sport events. Since the live action is usually captured by tracked camera, interactive viewpoint change is possible albeit not under the viewer's control.

Another example is the Fox-Trax system, used to make visible in every moment the hockey puck, but AR is also used in cars racing, swimming, etc.

Chapter 3 Technologies

In this chapter will be introduced the technologies used in this thesis work. The application has been developed for HoloLens with Unity 3D Engine.

3.1 Microsoft HoloLens

Microsoft HoloLens is a mixed reality head-mounted display (HMD) developed and manufactured by Microsoft. HoloLens was the first device running the Windows Mixed Reality platform under the Windows 10 computer operating system. Figure 17 shows the device.



Figure 17 Microsoft HoloLens

In this paragraph the technical specifications of the device are presented, also listed in the official documentation [7] and better analyzed in the article [32].

3.1.1 Hardware: Sensors, Optics, Processors and Memory

Most of the hardware is located in front of the unit, and it consists in a multitude of sensors, the computer unit, cameras and projection lenses.

More in detail, the sensors of HoloLens, visible in Figure 18, are:

- ◇ an inertial measurement unit (IMU) which includes an accelerometer, gyroscope, and a magnetometer
- ◇ 4 environment understanding cameras
- ◇ a depth camera 120°×120° angle of view
- ◇ a 2.4MP photographic video camera
- ◇ Mixed reality capture
- ◇ 4 microphones
- ◇ an ambient light sensor



Figure 18 HoloLens Sensors

The HoloLens must be calibrated to the interpupillary distance (IPD), or accustomed vision of the user. The images are projected and displayed in the lower half of the see-through holographic lenses (waveguides), shown in Figure 19 with a holographic resolution 1268x720 per eye.

Unfortunately, the field of view is limited to 30° horizontally and 17° vertically.



Figure 19 HoloLens Display System

HoloLens had an Intel Cherry Trail SoC, containing the CPU and GPU, and the Holographic Processing Unit (HPU).

The system architecture is essentially a mobile hardware with a 64 GB eMMC SSD and 2GB of LPDDR3 RAM (1GB for the SoC and the HPU); it is based on x86 architecture and runs Windows 10. The SoC carries out the majority of the work, since it handles the operating system, the applications and the shell.

The HPU main purpose is processing the data originated from the sensors, such as the wearer's gestures, and sending it to the SoC as a tidy package. The HPU's design allows Microsoft's engineers to assure latency and duty cycles during the elaboration.

3.1.2 Basic Usage

In addition to see the holograms projected in the real world, a user can also interact with them, and, thanks to the spatial sound, he can pinpoint holograms even if they are behind him. HoloLens input system is quite simple, and Microsoft explains it in detail in the official documentation [33] [34].

HoloLens glasses use the position and orientation of the user's head to determine the gaze vector: a ray that the applications intersect both with their holograms and with the spatial mapping mesh to establish at what virtual or real-world object the user is looking. Generally, apps should use a cursor to give the user confidence in with what they are about to interact.

The HoloLens device is able to recognize hand gestures that permit the user to take action in mixed reality. There are three main inputs:

- ◇ The “Bloom” gesture (Figure 20) to go Home (similar to pressing the Windows key on a keyboard), performed holding out the hand, palm up, with the fingertips together, and then opening the hand.

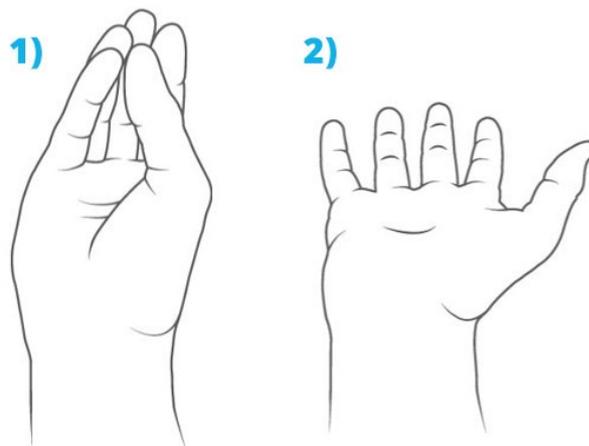


Figure 20 Bloom Gesture

- ◇ The “Select” or “Air Tap” gesture (Figure 21), the primary interaction to interact with an hologram, consisting of a press followed by a release; it is performed positioning the hand with its back in front of

the face, raising the index finger to the sky and then tap, by flexing the index finger down (the press) and then back up (the release).

The interacted element is the one that user is gazing at that moment.

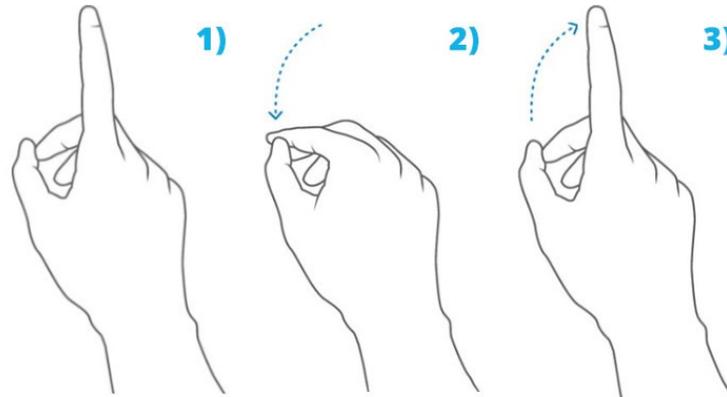


Figure 21 Air Tap Gesture

- ◇ Hold is simply maintaining the downward finger position of the air tap. The combination of air tap and hold allows for a variety of more complex "click and drag" interactions when combined with arm movement such as picking up an object instead of activating it or "mousedown" secondary interactions such as showing a context menu.

3.1.3 Developments: HoloLens 2

Microsoft HoloLens 2 [35], shown in Figure 22, is the follower of HoloLens, and it has been announced and released in 2019. It refines the holographic computing journey started by HoloLens (1st gen) to provide a more comfortable and immersive experience paired with more options for collaborating in mixed reality. HoloLens 2 has a diagonal field of view of 52°, improving over the 34° field of view of the first edition. Furthermore, this device not only recognized hand gestures, but it tracks hands position and action visually. With this feature, hands can be used to interact with hologram elements, for example pinching them between thumb and forefinger. The resulting mixed reality is much more immersive and the input system more intuitive.

This device will not be discussed in detail because it has not been used in this thesis work, but it is conceivable that will be used in future developments of the project.



Figure 22 Microsoft HoloLens 2

3.2 3D Scan and Modeling

The 3D models included in this thesis work are reproductions of the real props of the Rai TV show, use case of this work. They have been obtained from a process of 3D scan following by a 3D processing.

A structured-light 3D scanner has been used, a 3D scanning device for measuring the three-dimensional shape of an object using projected light patterns and a camera system. The specific device used is the Artec Eva [36] by Artec 3D, that has a 0.1mm 3D point accuracy.

It scans quickly, capturing precise measurements in high resolution, producing high resolution meshes, also with fine textures.

These meshes are too heavy to be manipulated in a 3D application, so it is necessary to process them with a 3D modeling software.

Blender [37], is the software used to obtain suitable meshes for the final application. It is a free and open-source 3D computer graphics software, and his supported features are modeling, rigging, animation, simulation, rendering, compositing and motion tracking, as well as video editing and game creation tools.

Although the capabilities of Blender extend to various fields related to creation of computer graphics, in this case it has been used simply to decimate the vertexes of scanned meshes, obtaining low poly meshes, suitable for the HoloLens application. Furthermore, a process of straightening and centering in the origin of 3D space must be done in Blender to make meshes ready to be imported in the game Engine.

Figure 23 shows an example of processing from a high poly mesh to a low poly mesh. In this case, a tower mesh in OBJ format, the original file has about 250k vertexes. Applying a Decimate operator with factor 0.1, it has been obtained a low poly mesh of about 4k vertexes, suitable for the application.

