SAPIENZA
UNIVERSITÀ DI ROMA

Ingegneria dell'informazione, informatica e statistica

Human Computer Interaction on the Web

JUNGLE BOOK

Presented By:

Mojtaba Taj (1993339)

Submitted To:

Professor: Emanuele Panizzi

Contents

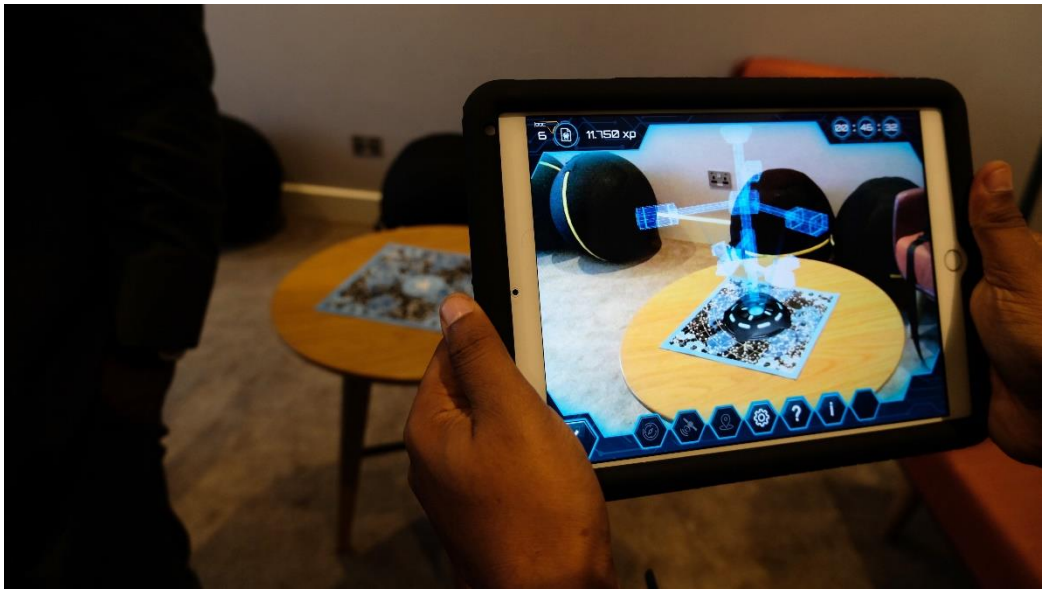
Introduction

Types of Extended Reality (XR)

Augmented Reality (AR): designed to add digital elements over real-world views with limited interaction.
Virtual Reality (VR): immersive experiences helping to isolate users from the real world, usually via a headset device and headphones designed for such activities.
Mixed Reality (MR): combining AR and VR elements so that digital objects can interact with the real world, means businesses can design elements anchored within a real environment.

1 Augmented Reality

Augmented reality is an enhanced, interactive version of a real-world environment achieved through digital visual elements, sounds, and other sensory stimuli via holographic technology. AR incorporates three features: a combination of digital and physical worlds, interactions made in real time, and accurate 3D identification of virtual and real objects. Augmented reality offers a better way to design, curate, and deliver consumable instructions by overlaying digital content in real-world work environments. When a business understands what AR is and how to utilize it successfully, everyone can work remotely while collaborating efficiently.



1.2 Types of Augmented Reality

In deciding which type of AR technology you'll need for your business, you'll first have to determine what kind of AR to use. There are two types of augmented reality: marker-based and marker-less. Choosing one of these types of AR will determine how you'll be able to display your images and information.

1.2.1 Marker-Based AR: is created using image recognition to identify objects already programmed into your AR device or application. When placing objects in view as points of reference, they can help your AR device determine the position and orientation of the camera. This is generally achieved by switching your camera to grayscale and detecting a marker to compare that marker with all the others in its information bank. Once your device finds a match, it uses that data to mathematically determine the pose and place the AR image in the right spot.

1.2.3 Marker-Less AR: is more complex as there's no point in which your device will focus on. Because of this, your device must recognize items as they appear in view. Using a recognition algorithm, the device will look for colors, patterns, and similar features to determine what that object is and then, using time, accelerometer, GPS, and compass information, it will or orient itself and use a camera to overlay an image of whatever you'd like within your real-world surroundings.

1.3 Types of Marker-less Augmented Reality

1.3.1 Location-Based AR: Location-based augmented reality is the technology that doesn't require any physical markers or identifiers to work. Unlike the marker-based augmented reality brands in merchandise and advertising, which typically connects to the products or shop locations, location-based technology doesn't require such connection. This technology is also called marker-less, position-based, or geo-based augmented reality, and it is bound to the GPS that gives the users' location. Here you can see an example of location-based AR in Fig.1.

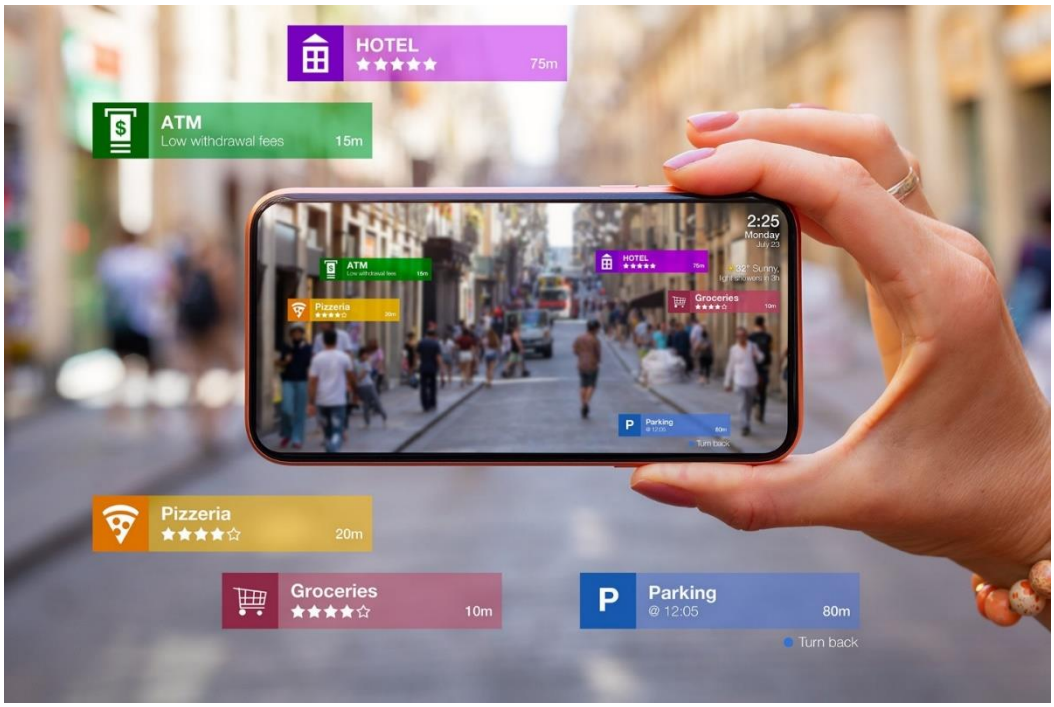


Fig.1 Location-Based AR

1.3.2 Projection-Based AR: Also known as Spatial Augmented Reality (SAR) or Projection Mapping, is a type of augmented reality that uses projectors to overlay digital content onto physical objects or surfaces in the real world. It involves the projection of computer-generated visuals onto real-world surfaces to create the illusion of interactive and dynamic augmented environments. Here you can see an example of projection-based AR in Fig.2.

