University of Science and Technology Chittagong (USTC)
Faculty of Science, Engineering & Technology
Department of Computer Science & Engineering

Lab Report

Course code : CSE 418

Course Title : Computer Graphics and Image Processing Lab

Project Name :

An Augmented Reality Image Tracking Application with Interactive Dragon Model.

Team Name : Dynamos.

Submitted by :

Team Members	ID
Abhilash Chowdhury	22010110
Tazrian Binte Mahfuz	22010116

Semester : 7th

Batch : 38th

Dept. : CSE

Submitted to:

Topu Biswas

[Assistant Professor]

Department of CSE

FSET, USTC

Objective

To develop an interactive Augmented Reality (AR) application using Unity and AR Foundation. The application detects a predefined image through the mobile device's camera and spawns a 3D dragon model over it. Users can then control the dragon's movement using an on-screen joystick interface. This project helps in understanding the core principles of AR such as image tracking, prefab instantiation, and real-time input interaction.

Tools & Technologies Used

- Unity 2021.3.6 (LTS version)
- AR Foundation (v4.2.6)
- ARCore XR Plugin
- Joystick Pack Asset
- 3D Dragon Model
- Android SDK, NDK, JDK
- Visual Studio (Code Editor)
- Android Device (for Testing)

Project Setup Steps

Unity Setup

- Installed Unity Hub and Unity Editor version 2021.3.6f1.
- Installed Android Build Support module with SDK, NDK, and JDK.

New Project Creation

- Created a new Unity project using the 3D Core template.
- Named the project: DragonController.

Implementation Steps

1. Importing Assets

Imported three main assets

- Reference image for image tracking (e.g., Game of Thrones poster).
- 3D Dragon model (as prefab).
- Joystick Pack (for movement control).

1. Build Settings Configuration

- Switched platform to Android.
- Modified Player Settings:
  - Graphics API: Vulkan only
  - Disabled Multi-threaded Rendering
  - Scripting Backend: IL2CPP
  - ARM64 architecture selected
  - Target API Level set to 24 or higher

2. AR Setup in Scene

- Removed the default camera.
- Added AR Session and AR Session Origin.
- Attached AR Tracked Image Manager to the AR Session Origin.
- Created a Reference Image Library and added the detection image.

3. Scripting the AR Image Detection

- Created a script PrefabCreator.cs that:
  - Listens to image detection events.
  - Spawns a dragon prefab above the tracked image.
  - Applies an optional position offset.

```
[SerializeField] private GameObject dragonPrefab;
[SerializeField] private Vector3 prefabOffset;

private GameObject dragon;
private ARTrackedImageManager aRTrackedImageManager;
```

4. Setting Dragon Prefab & Animation

- Scaled the dragon model for mobile AR display.
- Applied rotation so the dragon faces the camera.
- Configured animations (e.g., take-off and fly-forward).

5. Adding Joystick Controls

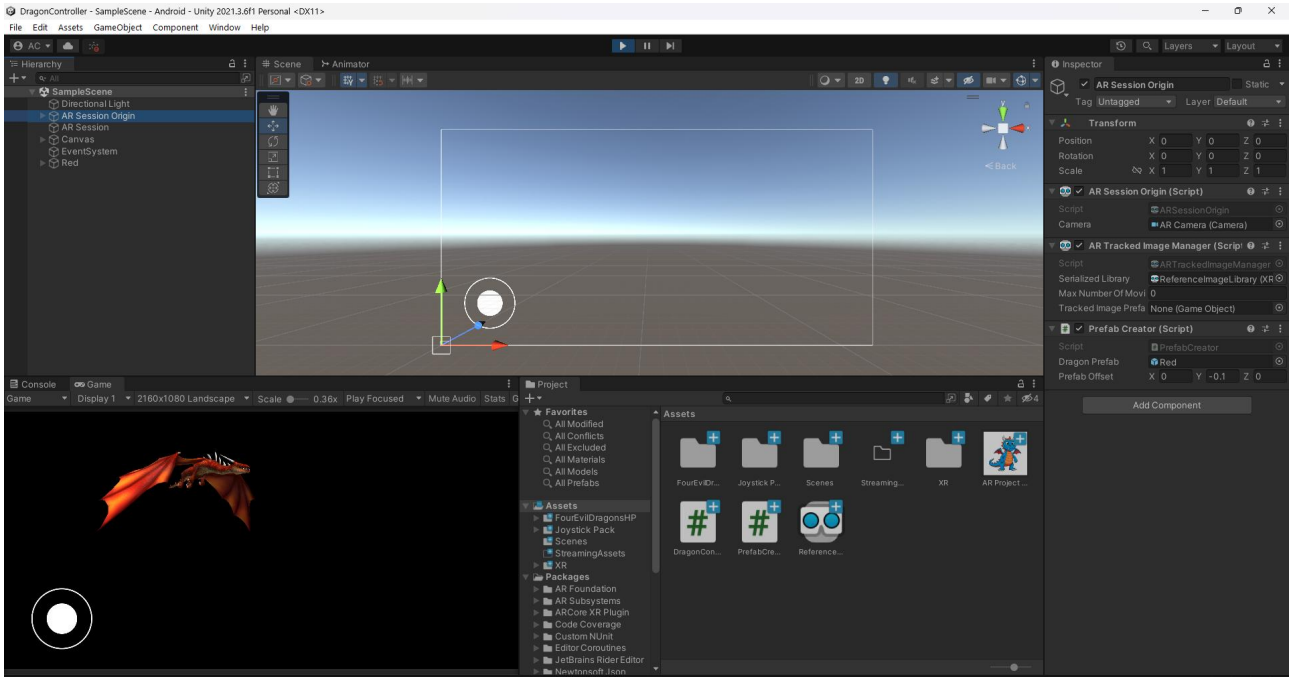
- UI Setup:
  - Added a Canvas with Scale With Screen Size mode.
  - Added a FixedJoystick from the joystick asset pack.