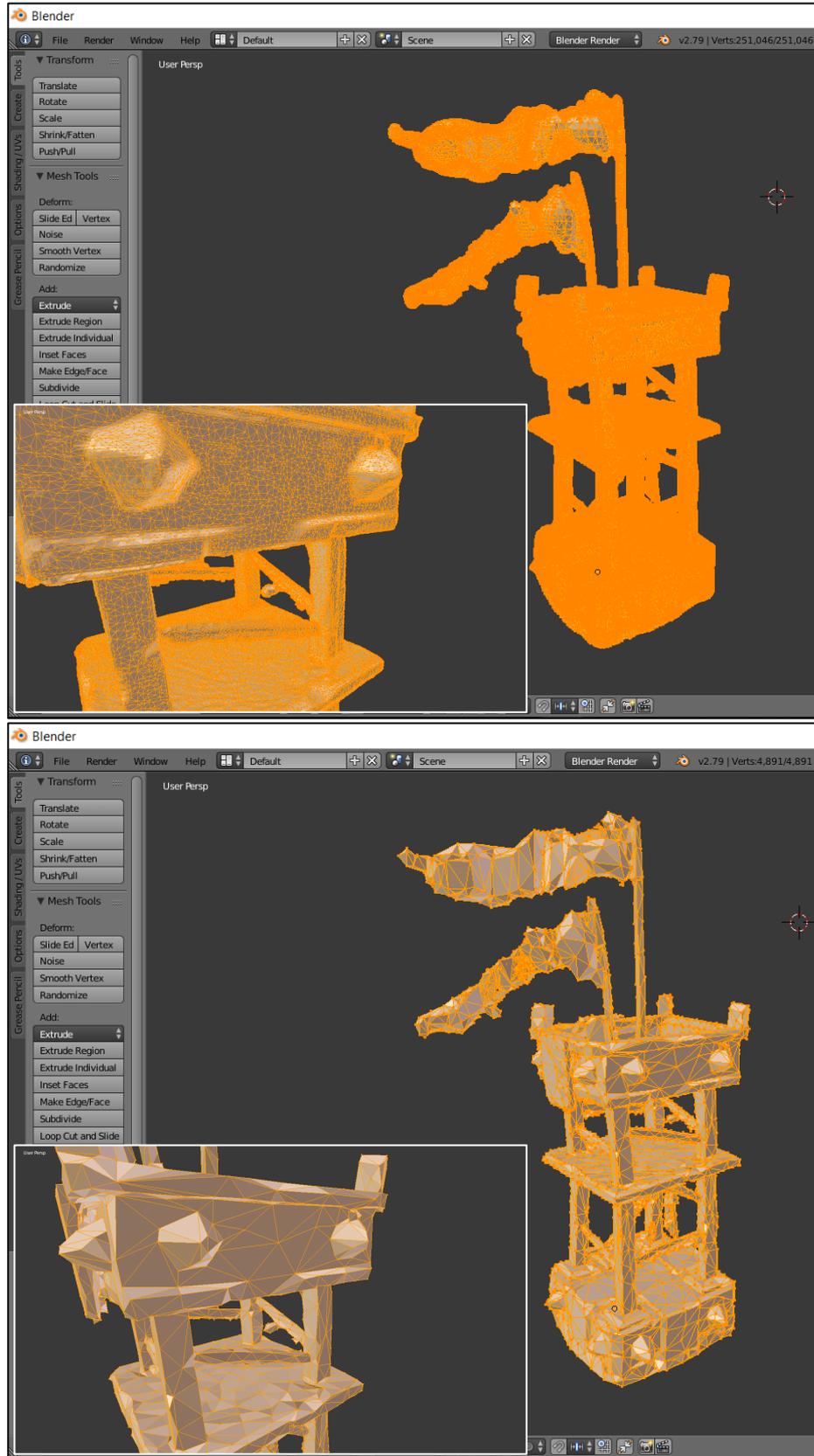


Figure 23 High to Low Poly in Blender

3.3 Unity 3D

Unity is a cross-platform game engine developed by Unity Technologies, first released in June 2005. As of 2019, after several major versions released, it can be definitely considered one of the most popular game engines. Its power is that it is easy enough for a beginner, thanks to its user-friendly interface and the abundance of documentation and tutorials, but at the same time is powerful enough for experts and businesses.

The engine can be used to create 2D, 3D, VR, and AR games, as well as simulations and other experiences.

One of the best advantages of Unity is that allows to develop cross-platform applications, so a developer can write one application and then deploy it for many devices or systems (Android, iOS, Windows, MacOS). This gives the possibility to have more people use and interact with the app.

Unity offers a scripting API in C#, created in the Unity editor in the form of plugins, and editable with Microsoft Visual Studio or with external code editors.

In this project has been used C# with Visual Studio for simplicity and for the compatibility with UWP (Universal Window Platform). With this platform it necessary to build the Unity Application and then flash it to HoloLens with Visual Studio.

The Unity Editor window is split up into a couple of sections, as show in Figure 24.

The main sections are:

- ◇ **Scene View.** Allows placement and movement of GameObjects in the Scene.
- ◇ **Game View.** Previews how the player will see the scene from the camera.
- ◇ **Inspector.** Provide details on the selected GameObject in the scene.
- ◇ **Assets / Project.** All prefabs, textures, models, scripts, etc. are stored here.

- ◇ **Hierarchy.** Enables nesting and structuring of GameObjects within the scene.

In the following a brief description of the main components of the engine will be presented.



Figure 24 Unity Editor

GameObjects are the core building block of everything in the Unity game engine. They do not accomplish much in themselves, but they act as containers for **Components**, which implement the real functionality. A GameObject always has a Transform component attached (to represent his position, orientation and scale) and it is not possible to remove it. The other components that give the object its functionality can be added from the editor or from a script.

In Figure 25 is shown an example of GameObject called Cube. Seeing the Inspector panel is possible to see that it has a Mesh Filter and Mesh Renderer component, to draw the surface of the cube, and a Box Collider component to represent the object's solid volume in terms of physics.

It is not in the purposes of this paragraph to present every component available in Unity, that offers several functionalities. The official manual [38] presents all the components in detail.

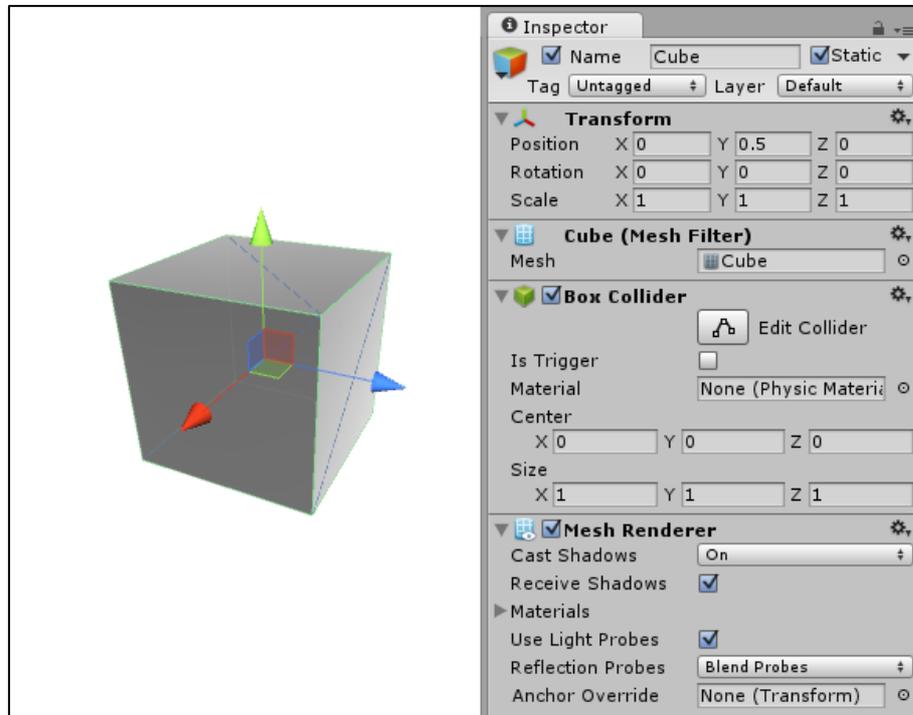


Figure 25 Game Object

Instead, it is important to introduce the simple notion of a **Prefab**. After a **GameObject** has been built in the Scene, it is possible to store it as a Prefab asset type, complete with components and properties, in the Assets folders. The prefab acts as a template from which is possible to create new object instances in the scene, from the editor or from a script.

3.4 HoloToolkit

3.4.1 Premise

In this project has been used the HoloToolkit (HTK) in its final stable version (2017.4.3.0). After that version the API changed his name in MixedRealityToolkit (MRTK), a new project whose purpose is to include the support both for the new HoloLens 2 and for the VR headsets compatibles with Microsoft APIs. At the beginning of this thesis work, MRTK was still in development, some features were not released yet, and a big part of the documentation was missing.

3.4.2 Introduction

The HoloToolkit is a collection of scripts and components intended to accelerate the development of holographic applications targeting UWP. It is provided to developers, for the development in the Unity Editor, as a simple *unitypackage* file importable into a Unity project. It also contains many useful example scenes for developers to explore and understand how to use various parts of the HoloToolkit. It includes several feature areas, organized in folders.

The following table lists each feature area and provides a brief description.

Feature Area	Description
Input	Allows developers to include input in their apps, such as gestures, clickers, gaze, and voice commands.
Sharing	Allows developers to make shared experiences. This allows many users to see and experience the same app together.
Spatial Mapping	Tools to enable and use spatial mapping. This allows your application to interact with the physical environment, such as walls, floors, tables, and objects nearby.
Spatial Understanding	Allows an application to understand the physical environment. For example, it can differentiate between chairs, tables, and other common structures.
Spatial Sound	Allows developers to include spatial sound capabilities, so that objects sound as if they are physically in your environment.
Utilities	A collection of useful utilities, such as frame rate viewers, object locators, ability for objects to follow you, menu items for configuring Unity, and more.
UX	Useful common UX controls for holographic applications.
Build and Deploy	Time-saving features that enable the ability to quickly build and deploy projects directly from Unity without having to first go through Visual Studio.

A more detailed description of the features used in the thesis work is given in the following.

3.4.3 Input: Gaze, Gestures, Voice

The input features in HoloToolkit provide developers with the ability to interact with holograms. It includes a collection of scripts that allow

applications to recognize gestures (such as the air-tap gesture and the hold gesture), inputs from devices such as clickers, gaze capabilities, and voice commands. There are a wide range of test scenes where it's possible to test various features included in the input feature set.

By taking advantage of the position and orientation data of user's head, HoloLens is capable of determining the head gaze vector. This can be described as a sort of ray pointing straight forward from user's eyes. The ray can be utilized to pinpoint holograms as well as to determine whether the participant is looking at a virtual or real-world object. To stimulate participant's intention in an application created for HoloLens, a very common practice is to indicate the gaze with continuous visual feedback such as a cursor. The cursor aids the user in understanding where his focus is at all times, as well as to indicate what possible hologram, area, or other point of interest might respond to input.

Specifically, the kit provides the "Input Manager" prefab, an input system, formed by a series of scripts, that manages gaze and the various input sources currently supported by HoloLens. It also includes a fake input source that allows you to simulate input when in the editor. The gaze cursor is provided as a prefab and must be loaded in the scene.

Whereas gaze is mostly responsible for determining the point of interest, interaction is formed by utilizing user's hand gesture to act upon a targeted hologram.

Gestures recognized by HoloLens, handled by Input Manager, had been already explained in Chapter 3.1.

In addition to the two key forms of input of gaze and gestures, HoloLens supports voice as source for receiving input from the user. Voice enables the participant to directly command a hologram without the need for utilizing separate hand gestures, allowing in many cases for a much quicker and simpler way to interact with the targeted hologram. Voice can be considered

excellent at managing complicated interfaces, as the participant may use a single voice command to quickly cut through the nested menus.