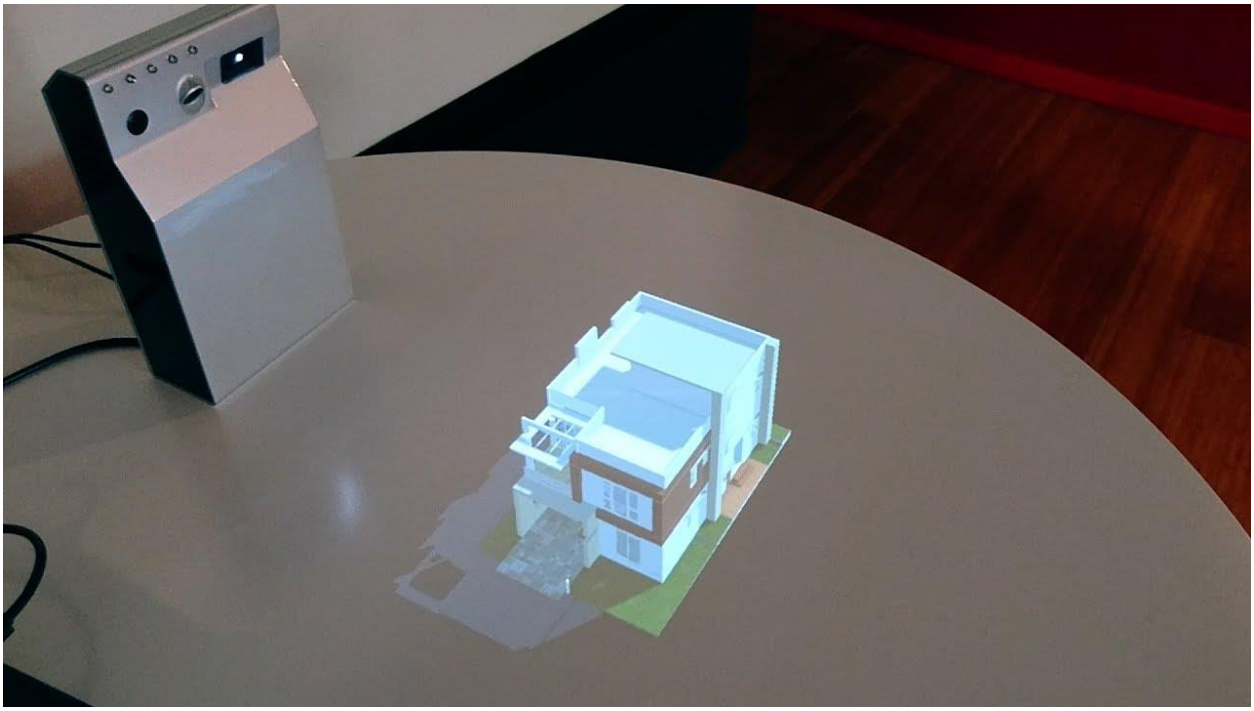


Fig.2 Projection-Based AR

1.3.3 Super Imposition Based AR: Superimposition based AR provides an 'alternate' view of the object in concern, either by replacing the entire view with an augmented view of the object or by replacing a portion of the object view with an augmented view. In this case, once again, object recognition plays a vital role. As instance, Doctors can use the technology to examine the patient from various angles in real-time. A live feed from an X-Ray machine can be used to superimpose the X-Ray view of the patient's body part on the real image to provide better understanding of the damage to bones. The application can be made to work via a head mounted display or special goggles. In other uses, the view can be shown on a screen where the video feed is taken from a real camera and X-Ray vision can be imposed on it. Here you can see an example of location-based AR in Fig.3.

Fig.3 Super Imposition Based AR

2 Virtual Reality

Virtual Reality (VR) is a computer-generated environment with scenes and objects that appear to be real, making the user feel they are immersed in their surroundings. This environment is perceived through a device known as a Virtual Reality headset or helmet. VR allows us to immerse ourselves in video games as if we were one of the characters, learn how to perform heart surgery or improve the quality of sports training to maximize performance. Although this may seem extremely futuristic, its origins are not as recent as we might think. In fact, many people consider that one of the first Virtual Reality devices was called Sensorama, a machine with a built-in seat that played 3D movies, gave off odours and generated vibrations to make the experience as vivid as possible. The invention dates back as far as the mid-1950s.

Virtual Reality

3 Difference between Augmented Reality and Virtual Reality

The main difference between the two is that VR builds the world in which we immerse ourselves through a specific headset. It is fully immersive and everything we see is part of an environment artificially constructed through images, sounds, etc. On the other hand, in augmented reality (AR), our own world becomes the framework within which objects, images or similar are placed. Everything we see is in a real environment and it may not be strictly necessary to wear a headset. The clearest and most mainstream example of this concept is Pokémon Go.

4 Main Goal

The application is (Marker-Based AR) which is designed for using image recognition to identify objects already programmed into your AR device or application. For this purpose, I have considered wild animals such as (Lion, Tiger, Leopard and Wolf) which live in the jungle, so the name of the application is Jungle Book. This application provides people especially children with 3D models of wild animals. Children at school learn about the jungle and animals from books and pictures but they are just images. The first and most important goal, is getting children familiar with how animals look like by using AR technology in a tangible way and making them seem real. In addition, technology is growing fast, so it is better to adapt ourselves to it and exploit it in our education systems.

5 Requirements

For designing and implementing the project we have to consider some aspects:

5.1 Development Engine: If you have any interest in game development, then learning Unity should be your top priority. Unity is the tool used by a large number of game developers to create and power their creations. Unity software is powerful, extremely easy to use, and free until you start making the big bucks. As a game engine, Unity is able to provide many of the most important built-in features that make a game work. That means things like physics, 3D rendering, and collision detection. From a developer's perspective, this means that there is no need to reinvent the wheel. Rather than starting a new project by creating a new physics engine from scratch–calculating every last movement of each material, or the way light should bounce off of different surfaces. What makes Unity even more powerful though, is that it also includes a thriving "Asset Store." This is essentially a place where developers can upload their creations and make them available to the community. As well as a game engine, Unity is an IDE. IDE stands for "integrated development environment," which describes an interface that gives you access to all the tools you need for development in one place. The Unity software has a visual editor that allows creators to simply drag and drop elements into scenes and then manipulate their properties.

5.2 Programming Language: Unity primarily uses the C# coding language for game development. C# is a modern, object-oriented programming language developed by Microsoft. It is widely known for its simplicity, versatility, and robustness. Unity's choice to use C# as its main programming language offers several advantages. C# provides a high-level and expressive syntax that allows developers to write clean and maintainable code. It offers strong typing, automatic memory management through garbage collection, and a comprehensive standard library. These features make C# a suitable language for building complex and performance-intensive applications like games. With C# in Unity, developers can take advantage of the extensive Unity scripting system to manipulate objects, handle input, create user interfaces, implement game mechanics, and more. Unity's API is well-documented, making it easier for developers to learn and utilize the available features. So, if you want to develop games with Unity, then C# is the only language you will need.

5.3 Software Development Kit: Vuforia is a cross-platform Augmented Reality (AR) and Mixed Reality (MR) application development platform, with robust tracking and performance on a variety of hardware (including mobile devices and mixed reality Head Mounted Displays (HMD) such as the Microsoft HoloLens). Unity's integration of Vuforia allows you to create vision apps and games for Android and iOS using a drag-and-drop authoring workflow.

Vuforia supports many third-party devices (such as AR/MR glasses), and VR devices with back-facing cameras (such as the Gear VR). You can use any device with a camera to test AR/MR games and applications built in Unity with Vuforia.

6 Implementation Phase

This phase has some aspects which are crucial to implementing the project. As I mentioned, the main goal of the project is to create marker-based AR model in such a way that it is going to be real through our devices.

6.1 Tracking: Tracking in AR finds for a specific pattern or image that an AR APP can recognize. Once the app finds the pattern, it constantly tracks the position of the pattern in real world space so that the app can accurately place a digital object onto the marker that is being tracked.