- Positioned it for easy thumb access on screen.

■ Dragon Movement Script:

- Created a script DragonController.cs to:
  - ➢ Get joystick values.
  - ➢ Move the dragon in 3D space.
  - ➢ Rotate it in the direction of motion using Euler angles and Mathf.Atan2.

Development and Deployment using Unity



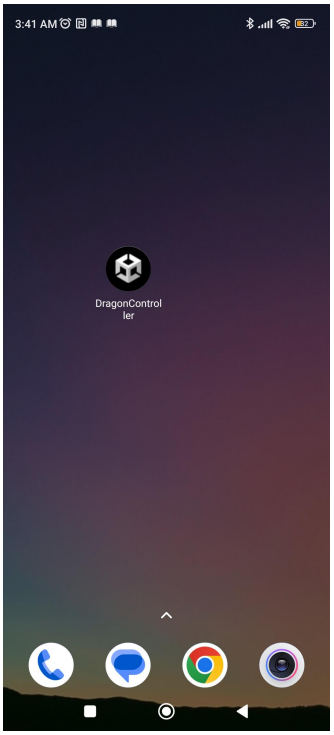


Figure: Application on Android device after installation

Figure: Splash screen while launching the Unity app

Result

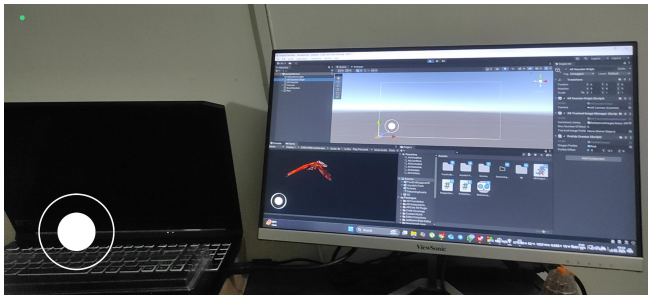


Figure: Opening the app accesses the camera

Figure: Dragon model appears on top of the target image

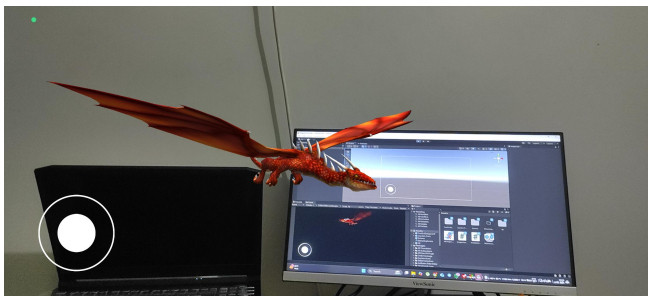


Figure: Moving the Dragon

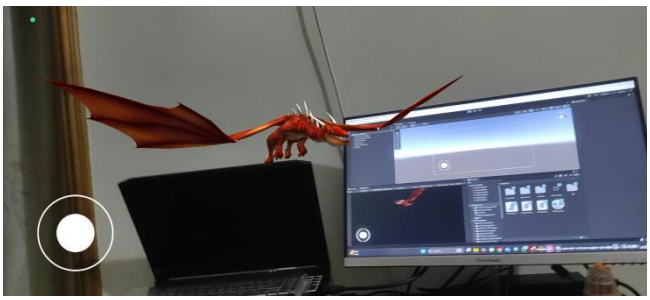


Figure: Switching the Camera angle

The AR application successfully detects the chosen image, spawns a dragon model on it, and allows the user to move it using an on-screen joystick. The animations run smoothly, and the dragon's movement direction adapts based on joystick input.

Future Improvements

- Add multiple image tracking with different prefabs.
- Enable object scaling/rotation via touch gestures.
- Implement scene transitions or game logic based on AR triggers.

Conclusion

This project demonstrates a real-world application of Unity's AR Foundation by combining image tracking with interactive 3D elements. It helped in gaining practical experience in configuring AR settings, integrating prefabs, customizing UI for control input, and deploying the app on Android. This hands-on project reinforced the understanding of AR pipelines and cross-platform deployment.