The `SpeechInputSource.cs` Unity script provides ways to explicitly set keywords, or voice commands, in editor rather than code.

3.4.4 Spatial Mapping & Spatial Processing

With HoloLens it is possible to constantly track the environment and build a 3D model of the room where it is. This feature is called Spatial Mapping.

How it is summarized in [39], spatial mapping is important for several reasons:

- ◇ **Occlusion:** This tells the HoloLens which holograms to hide from view. For example, if the user places a hologram in the hallway and then walks into another room, the spatial map of that room's walls will prevent him from seeing the hologram in the hallway. Without this feature, the hologram would be visible through the wall, causing an unrealistic experience.
- ◇ **Physics & Placement:** This allows users to interact with the spatial map, for example, to pin items to the walls, or put a virtual object on a table. Objects collide with or bounce off the walls, furniture, and floors, resulting in a realistic experience.
- ◇ **Persistence:** This allows for holographic persistence, which is the ability for holograms to stay where the user left them, even after turning off the device. The HoloLens will be able to remember the tracked environment and restore any holograms previously placed in that space.

The toolkit provides a simple prefab, called **Spatial Mapping**, with three scripts:

- ◇ `SpatialMappingObserver.cs`: This script is responsible for managing the surfaces observed on the HoloLens and renders them

so they can be displayed in the scene. It offers parameters in the Inspector panel by which adjust the tracking of the environment.

- ◇ `ObjectSurfaceObserver.cs`: This script is used when HoloLens device is not used for spatial mapping but instead, a pre-existing 3D model of a room, or area, is used within the Unity Editor. It is possible to specify a custom 3D model in the Inspector panel.
- ◇ `SpatialMappingManager.cs`: This script is used to manage the feature, allowing application to use the Surface Observer or a stored Spatial Mapping mesh (loaded from a file).

Another useful component, called **Spatial Processing**, is excellent for scenarios where is needed to identify planes. With this system is possible, after the scanning done by the Surface Observer, to identify and create the floor, the walls and the other surfaces in the room.

It is made up of three scripts:

- ◇ `SurfaceMeshesToPlanes.cs`: This script is responsible for finding planes from the mesh and generating planes. In the Inspector, is possible to set the `MinArea`, which is the minimum area required before a plane will be created, and the type of surfaces that must be showed up. The `Surface Plane` prefab determines the appearance of the planes, with a material for each type of surface.
- ◇ `RemoveSurfaceVertices.cs`: This script is responsible for removing vertices (removing parts of the spatial mapping mesh) that fall within boundaries specified. A developer may want to remove vertices if he needs holes in the spatial mapping mesh or if he wants to reduce polygon count in order to improve the performance of his application.
- ◇ `SpatialProcessingTest.cs`: This is an example script that manages the system and allows to control a variety of settings related to spatial processing. This first control the Surface Observer, then it lets start the processing.

The feature of the **Occlusion** is implemented with a simple material, that apply a custom occlusion shader and can be used with spatial mapping meshes.

3.4.5 Sharing

One of the more powerful features included with the HoloToolkit is the sharing module. The sharing module allows multiple people to share the same Mixed Reality experience, locally or remotely. For example, several people wearing Mixed Reality headsets in the same room will be able to see and interact with the same holograms together. For remote users, the sharing module allows users to see avatars of each other, hear each other, interact, and collaborate.

The HoloToolkit provides components that make easy to develop a sharing experience.

This features area consists of:

- ◇ A server executable (**Sharing Service**) that facilitates discovery and connecting remote users. The server can run as a Windows Service or as a command-line application.
- ◇ A library that allows apps to connect the Sharing Service and to include sharing features.

Therefore, the HoloToolkit Sharing is built like client server architecture, as shown in Figure 26, where a service component needs to run on a mediator or server. HoloLens devices are essentially clients that use the HoloToolkit Sharing client component to send and receive communication routed through the service/mediator.

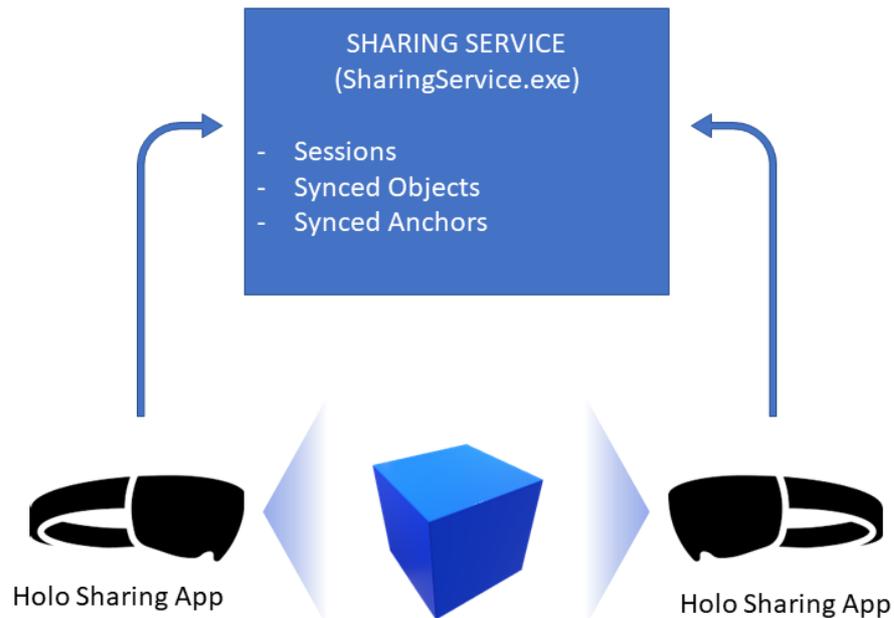


Figure 26 Sharing Service Client-Server System

In editor, the main component is the **Sharing prefab**, that enables sharing and networking.

The fundamental scripts of this feature are:

- ◇ `SharingStage.cs`: A Singleton behavior that is in charge of managing the core networking layer for the application. The `SharingStage` has the following responsibilities:
 - server configuration (address, port and client role);
 - establish and manage the server connection;
 - create and initialize the synchronized data model (`SyncRoot`);
 - create the `ServerSessionsTracker` that tracks all sessions on the server;

- create the `SessionUsersTracker` that tracks the users in the current session.
- ◇ `AutoJoinSessionAndRoom.cs`: Automatically join a default session and room on connection;
- ◇ `SharingWorldAnchorManager.cs`: This script is responsible to handle the feature of the shared world anchors. World Anchors are shared through the “World Anchor Transfer Batch”, which allows to export and import them between apps.

This library includes a “sync version” of every C# primitive, such as `SyncBool` or `SyncDouble`, but also of more complex types, such as `SyncVector3` or `SyncTransform`. Other types can be built as needed by inheriting from **`SyncObject`**, as in the following example.

```
public class MySyncObject : SyncObject
{
    [SyncData]
    public SyncSpawnArray<MyOtherSyncObject> OtherSyncObject;

    [SyncData]
    public SyncFloat FloatValue;
}
```

Any `SyncPrimitive` tagged with the `[SyncData]` attribute will automatically be added to the data model and synced in the current sharing session.

Another key component is the script `DefaultSyncModelAccessor.cs`, a component that allows other components of a game object to access the shared data model as a raw `SyncObject` instance.

3.4.6 UX (User Experience)

HoloToolkit offers a collection of building blocks to design a 3D UI and interact with objects in 3D space. The feature area includes holographic buttons, 3D labels and other useful controls, some of which are shown in Figure 27.

Furthermore, the kit includes a specific UI to manipulate 3D objects, made of two main components: the Bounding Box Rig and the AppBar. It is shown in Figure 28.