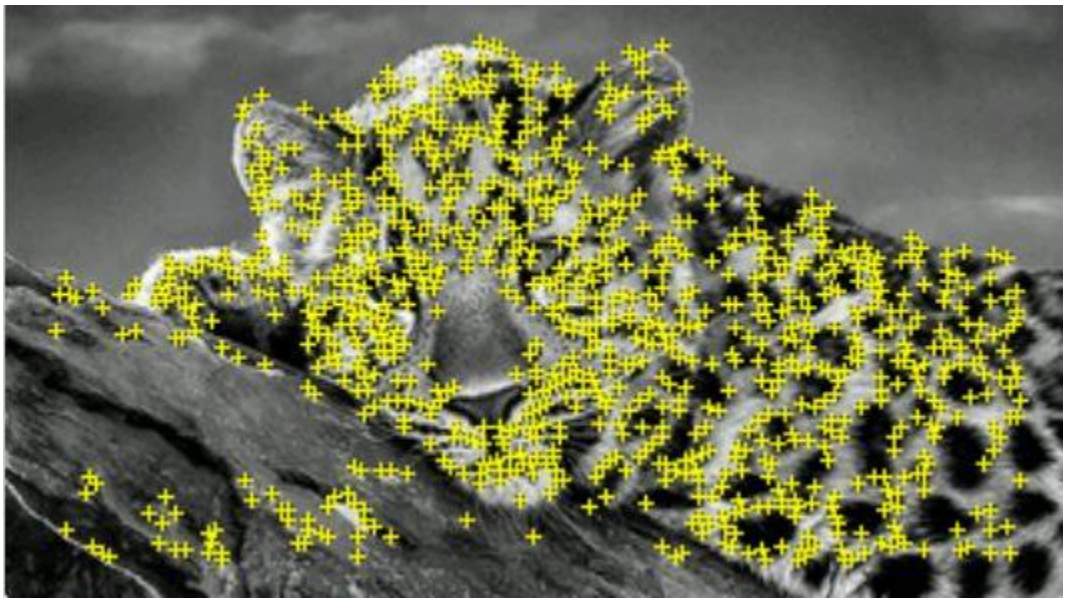


Fig.4 Leopard Image

As you can see there are multiple plus in Fig.4. These are feature points of the image that help us tracking markers.

The tracking process involves a few fundamental steps, which are:

6.1.1 Marker detection: The AR system's camera scans the environment for markers. These markers are usually simple, high-contrast patterns that can be easily recognized.

6.1.2 Pose estimation: Once a marker is detected, the AR system calculates its position and orientation in relation to the camera.

6.1.3 Content overlay: Based on the marker's pose, the AR system overlays digital content onto the marker in real-time. This content could be 3D models, videos, animations, or textual information.

7 AR System Architecture

System Architecture consists of 3 phases. The most significant phase is the middle phase. Its components are the Capturing Module, Tracking Module, Virtual Component and Rendering Module. Fig.5 shows the architecture of how AR is implemented.

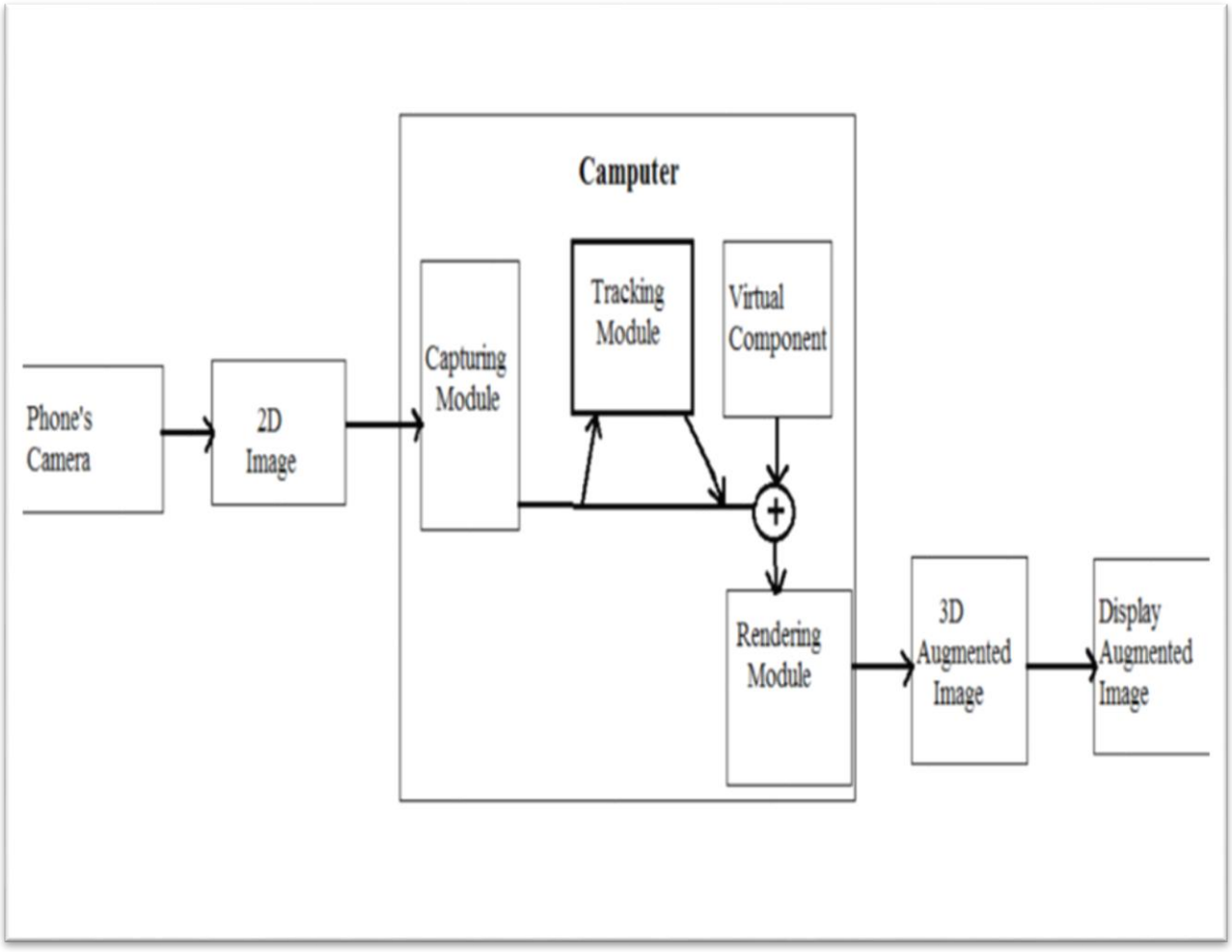


Fig.5 AR System Architecture

7.1 Capturing Module:

Captures the image from the camera.

7.2 Tracking Module:

The tracking module is the core of the AR system. It calculates the relative pose of the camera in real time. The term “POSE” means basically the 6 degrees of freedom that is the 3D location and orientation of an object.

7.3 Rendering Module:

The rendering module helps us to combine the virtual components and real image into one using the calculated pose and after all these it renders the augmented digital object on the image.

7.4 Virtual Component:

The virtual component augmented is usually interactive in nature. This gives the user a feeling that the object is present in the view even though it is physically absent

Pros:

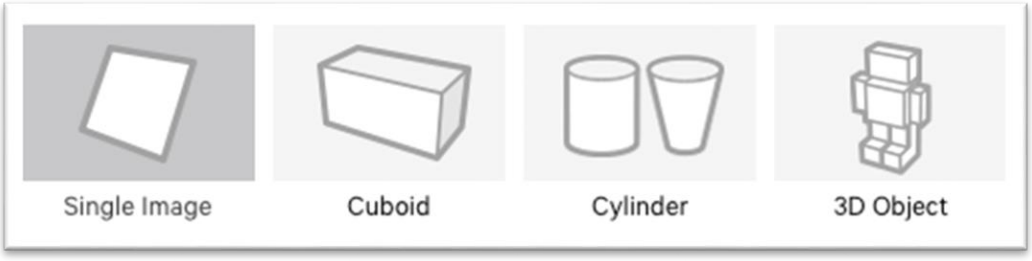
Simple and user friendly for first time AR users
Tracking is very stable with marker based AR
Minimum production cost

Cons:

Only works when the camera is near the marker
If the marker image has light reflecting, it causes issues with the AR overlay
If there are not strong contrasts between colors the tracking will be unstable
The app relies on the existence of a separate asset such as marketing material to launch the experience. While marker-less AR can be used anytime and anywhere.