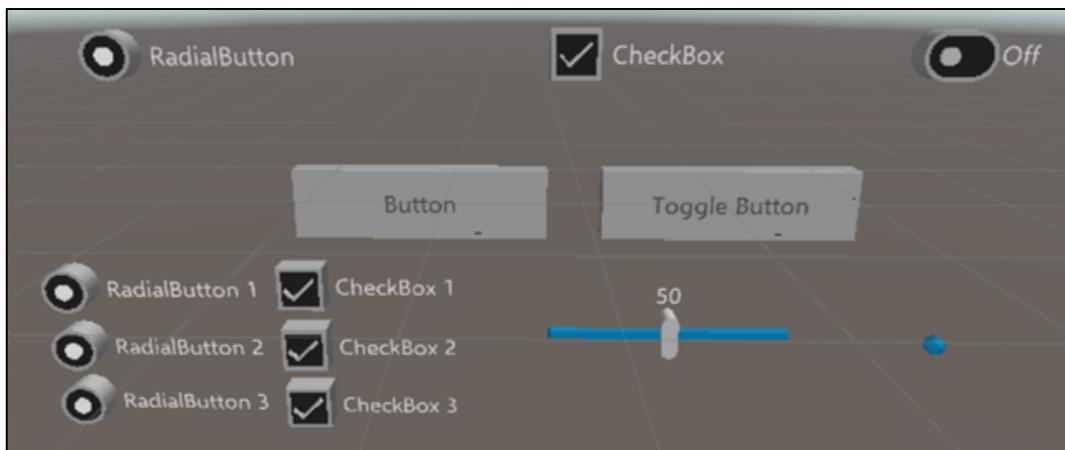


Figure 27 HoloToolkit UI controls

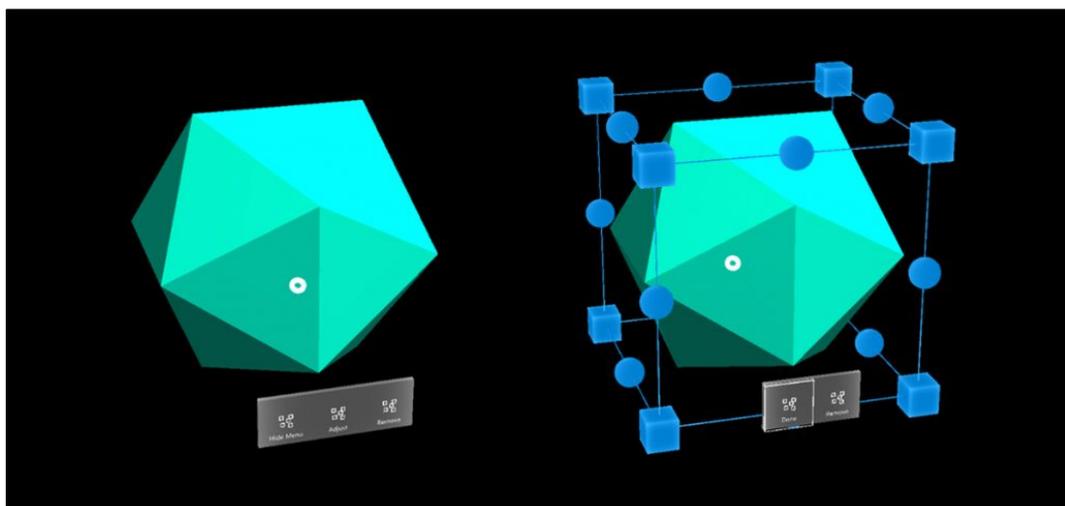


Figure 28 3D objects interaction UI

Chapter 4 A HoloLens Application to Support Television Set Design

4.1 Project Requirements

The project described in this thesis has been proposed by RAI Radiotelevisione Italiana, and it has been realized in collaboration with RAI-CRITS [40] (Centro Ricerche, Innovazione Tecnologica e Sperimentazione), within the context of a European project called 5G-TOURS.

The aim of this project is to develop a mixed reality application, with a novel visual user interface and features of sharing and environmental awareness, useful in set creation, in the context of TV pre-production.

With this application, the set designer and the artistic director of a TV show should be able to interact with the environment and collaborate to design the scene set with free hand gestures or vocal commands. They should be able to set up the set trying different layouts and then save these configurations. The experience offered by the application must be accessible also in single user, when just one person wants to use it, and he doesn't need to collaborate with someone else.

The application has been developed for a real use case, in order to try to introduce a new approach of set creation in a real tv production context.

Rob-o-Cod, a tv program broadcasted on Rai Gulp, is a kid's show where boys and girls challenge each other using programmable robots. They have to move robots in competition fields, which are made up of tables where some routes with many obstacles are outlined. Figure 29 shows the competition fields and the props of the tv show.

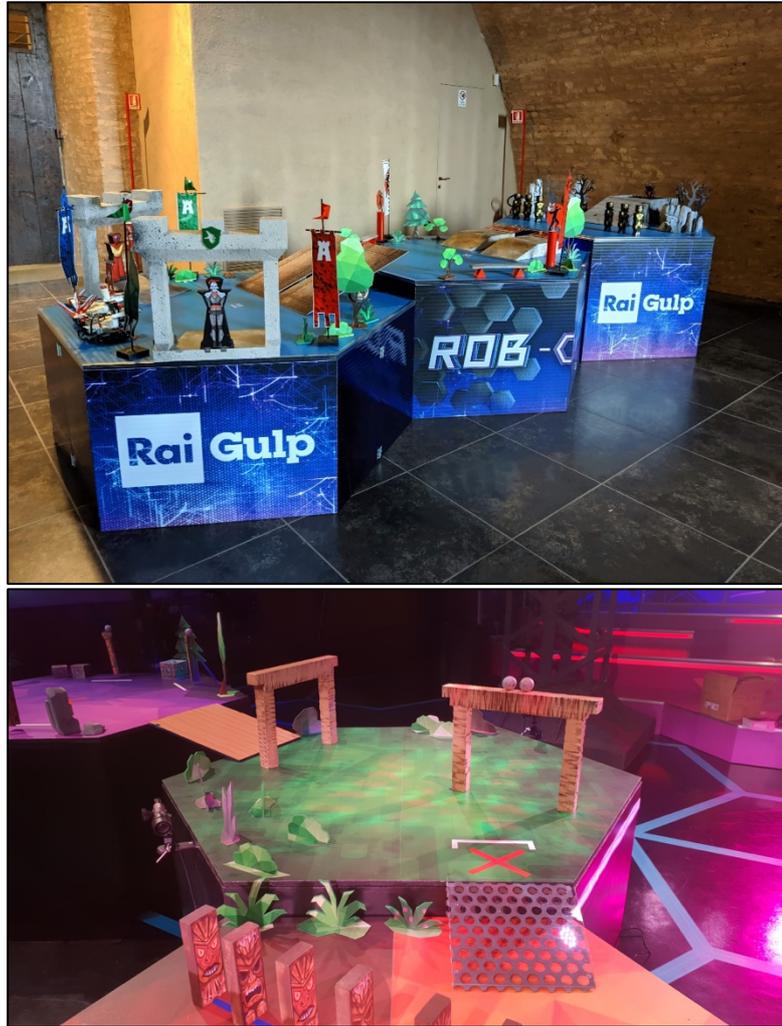


Figure 29 Rob-O-Cod Props

Summarizing, the main features required, formulated thanks to several meetings with scenographers and set directors in RAI, are:

- ◇ Scan the set to interact with it;
- ◇ Place holographic objects on the surfaces, from a set of available props, and have the chance to move, rotate and scale them;
- ◇ Take pictures of the equipped set;
- ◇ The experience must be available to a single user or as a collaborative experience for more users.

4.2 System Architecture

In order to meet the requirements and for the device available, Microsoft HoloLens has been used as MR device. As widely described previously, it allows an easy and intuitive interaction and a good environment perception, thanks to his sensors.

HoloToolkit components (Section 3.4), taken from the basic toolkit and from the example package, have been used, combined and modified to implement the required features of the application.

In this section will be presented the system's architecture with his main cores and workflows involved.

4.2.1 Contents creation workflow

Real scene objects have been scanned with a 3D scanner, the Artec Eva presented in Section 3.2, producing high resolution textured meshes, and then processed with Artec 3D Studio [41] to obtain a first low-poly version

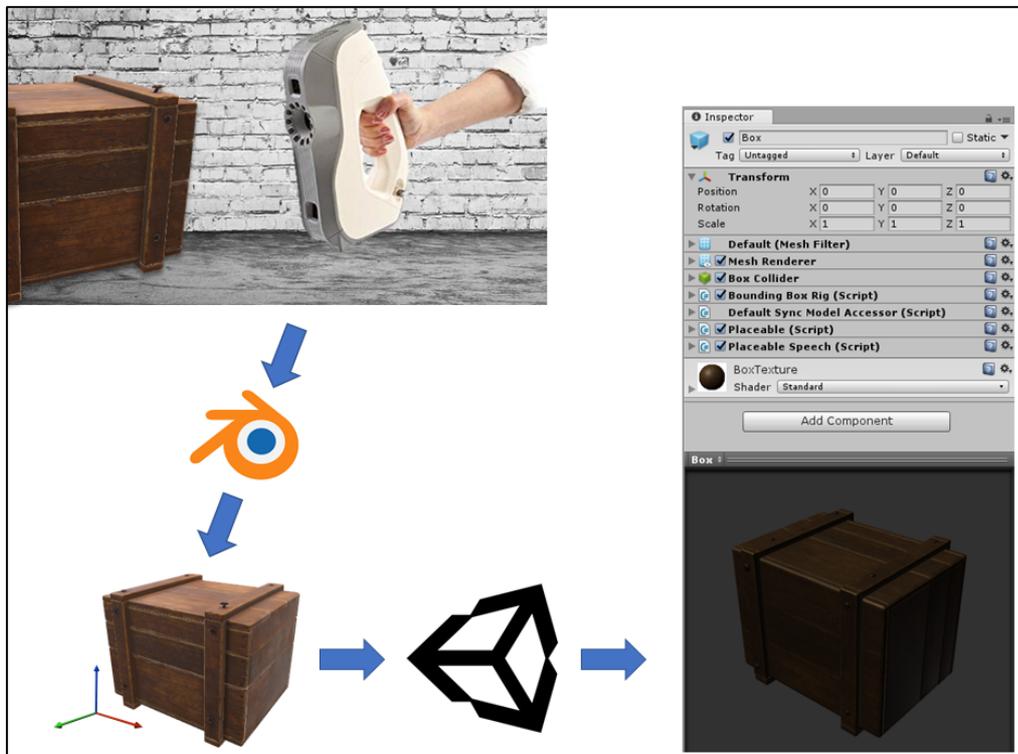


Figure 30 Contents creation workflow

of the meshes. Then these meshes have been further processed with Blender (Section 3.2) to reduce polygons and fix some details, obtaining the final low resolution meshes, suitable to be imported into Unity (Section 3.3). Finally, in the game engine, a GameObject has been created for every 3D model, with the mesh and all the other components and scripts used in the logic of the application, that will be presented in the next sections.

A simple scheme of the described workflow is represented in Figure 30.

3D models are saved locally in this application. Another thesis work developed parallelly, linked to the same project in RAI, focuses on the development of the system of loading of meshes and textures from a remote database, since natively Unity does not offer this feature. Furthermore, it aims to develop a search system and an interface to find models through tags and properties of 3D models.

4.2.2 Application Manager and Interface

To manage the phases of the application flow, an instance of the AppManager class handles a variable, called CurrentAppState, that tracks the current state of the experience. The application switches from a state to another depending on the choices of the user in the menu.

These phases have been defined in the following structure:

```
public enum AppState
{
    Starting = 0,
    SelectingExperience,
    Connecting,
    SelectingScan,
    Scanning,
    ChoosingPlanes,
    MainMenu,
    EditingScene
}
```

During every phase every necessary component is activated, and unnecessary components are disabled. Furthermore, during every phase the menu is updated, showing the controls useful to the user to proceed in the experience.

The menu constitutes the main user interface, and the user can choose to interact it with the standard gestures of the device or with vocal commands. The menu offers user buttons by which he can choose what to do during the experience, and some labels explain user what he can to do.

In order, the menu gives user the opportunity to:

- ◇ Choose between “Single user experience” or “Multi User Experience”;
- ◇ Insert the Server IP address (in case of Multi User);
- ◇ Start scanning the environment;
- ◇ Choose the wanted planes between the ones found by the application, or rescan the environment;
- ◇ Edit the scene placing holographic props on top of the planes;
- ◇ Take a picture of the built mixed reality tv set.

A view of the menu, in the phase of scene editing, is shown in Figure 31.

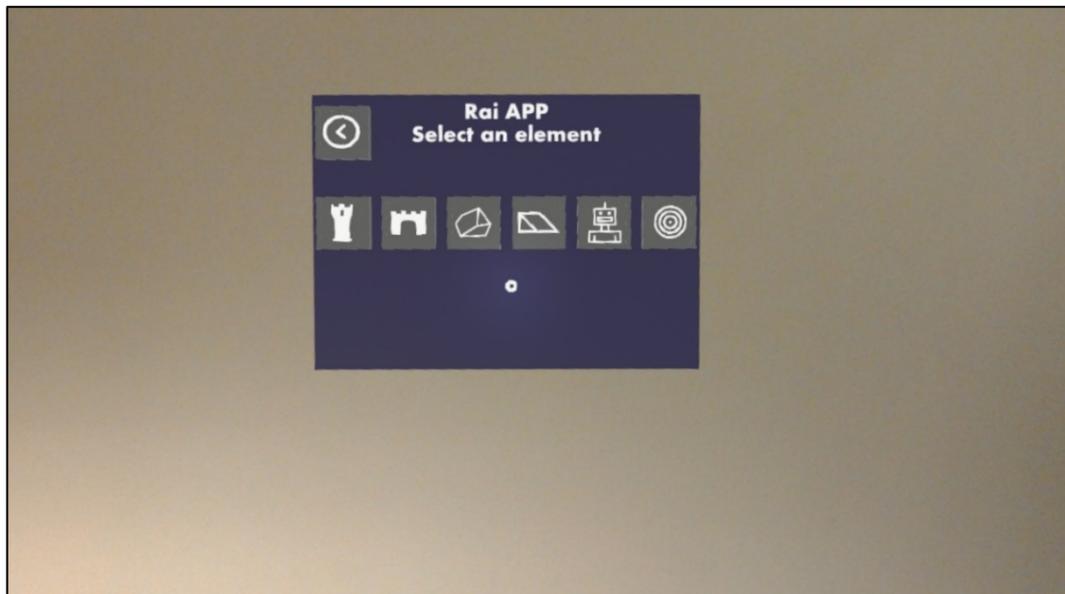


Figure 31 Menu appearance

4.2.3 Environmental Awareness System

The surfaces on which props can be placed are created as a reconstruction of the tables found in the user's room.

Following the approach of the `SpatialProcessingTest.cs` example script, proposed by the HoloToolkit, has been developed a custom script, called `SpatialProcessingManager`, that is responsible to manage the whole process of scanning and processing of the environment.

This process starts with the Spatial Mapping component that scan the environment for an amount of time, limited by a timer, during which the user can visually see the creation of the mesh of the environment, as in Figure 32.

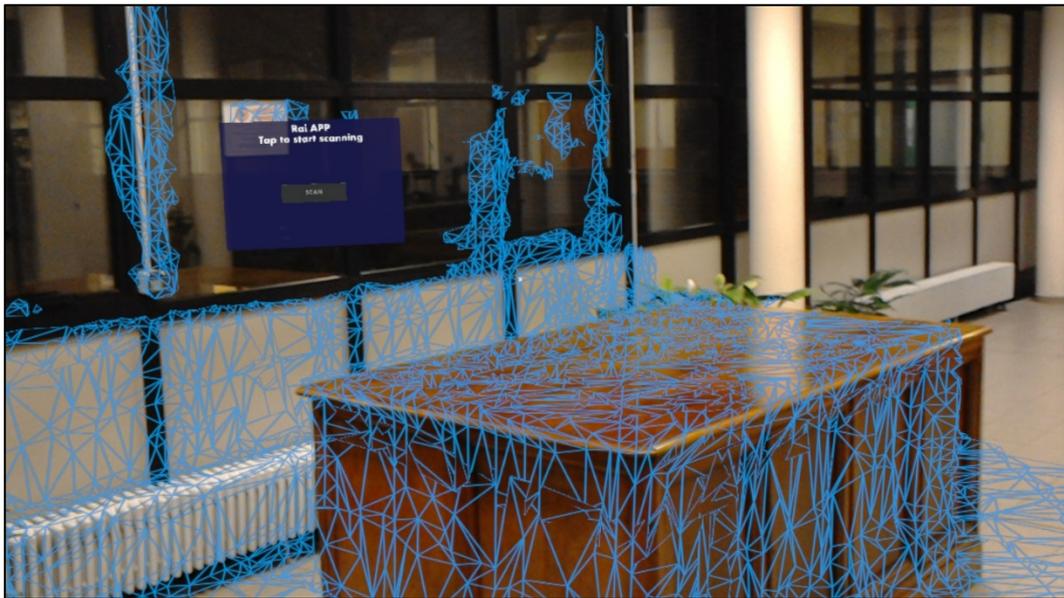


Figure 32 Spatial Mapping

Then, the mesh is processed by the Spatial Processing component, that is set for this application to find only the surfaces classified as “tables”.

These table surfaces are shown to the user, and he must choose at least one of these surfaces to proceed in the experience, or, if he is not satisfied with the scanning, he can retry to scan the environment to get better results, maybe adjusting room lighting.

Moreover, the user can choose one or more surfaces of interest by tapping on them, or selecting them by voice, and they change color to green as a feedback for the user.

Table surfaces appears like in Figure 33.

Finally, the user proceeds following the instruction on the menu to place objects on the chosen planes.



Figure 33 Surface Plane

This functionality is implemented with a prefab object that represent a plane, that is loaded in scene, scaled and rotated, for every surface found. The Spatial Processing Manager keeps track of active planes, saved in an array, and deletes the ones discarded by the user.

The described workflow is summarized by the scheme in Figure 34.

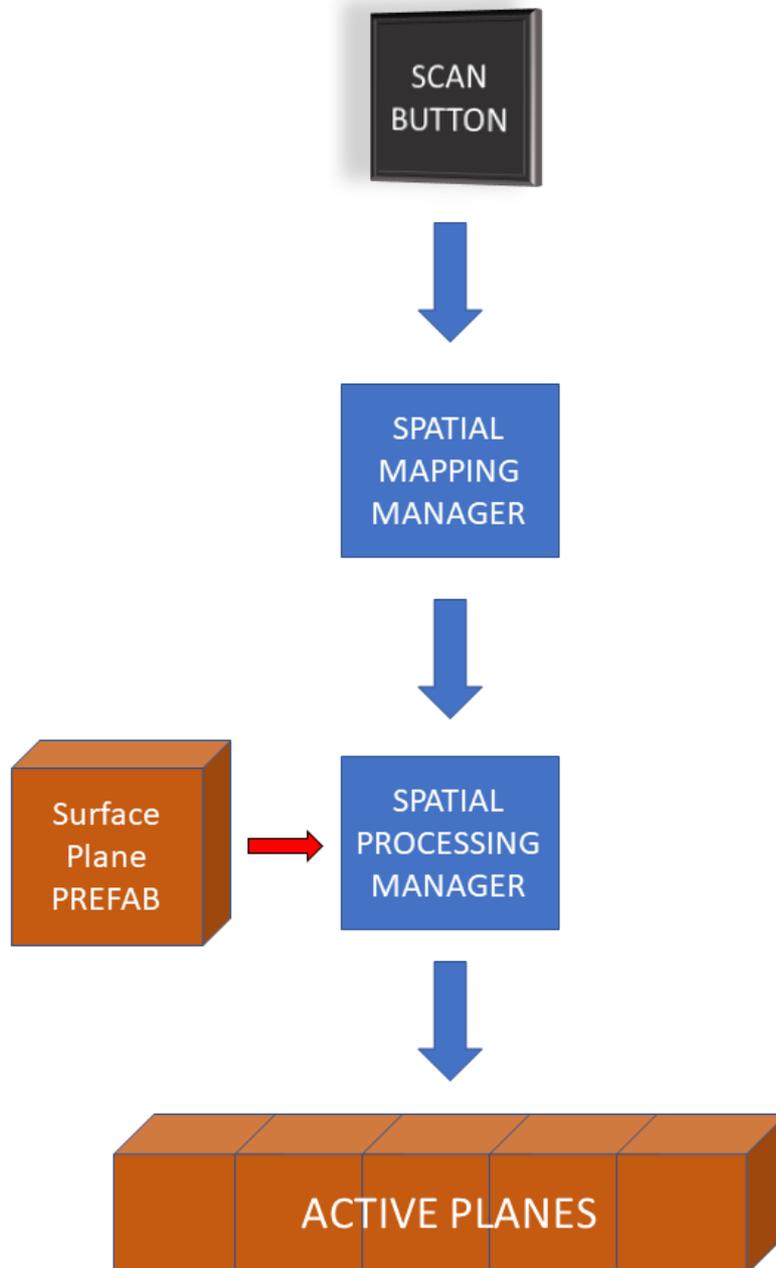


Figure 34 Spatial Processing System

4.2.4 Placing and Editing System

With a surface identified by the device, the user is allowed to select items from the menu to use them in the scene. The Placing Manager component is responsible to directly spawn items in the 3D space when the experience is set in “Single User mode”; instead, in “Multi User mode”, it predisposes the loading but others sharing component are involved in the process. The latter will be better explained in Section 4.2.5.

When a spawn button in the menu is selected, the respective item appears on top the plane nearer to the user. Then, the user can select the item to move it across that the surface or on another active plane. A visible shadow



Figure 35 Placing System

indicates where the object is attempting to be placed, and it is green to notify a valid placement (on top of active planes), red to notify an invalid placement. Figure 35 shows the described feedback system.