8 Vuforia SDK

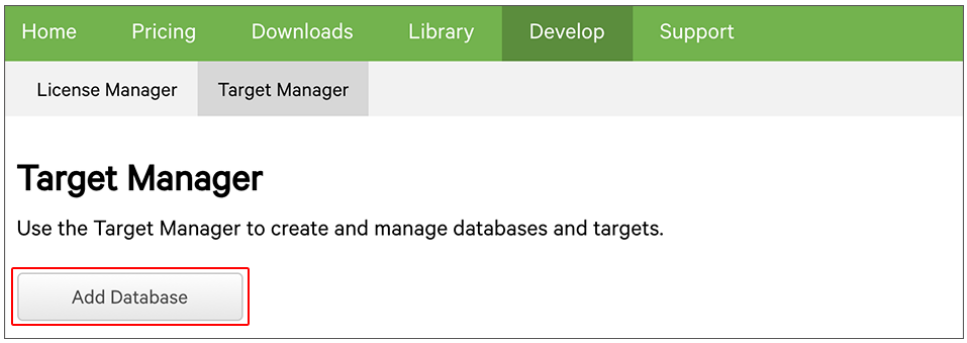
Vuforia Engine is a software development kit (SDK) for creating Augmented Reality apps. With the SDK, you add advanced computer vision functionality to your application, allowing it to recognize images, objects, and spaces with intuitive options to configure your app to interact with the real world. Vuforia Engine supports AR app development for Android, iOS, Lumin, and UWP devices. In AR, markers are images or objects registered with the application which act as information triggers in your application. When your device's camera recognizes these markers in the real world (while running an AR application), this triggers the display of virtual content over the world position of the marker in the camera view. **Marker-based tracking** can use a variety of different marker types, including QR codes, physical reflective markers, image Targets and 2D tags. The simplest and most common type of marker in game applications is an Image Target.



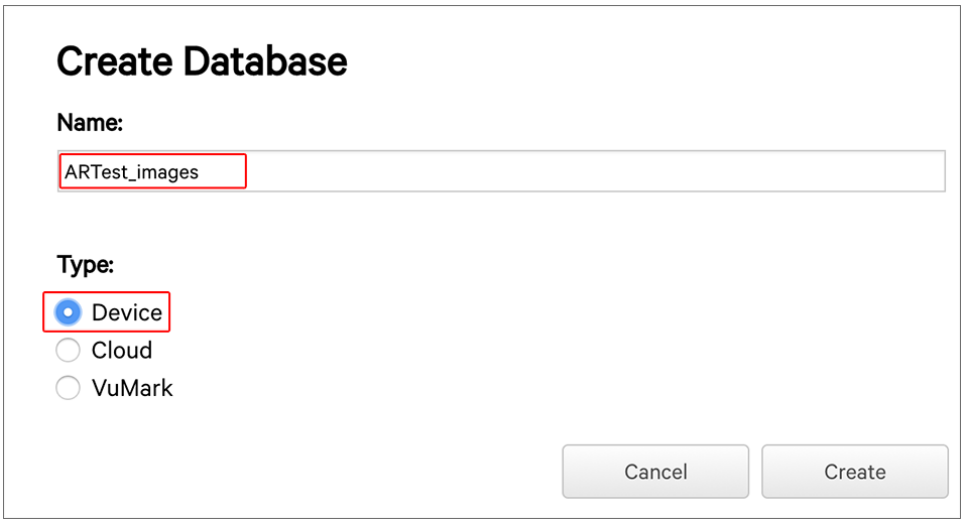
Common Image Target Types

8.1 Image Targets: Image Targets are a specific type of marker used in Marker-based tracking. They are images you manually register with the application, and act as triggers that display virtual content. For Image Targets, use images containing distinct shapes with complex outlines. This makes it easier for image recognition and tracking algorithms to recognize them.

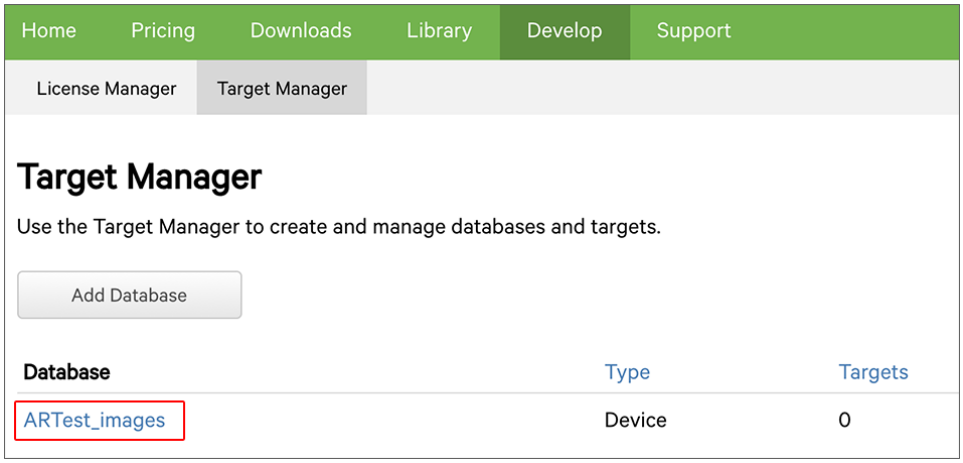
8.2 Setting up Image Targets: To allow your application to recognize images and use them as Targets to trigger gameplay, display graphics or information, you need to create a Target database. You can create Target databases directly from the Vuforia Developer Portal.