Placed objects can be rotated, scaled or removed in two ways.

The first method consists in simple vocal commands: gazing an item and pronouncing “rotate”, “scale up”, “scale down” or “remove”, it responds doing the respective adjustment or disappearing. Some fixed values of rotation and scaling have been defined to obtain a discrete scale of values useful in the application.

The second method of interaction is offered through the gestures, thanks to a visual interface. Every item has a button bar floating above itself, that gives user the choice to:

- ◇ hide the bar;
- ◇ remove the item;
- ◇ active the adjustment UI.

The adjustment UI, anticipated in Section 3.4.6, looks like a grid around the item, and gives user the opportunity to rotate or scale the item (Figure 36).

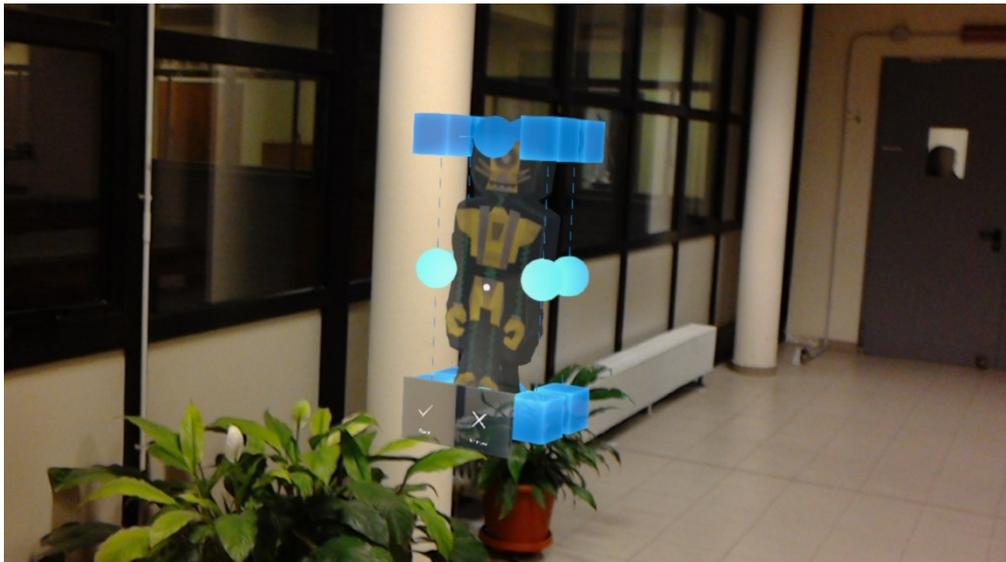


Figure 36 Adjustment UI

4.2.5 Sharing System

If the experience chosen is the “Multi User Experience”, a holographic numpad is shown to the user and he have to insert the IP address of the Sharing Service, that has the role of server in the system, as better explained in Section 3.4.5. It looks like in Figure 37.

Therefore, to proceed in the collaborative experience, an instance of the Sharing Service executable must be running in a machine connected in the local network.

Here too, user can insert the IP address as he prefers through the gaze and gestures input system or by voice pronouncing the IP one digit at a time. Once done, he can try to connect to the server by tapping the “Connect” button or by saying “Connect”.

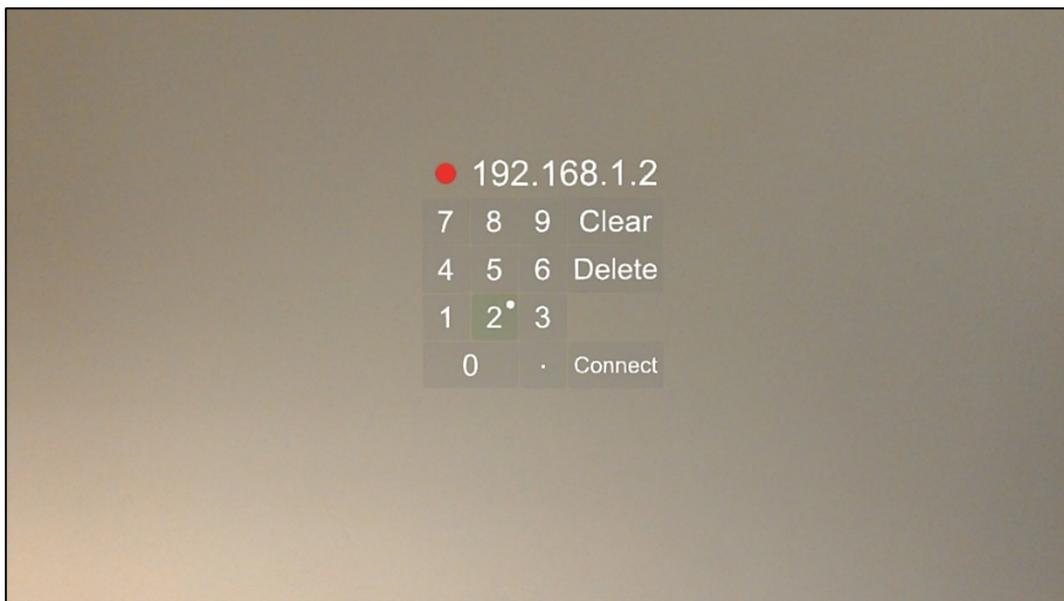


Figure 37 Connection Numpad

When the application is correctly connected to server, the numpad give way to the main menu and the experience continues with the Sharing component activated.

In this mode, during the experience every user connected can spawn props in the scene from the menu and interact with them. Every action and modification done to the holograms is done to the shared instances of the

objects, so that all users are synced to the same mixed reality. With this feature, for example, a set designer can share the view of the studio to the director of the show so that they can collaborate to make decisions about the setting.

Figure 38 shows two users with HoloLens synced in the same session.



Figure 38 Sharing Experience

Essentially, when a user places a holographic element in the environment, the Placing Manager handles the feature. This component is better analyzed in Section 4.2.4.

In “Multi User Experience”, GameObjects are not just instantiated locally, indeed they are shared. Every GameObject to share is encapsulated in a Sync Object and then it is loaded to the SyncSource, a SyncArray structure where every element shared in the session is saved. Then, the Spawn Manager system of every user takes care of showing or removing all the shared elements, following users’ actions.

Every shared hologram has a DefaultSyncModelAccessor component attached, by which the system manages the synchronization of his position, rotation and scale, as given by user.

Figure 39 schematizes the workflow of the sharing system.

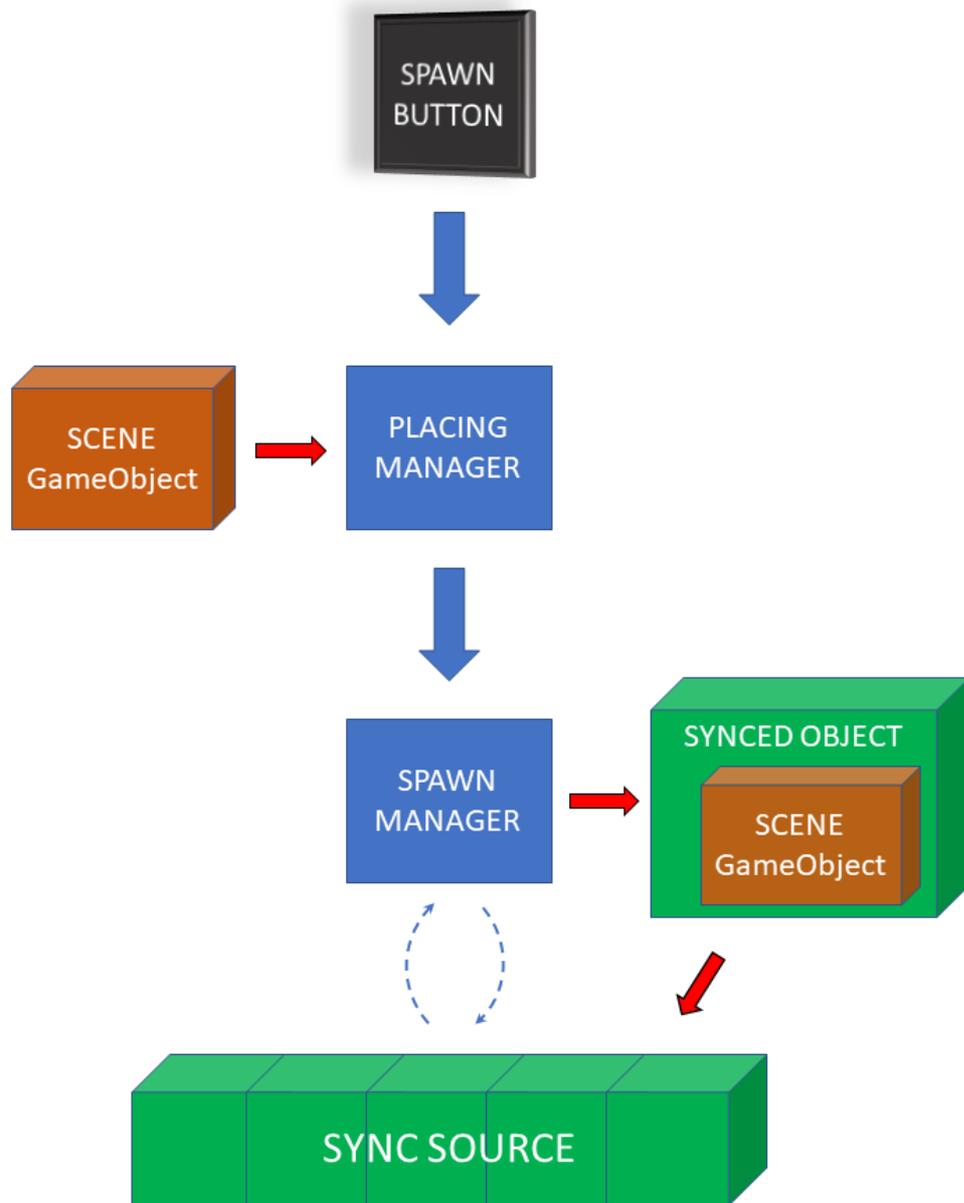


Figure 39 Sharing System Scheme

4.2.6 Scene Capture

At the end or during the set building with this application, it is possible to take snapshots of the scene with a button in the menu.

The photos produced by the device replicates the first-person view of the user that is wearing it, so they mix the real and the digital contents.

These photos constitute the output of the workflow of the application. For example, a set designer can set up the studio and then show snapshots of the studio to the direction of the show. Figure 40 shows an example of snapshot taken by the application.



Figure 40 Snapshot taken by the application

Chapter 5 Test and Results

5.1 Test

The project presented in this thesis is been focalized on the creation of an application by which create a virtual set in augmented reality, for the pre-production phase of TV production. The application has been developed to be as simplest as possible, although the input system of the device could be not so easy to learn for new users. So, to understand if the usability of the application has resulted sufficient, it has been prepared a test.

The application usability has been tested by about twenty people, half women and men aged between 18 and 30, all with little or no confidence with HoloLens.

After a brief explanation of the technology and of the interface, testers have been assigned the same task.

The task is been to use the application, feeling free to use gestures and voice, following the instruction in the menu, try to build a simple scene on

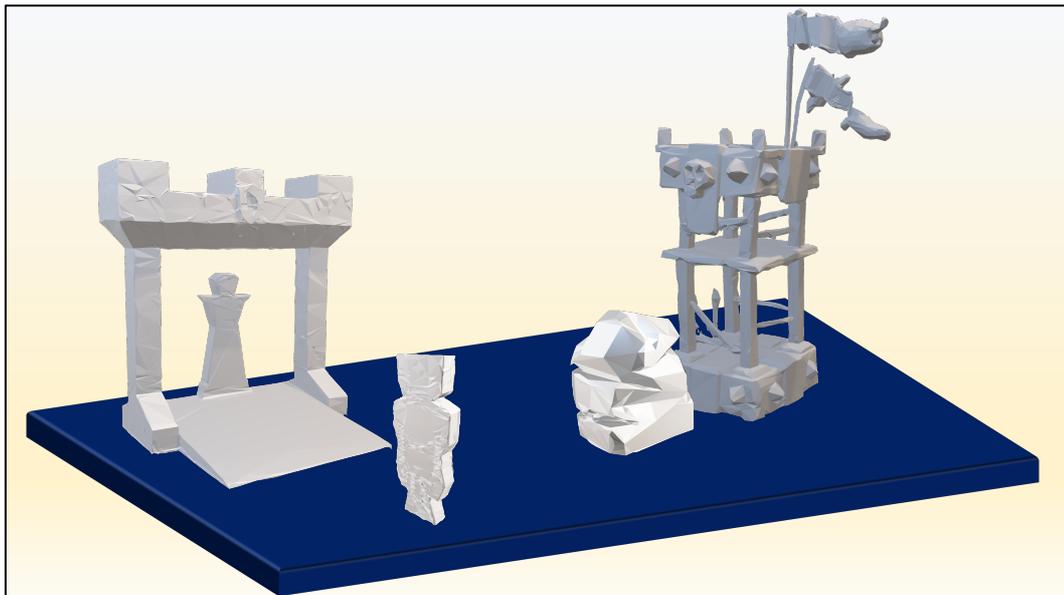


Figure 41 Test Scene

top of a table, similar to the one in Figure 41, and finally take a shot of the produced scene.

The test is a usability test, mainly linked to the understanding of the interface and to the features of placing and adjusting holograms in the environment.

At the end, testers completed a standard questionnaire, called SUS (System Usability Scale), that consists of ten questions and an answer scale from 1 (totally disagree) to 5 (totally agree), with a standard score.

5.2 Results

Each tester has been able to complete the task, producing a scene close enough to the original one in about 5-8 minutes, including the whole experience. The scenes produced looks like in Figure 42.



Figure 42 Produced Scene

The SUS questionnaire results are reported in Figure 43 and average score is 87.6.

Based on research, a SUS score above a 68 would be considered above average, so the usability of the application has resulted sufficient in this test.

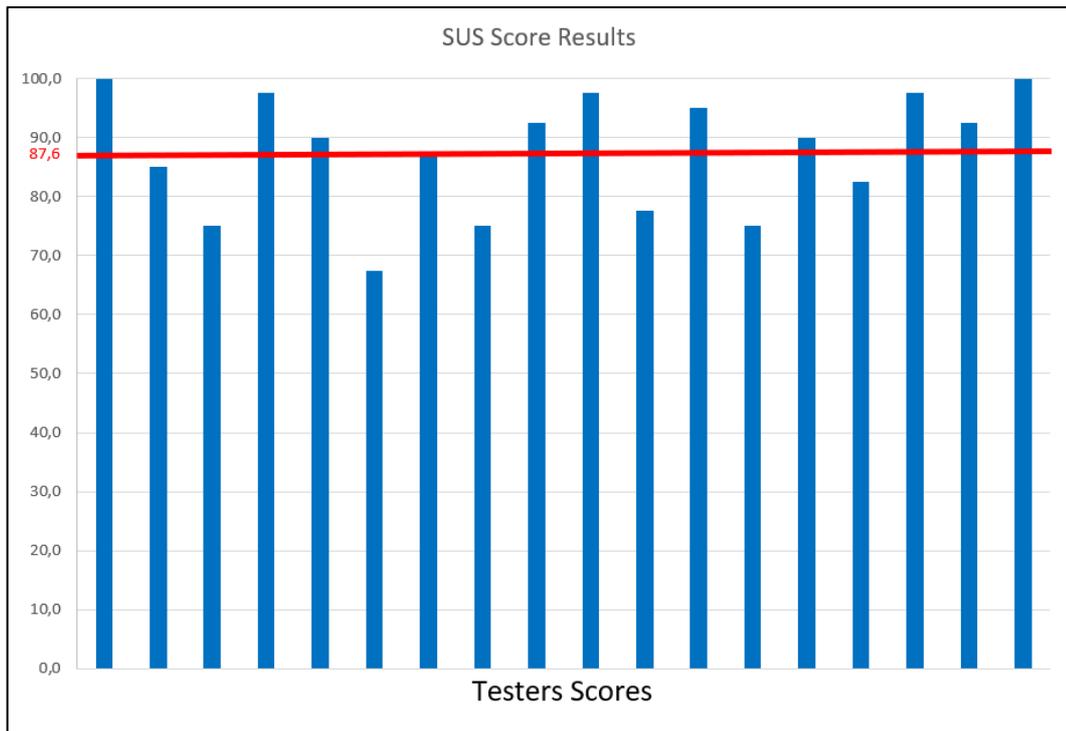


Figure 43 SUS Scores

Furthermore, testers expressed their preference about the types of input, as reported in Figure 44.

The majority preferred the gestures, because the feedback of voice commands has resulted a bit slowly, and not always the device recognize the voice commands, so it is necessary to repeat.

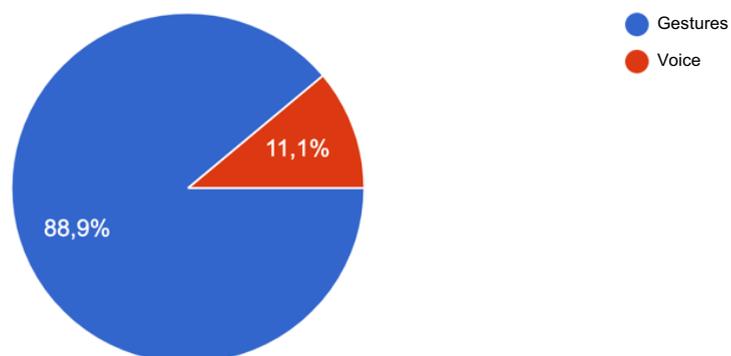


Figure 44 Gestures & Voice preference

After few minutes of practice, the majority of users familiarized with the gesture system, being able to use the user interface.

5.3 Future developments

One of the most important limitation of HoloLens is the small field of view, only 30×17 degrees, because of the technology limit of the hardware. Many users, being unfamiliar with the device, can't see objects just because they did not look in the correct direction.

Figure 45 show a comparison between the field of view of HoloLens 1, HoloLens 2 and Magic Leap One. The last new devices offer a field of view pretty wider. Using these devices to develop a similar application could improve the user experience in this sense.

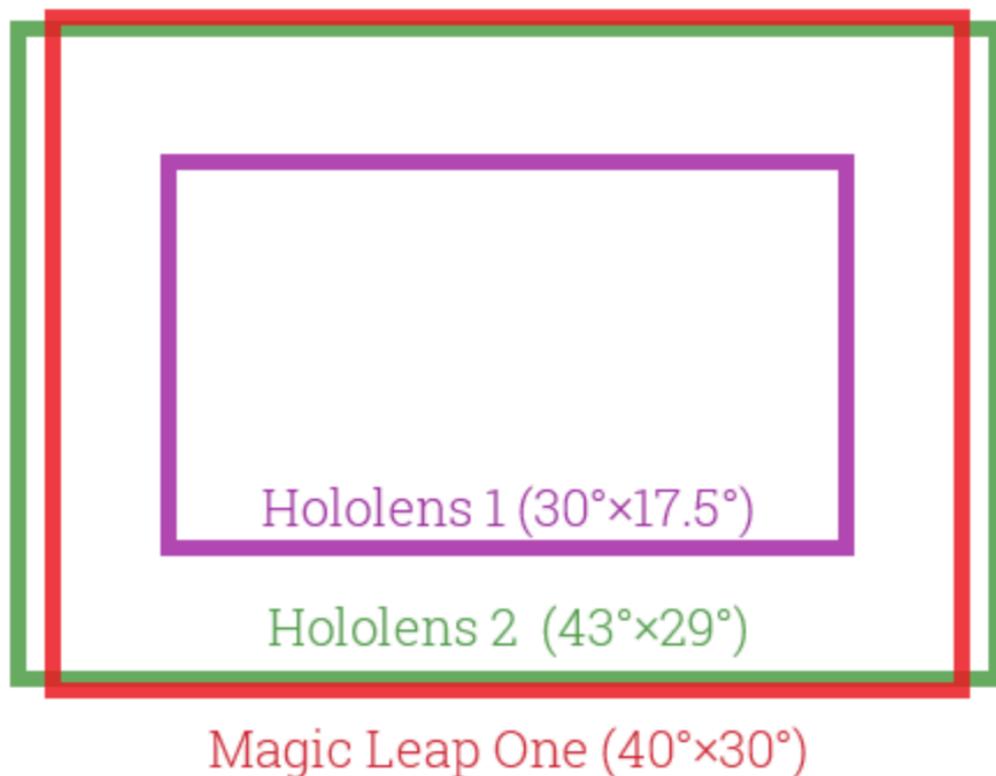


Figure 45 FOV Comparison

The recognition of planes is been sufficiently precise, but only in a condition of good and diffuse illumination, becoming less precise with less illuminated rooms or dealing with reflective surfaces. These performances too are limited by hardware.