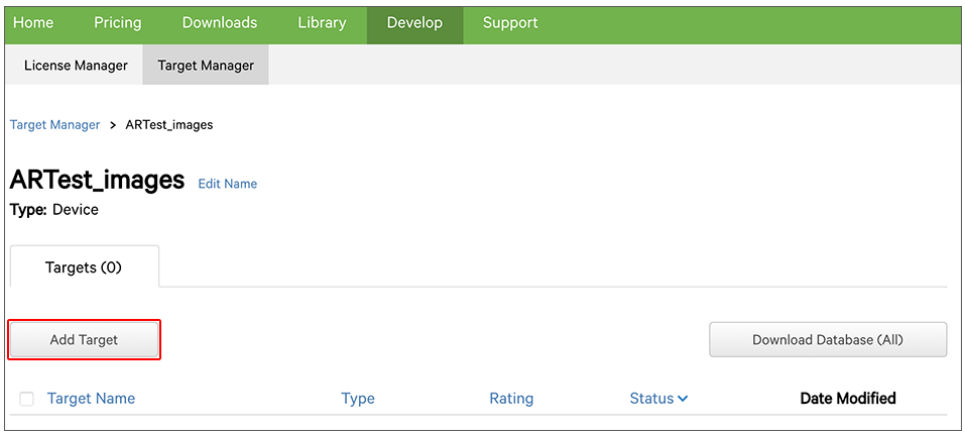
Creating a New Target Database



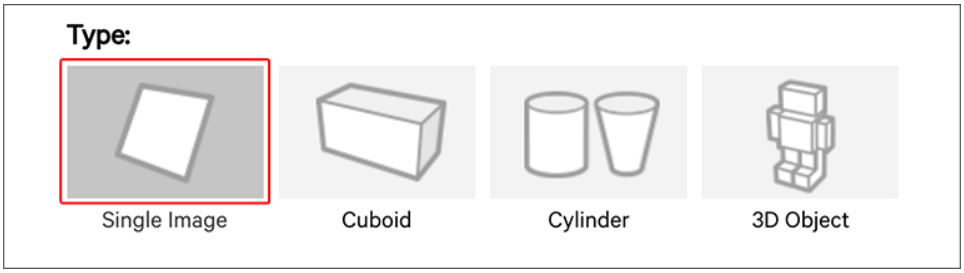
Managing the New Target Database



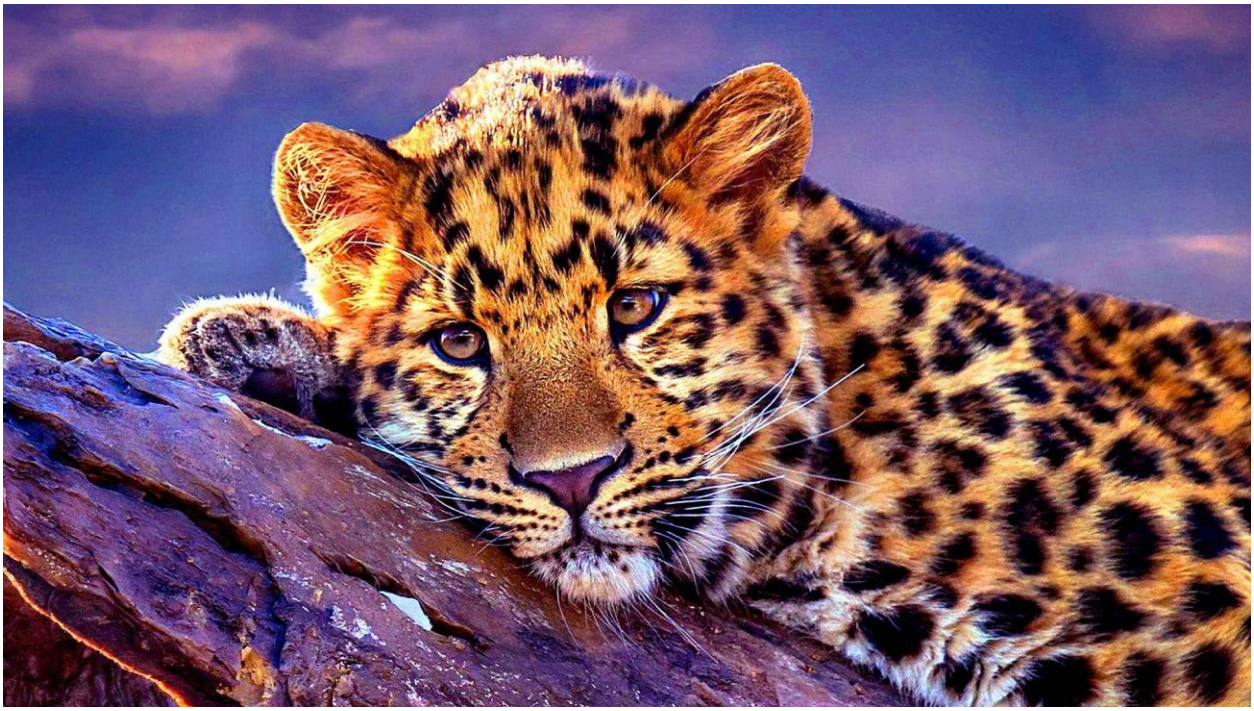
Open Add Target Window



Choosing Target Type

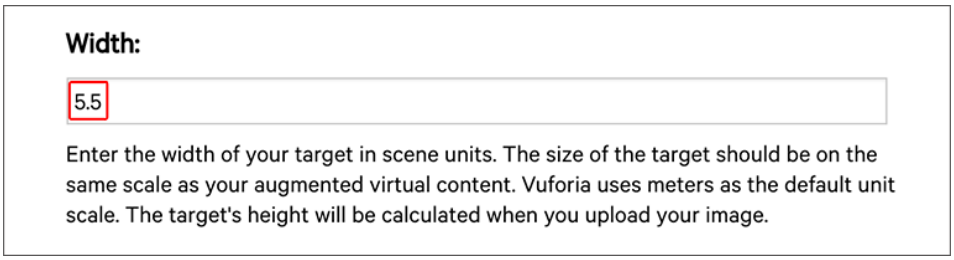


Example of Image Target Used



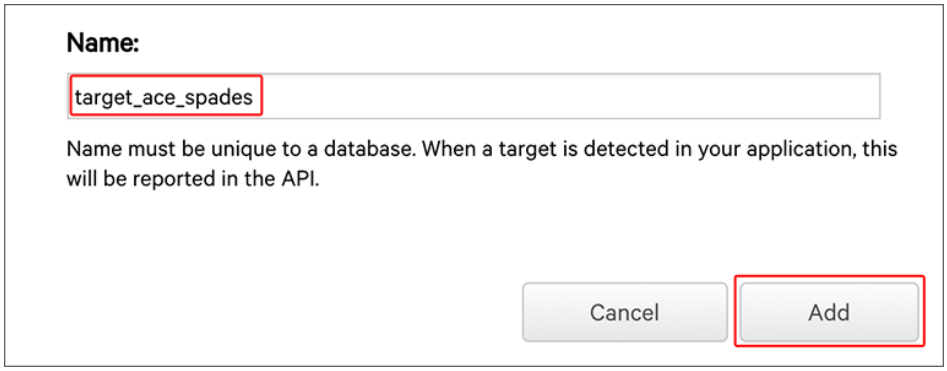
Choosing Target Image

Setting a Target Width

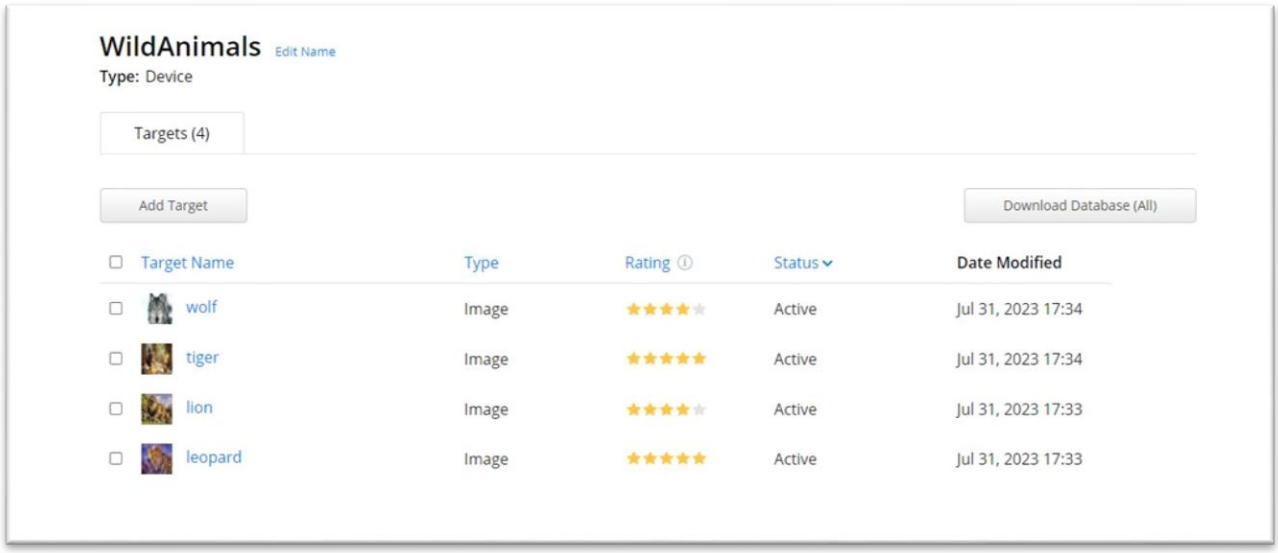


The Width value is a scale value that you need to set to the size you want the image to appear in your Unity Scene (in real-world units). Unity measures everything in your Scene in relation to the size of your Target image. The width of the **Leopard** is 6 cm, so you would use 6 cm as the Width value. If you need a larger size Target, then increase this Width.

Naming the Target and Adding it to the Database

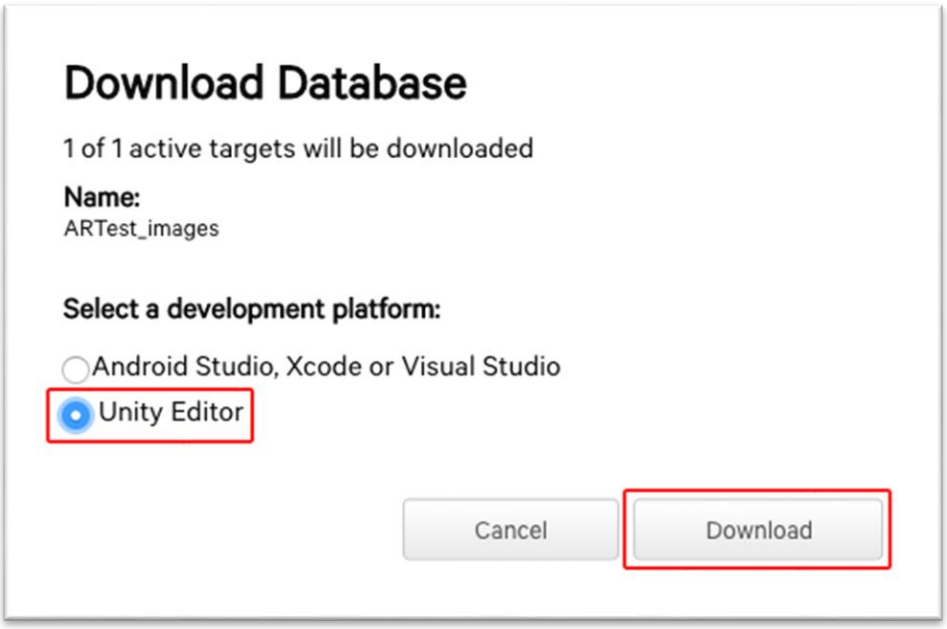


Downloading the Target Database and Target Quality Rating



NOTE: Use any image, but make sure that the image has enough detail to be rated as a **5-star** or **4-star** Target so that the camera can easily track it.

Downloading Database Unity package



NOTE: On the Download Database window, under Select a development platform, select Unity Editor, then click the Download button. This downloads a Unity package of the Target database that you can save on your hard drive.

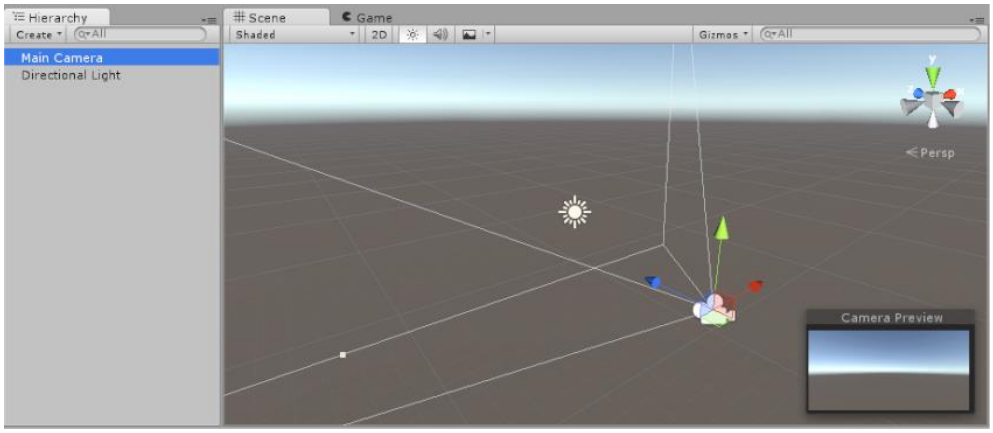
9 UNITY ENGINE

Unity is a cross-platform game engine developed by Unity Technologies, first released in 2005 as a Mac OS Game Engine. The engine was initially developed to create animations and interactive content for the web, but it quickly expanded to support a variety of platforms, including desktop computers, mobile devices, and game consoles. The Unity game engine has a strong developer community with many resources already developed, which makes it easy to get started and develop games. It provides the tools and features you need to bring your game ideas to life.

9.1 AR development in Unity: To place an object in the real world, you must first determine where to place for it. For example, you may want to place a virtual painting on a physical wall. If you place a virtual potted plant, you may want it on a physical table or the floor. An AR app receives information about the world from the user's device, such as the locations of planar surfaces, the detection of objects, people, faces, and so on; and must decide how to use this information to create a good experience for the user. When you open a typical AR scene in Unity, you will not find many 3D objects in the scene or the Hierarchy view. Instead, most GameObjects in the scene define the settings and logic of the app. 3D content is typically created as prefabs that are added to the scene at runtime in response to AR-related events.

9.2 Scene: Scenes are where you work with content in Unity. They are assets that contain all or part of a game or application. For example, you might build a simple game in a single scene, while for a more complex game, you might use one scene per level, each with its own environments, characters, obstacles, decorations, and UI .You can create any number of scenes in a project.

The default Unity sample scene, which contains a Main Camera and a directional Light



9.3 Markers: For the project, four animals are considered (Lion, Wolf, Tiger and Leopard). Before importing the 3D model into Unity we have to import markers. As we remember, we created a database in Vuforia and downloaded the database from Vuforia so at this level we have to import that database into Unity to launch our markers. The Lion marker is one of the markers in the database. Here you can take a look at Fig.6 and Fig.7

Fig.6 Lion Marker

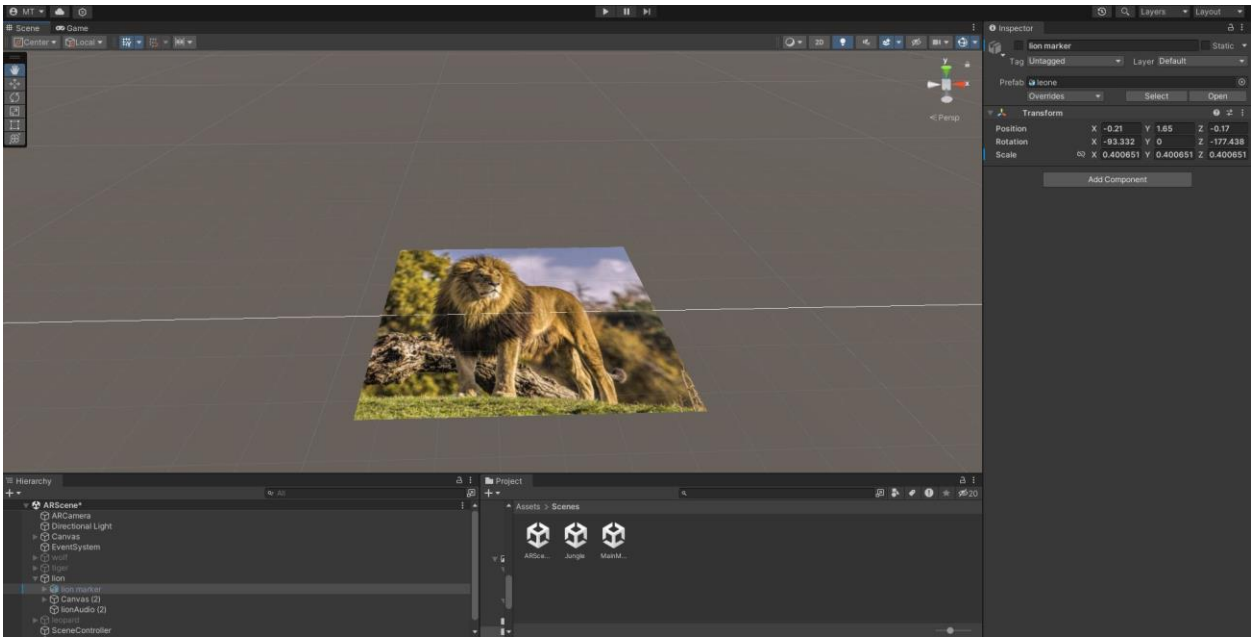


Fig.7 Lion Marker in Unity Environment

9.4 3D Model: It is time to import our 3D model into Unity. 3D model should be compatible with Unity so we have to take the format of the 3D model into consideration. The format which are compatible with Unity are .fbx, .dae, .obj, .dxf, .skp, and .stl

.OBJ (interchange) files are 3D data files that comprise only the geometry of a 3D model without any texture maps or other properties. If you export .obj from a 3D modeling software it will likely be accompanied by a .mtl file type which is the file that reserves information about the material ID's and UV mapping of an object. Fig.8 shows the 3D model of lion.