HoloLens 2 offers the feature called “Articulated Hands”, a system of hand and fingers tracking, which mirrors up to 25 joints in the human hand, giving the chance to develop a new type of interaction, more natural and intuitive than the gestures. A feature like this could improve a lot the usability and the user experience of an application like the one developed in this work.

Conclusion

The development of the project introduced in this thesis is been focalized on the creation of a mixed reality application, with a novel interface, to support scenographers and set directors in TV studio production. The work proposes an alternative new way to create a virtual set in the phase of previsualization, offering a collaborative platform for more users.

The requirements given by the scenographers had been accomplished, although the hardware limited some aspects of the user experience, from the input methods to the precision of environmental awareness.

From the results and the comments given by the testers, the application has anyway a good usability. Overall, users liked the application and the mixed reality experience.

Since the results are good, RAI will continue the development of this project. At first, the application will be merged with the one developed parallely within the same project, with the features of loading of 3D models and textures from a remote DB, and with the search system and the interface to find models through tags and properties of 3D models.

With new developments of libraries and with new head mounted displays, with improved features, will be possible to develop this type of application, hopefully with having less hardware limitations.

References

- [1] F. D. Pace, Augmented Reality to Assist Television Production.
- [2] H. Zettl, Television Production Handbook.
- [3] "Viz Virtual Studio," Vizrt, [Online]. Available: www.vizrt.com/products/viz-virtual-studio.
- [4] O. E. Dictionary. [Online]. Available: http://www.etymonline.com/index.php?allowed_in_frame=0&search=virtual.
- [5] F. K. Paul Milgram, A Taxonomy of Mixed Reality Visual Displays.
- [6] "Microsoft, Mixed Reality," [Online]. Available: <https://docs.microsoft.com/en-us/windows/mixed-reality/mixed-reality>.
- [7] "HoloLens Docs, Hardware," [Online]. Available: <https://docs.microsoft.com/en-us/hololens/hololens1-hardware>.
- [8] "Magic Leap," [Online]. Available: <https://www.magicleap.com/magic-leap-one>.
- [9] "Vuforia," [Online]. Available: <https://www.ptc.com/en/products/augmented-reality/vuforia>.
- [10] "Vuforia Object Recognition," [Online]. Available: <https://library.vuforia.com/content/vuforia-library/en/articles/Training/Object-Recognition.html>.
- [11] "EasyAR," [Online]. Available: <https://www.easyar.com/>.
- [12] "ARCore," [Online]. Available: <https://developers.google.com/ar>.
- [13] "ARKit," [Online]. Available: <https://developer.apple.com/augmented-reality/arkit/>.
- [14] X.-b. F. Z.-w. S. M. X. S. X. X.-h. W. Z.-w. Y. Hong-zhi Hu, "Application and Prospect of Mixed Reality Technology in Medical Field," *Current Medical Science*, 2019.
- [15] "Quiver Education App," [Online]. Available: <http://www.quivervision.com/apps/quiver-education/>.

- [16] "Story of the forest," [Online]. Available:
<https://www.nationalmuseum.sg/our-exhibitions/exhibition-list/story-of-the-forest>.
- [17] "Skin and Bone," [Online]. Available:
<https://naturalhistory.si.edu/exhibits/bone-hall>.
- [18] "Petersen Automotive Museum, Hollywood Dream Machine Exhibit," [Online]. Available: <https://www.petersen.org/hollywood>.
- [19] R. I. A. M. M. F. M. A. S. D. Z. Federico Debandi, "Enhancing cultural tourism by a mixed reality application for outdoor navigation and information browsing using immersive devices," 2018.
- [20] S. F. Steven Henderson, "Exploring the Benefits of Augmented Reality Documentation for Maintenance and Repair," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, 2011.
- [21] "ARMEISTERBOCK," [Online]. Available:
<https://www.illogic.xyz/case/armeisterbock/>.
- [22] "Pokémon Go," [Online]. Available: <https://www.pokemongo.com/>.
- [23] "US Naval Research Laboratory - BARS," [Online]. Available:
<https://www.nrl.navy.mil/itd/imda/research/5581/augmented-reality>.
- [24] F.-Y. C. Chia-Chi Mao, "Augmented Reality and 3-D Visualization Effects to Enhance Battlefield Situational Awareness".
- [25] "SketchUp Viewer," [Online]. Available:
<https://www.sketchup.com/products/sketchup-viewer>.
- [26] "Kubity," [Online]. Available: <https://pro.kubity.com/>.
- [27] F. X. I. M. Michael Shyu, "A New Design Paradigm in Mixed Reality: Using HoloLens for Architectural Design".
- [28] "FrameForger Storyboard Studio," [Online]. Available:
www.frameforge.com/.
- [29] "eStudio," Brainstorm, [Online]. Available:
<http://www.brainstorm3d.com/products/estudio/>.
- [30] "Ncam," [Online]. Available: <https://www.ncam-tech.com/>.
- [31] "SolidTrack," [Online]. Available: <http://www.solid-track.com/>.

- [32] "Tom's Hardware, Microsoft HoloLens Components," [Online]. Available: <https://www.tomshardware.com/news/microsoft-hololens-components-hpu-28nm,32546.html>.
- [33] "Microsoft Docs, Gestures," [Online]. Available: <https://docs.microsoft.com/it-it/windows/mixed-reality/gestures>.
- [34] "HoloLens Docs, Basic Usage," [Online]. Available: <https://docs.microsoft.com/en-us/hololens/hololens1-basic-usage#use-hololens-with-your-hands>.
- [35] "HoloLens 2 Documentation," [Online]. Available: <https://docs.microsoft.com/en-us/hololens/hololens2-hardware>.
- [36] "Artec Eva," [Online]. Available: <https://www.artec3d.com/portable-3d-scanners/artec-eva>.
- [37] "Blender," [Online]. Available: <https://www.blender.org/>.
- [38] "Unity 3D Official Manual," [Online]. Available: <https://docs.unity3d.com/2017.4/Documentation/Manual/index.html>.
- [39] S. Ong, Beginning Windows Mixed Reality Programming, 2017.
- [40] "RAI CRIT," [Online]. Available: <http://www.crit.rai.it/>.
- [41] A. Studio. [Online]. Available: <https://www.artec3d.com/3d-software/artec-studio>.

List of Figures

<i>Figure 1 Sketch of a RAI TV set</i>	10
<i>Figure 2 A television set before/after the mixing.</i>	12
<i>Figure 3 Virtuality Continuum</i>	17
<i>Figure 4 Meta 2</i>	20
<i>Figure 5 Magic Leap One</i>	21
<i>Figure 6 Epson Moverio BT-300</i>	22
<i>Figure 7 Application of MR technique in preoperative discussion between doctors</i>	25
<i>Figure 8 MR application for outdoor navigation</i>	27
<i>Figure 9 Google Maps AR</i>	28
<i>Figure 10 ARMeisterbock</i>	29
<i>Figure 11 Pokémon Go</i>	30
<i>Figure 12 Storyboard sketch</i>	33
<i>Figure 13 FrameForge Storyboard Studio</i>	34
<i>Figure 14 TV Show Rendering & real shooting</i>	35
<i>Figure 15 Viz Virtual Studio components.</i>	36
<i>Figure 16 AR in a tv studio</i>	37
<i>Figure 17 Microsoft HoloLens</i>	39
<i>Figure 18 HoloLens Sensors</i>	40
<i>Figure 19 HoloLens Display System</i>	41
<i>Figure 20 Bloom Gesture</i>	42
<i>Figure 21 Air Tap Gesture</i>	43
<i>Figure 22 Microsoft HoloLens 2</i>	44
<i>Figure 23 High to Low Poly in Blender</i>	46
<i>Figure 24 Unity Editor</i>	48
<i>Figure 25 Game Object</i>	49
<i>Figure 26 Sharing Service Client-Server System</i>	56
<i>Figure 27 HoloToolkit UI controls</i>	58

<i>Figure 28 3D objects interaction UI</i>	58
<i>Figure 29 Rob-O-Cod Props</i>	60
<i>Figure 30 Contents creation workflow</i>	61
<i>Figure 31 Menu appearance</i>	63
<i>Figure 32 Spatial Mapping</i>	64
<i>Figure 33 Surface Plane</i>	65
<i>Figure 34 Spatial Processing System</i>	66
<i>Figure 35 Placing System</i>	67
<i>Figure 36 Adjustment UI</i>	68
<i>Figure 37 Connection Numpad</i>	69
<i>Figure 38 Sharing Experience</i>	70
<i>Figure 39 Sharing System Scheme</i>	71
<i>Figure 40 Snapshot taken by the application</i>	72
<i>Figure 41 Test Scene</i>	73
<i>Figure 42 Produced Scene</i>	74
<i>Figure 43 SUS Scores</i>	75
<i>Figure 44 Gestures & Voice preference</i>	75
<i>Figure 45 FOV Comparison</i>	76