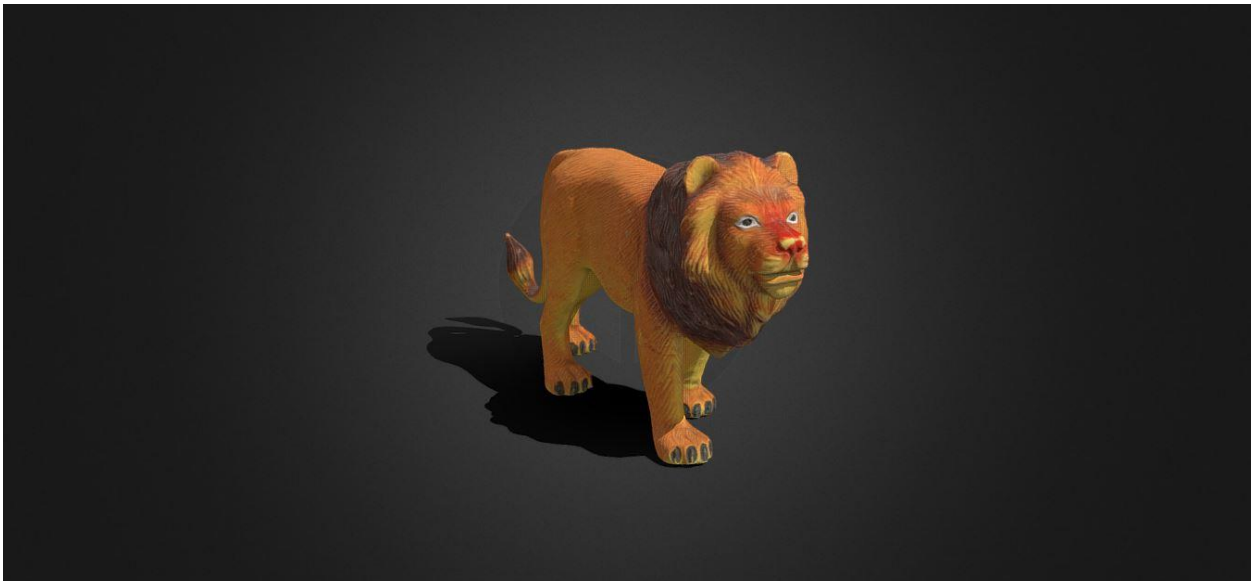
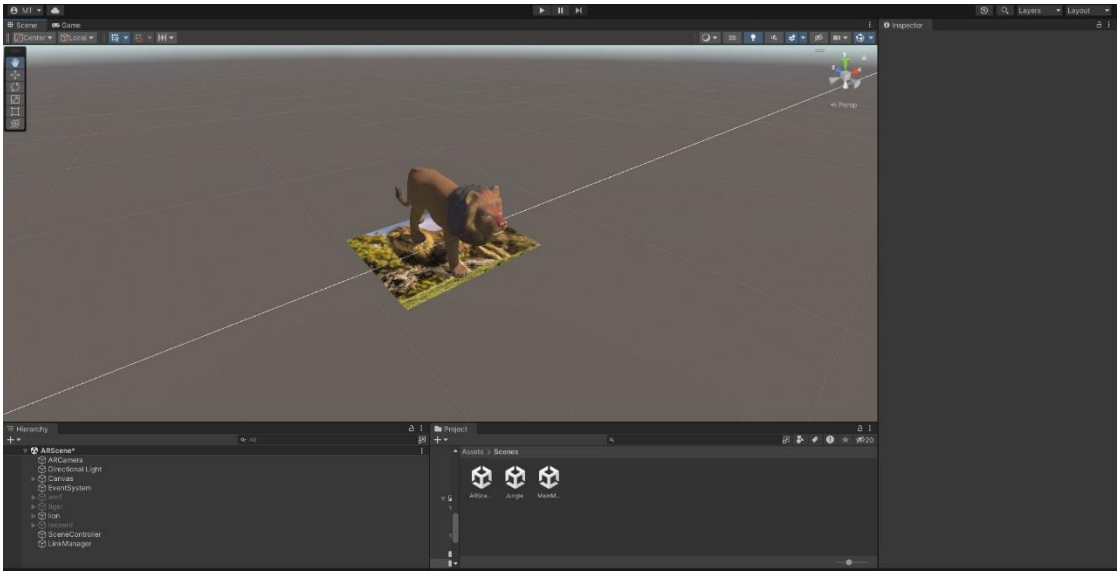


Fig.8 3D Model of Lion



3D Model of Lion in Unity

9.5 Text: The UI text element displays a non-interactive piece of text to the user. Text elements can be used to provide captions or labels for other GUI controls or to display instructions or other text. We can use text as a description or extra information in each scene.

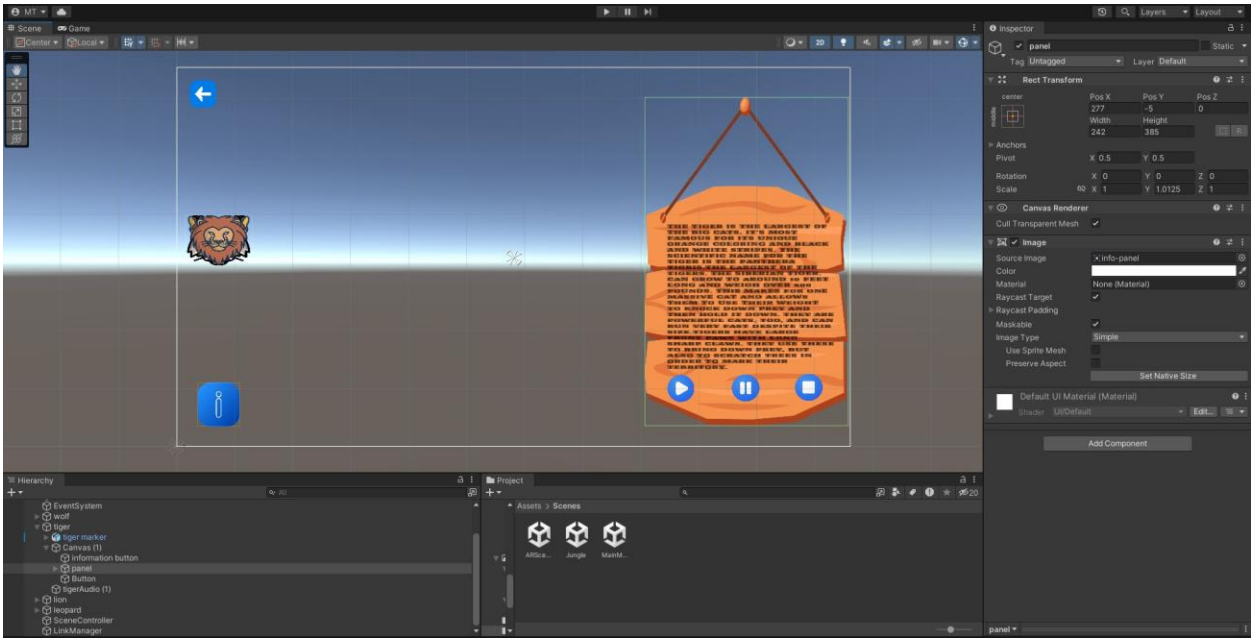
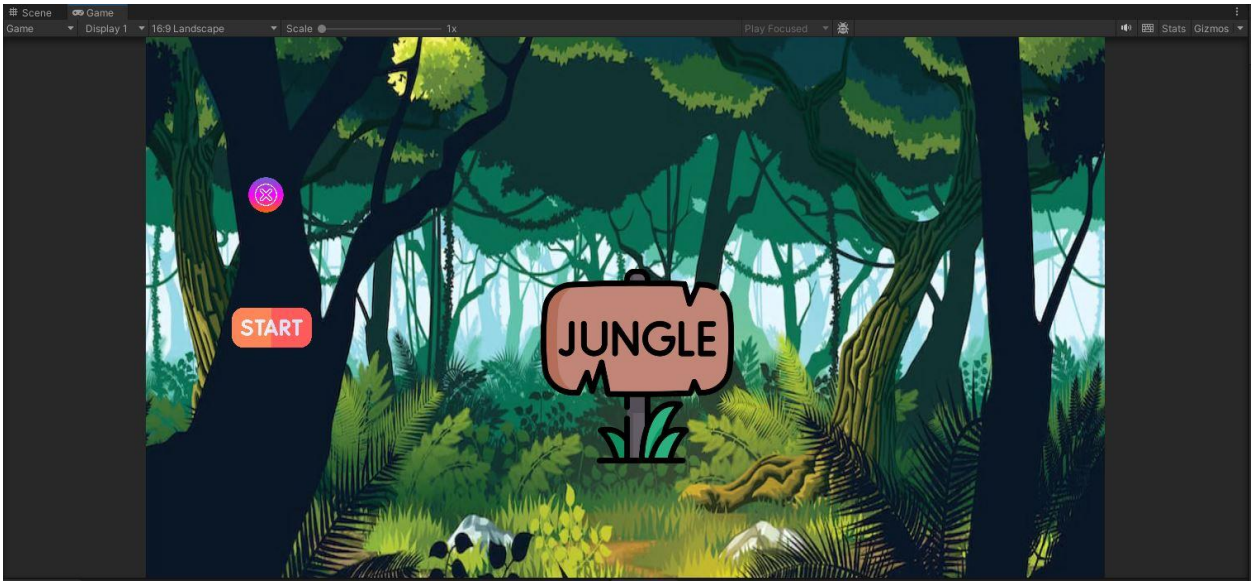


Fig.9 Text in Panel

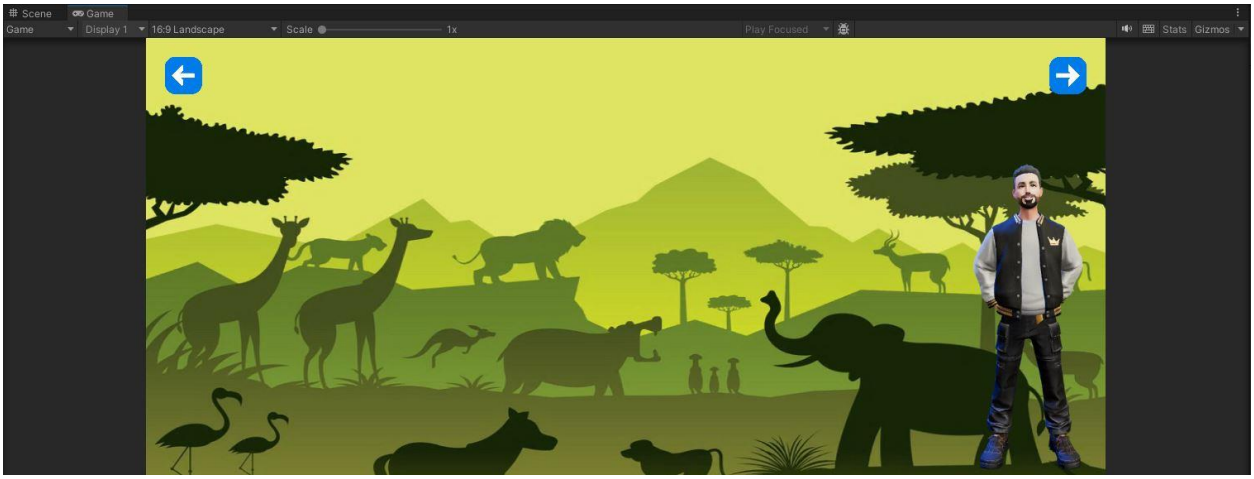
9.6 Sound and Audio: In Unity, the positioning of a source of audio is important to characterize its source. For example, we need to make sure that a gameObject playing the sound of a waterfall matches with an actual waterfall gameObject, and that the player's perception of that sound feels real. Getting louder as the player gets closer, making sure the audio panning varies as the relative position of the waterfall changes with respect to the player, and so on. The information about each animal has been accompanied by an audio clip that describes it specifically. You can play, pause and stop the audio.

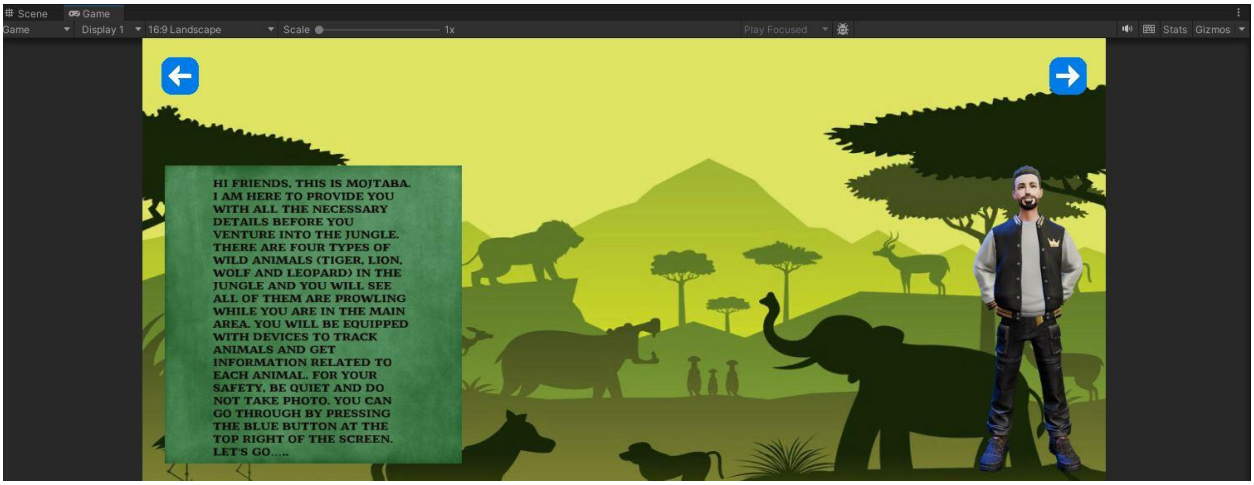
10 Game Play

The jungle book consists of three scenes. Each scene has its own info and components to affect as much as possible the user. The first scene is the starting point to venture through into the jungle. When we open the application, the beginning audio plays continuously. Here is the spot you can start your adventure. After the first scene, you will find yourself in the jungle. There is an avatar that plays the role of a tour leader to provide you with essential information before you start searching. Once you click on the avatar it will show you a table with information carved into it. The last scene shows wild animals but it is your duty to use your cellphone to see the animals. In addition, if you click on the animal head image you will be directed to the website for detailed information. All the procedures including of moving between scenes or playing the audio have been written in C#.



First Scene


Second Scene



Second Scene with Table

11 How to Play

This is simple. You need only install the **Jungle Book.apk** on your cellphone and run the application. Before doing this, just be sure that you put images of animals on your laptop or other devices. When you have done all of the procedures, it is time to enjoy the application. Fig.10 depicts the final level of the project.

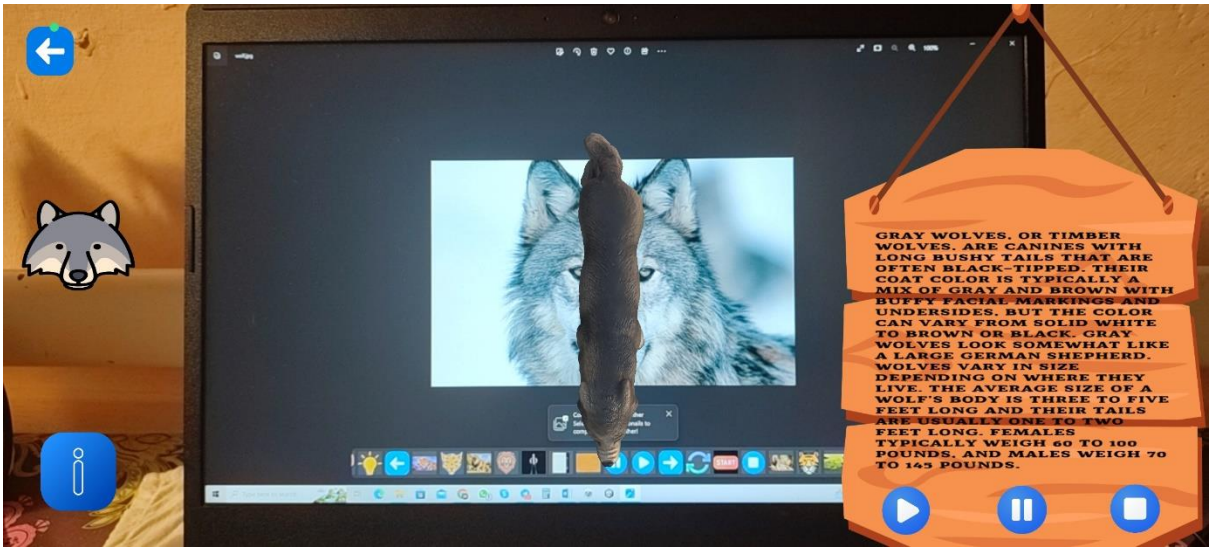


Fig.10 Final Level

References

https://dynamics.microsoft.com/en-us/mixed-reality/guides/what-is-augmented-reality-ar/

https://unity.com/

https://docs.unity3d.com/2017.2/Documentation/Manual/vuforia-sdk-overview.html

https://developer.vuforia.com/

https://www.flaticon.com/

https://sketchfab.com/search?q=lions&type=models