GOOD

Unity

→ Unity is a cross-platform game development system.

→ It consists of a game engine and an IDE

Basic Concepts (Very important)

project : It consists of all the models, game objects, assets, scripts, scenes and so on.

Scene :- It consists of the game objects that constitute the world the player sees

Packages :- It's an aggregation of the game objects and their associated metadata.

prefabs :- It's a template of grouping various assets under a single header.

→ It's used for creating multiple resources of the same object

Ex: trees, leaves

→ prefabs can be instantiated during runtime.

Game objects :- These are the thing that constitute the scene

→ light sources

→ Audio sources

→ cameras

→ game play logic

→ UI, etc.

Scene graph :- Its a graph/tree based hierachial structuring of the nodes

→ all tree nodes will have a single parent but can have multiple children.

→ operations applied to parents are applied to children also but vice-versa doesn't happen.

Components :- These are functional pieces of every game object.

→ These contain properties which can be edited and used to define the behaviour of the game object.

Example :- mesh filter, mesh renderer, Rigidbody, colliders, videoplayer, etc.

Scripts :- many inbuilt components already exist but if you want to build a new component we can use scripts.

→ These scripts inherit from MonoBehaviour

Assets :- It's a resource that will be used as a part of an object's component

(or)

It's an item that can be used in a unity project

ex :- scenes, prefabs, scripts, textures, animations.

Shaders :- Unity has several builtin shaders

(i) standard shader

A shader is used to render real world objects such as wood, glass, plastic and metal.

It supports wide range of shader types and combinations.

Lighting :- It describes abt the light property

→ point lights → ex lamps, candles

→ spot lights → ex, flash, torch, car headlights

→ directional lights → ex, sun, tubelights

→ Area lights

Scripting in Unity :-

→ Its done using C#

→ Scripts are an example of a component associated with a game object.

MonoBehaviour (Fundamental class)

→ Every script created in unity extends this monoBehaviour class.

→ It contains some prebuilt methods which are available to game objects

such as Awake(), Start(), Update(), Destroy(), Instantiate(), etc.

GameObject (Fundamental class)

→ Its a generic type from which all the game objects are created.

→ GameObjects have an associated name and tag.

ex: how to find main camera reference by its name

```
GameObject camera = GameObject.Find("Main Camera");
```

Transform (Fundamental class)

→ every game object in a scene has this Transform.

→ Its used to store and manipulate the position, rotation and scale of the object.

position → transform.position

rotation → transform.eulerAngles

Virtual Reality

Inducing target behaviour in an organism by using artificial sensory stimulation, while the organism has little or no awarness of the interference.

targeted behaviour :- the natural experience designed by the creator

organism :- any life form

artificial sensory stimulation :- The senses of an organism are integrated and their ordinary inputs are replaced or enhanced by artificial stimulation.

awarness :- being foole

Different Types of AR

→ Projection Based AR

- Laser imposed keyboards, etc.

→ Recognition Based AR

→ Marker Based :- It works by scanning a marker which triggers an augmented object to appear on the device.

→ Marker less :- It works by scanning an area by which the device overlays the augmented objects.

Ex:- Pokemon go, Ikea app.

→ Location Based AR

It relies on GPS to overlay the augmented experience

Ex:- google lens, etc.

Outline AR : In this type of AR, the device outlines some of the important boundaries

Superimposed AR : In this type of AR, it provides an alternate view of the object in concern.

ex: superimposing of hand bones structure when a hand is captured on a device.

What is AR?

It's a combination of real scene viewed by a user and virtual scene generated by the computer that is augmented to the real scene.

(or)

It's the augmentation of digital objects or scenes to the physical world.

Milgram's Reality-virtuality Continuumm

Mixed reality

Real world
consist of physical world

AR
physical world augmented with digital elements

AV
virtual world augmented with real world or physical objects

VR
virtual world containing completely digitally generated elements.

AR applications

- In archaeology
- Architecture
- Urban design and planing
- Education
- Navigation
- Medical
- E commerce
- Games.
- Industry & Manufacturing

AR Vs VR

AR

- System augments digital elements to the real world.
- User can maintain the sense in the real world.
- Needs a mechanism to combine real world and virtual objects.
- Hard to register real and virtual scenes

VR

- An immersive environment is generated by the system.
- Senses are maintained and controlled by system.
- Needs a mechanism to make user feel the virtual world.
- Hard to make VR worlds interesting.

Difficulties of AR

Combining real and virtual worlds :-

- → We need precise models
- → We need location and optical properties of the viewer
- → Calibration of devices.
- → To combine local co-ordinate systems and the objects in the scene in a global co-ordinate system.
- → Track the objects in the scene as the user moves.

Realistic merging

- → It requires objects to behave in physically plausible manners when manipulated.

- → occlusion
- → collision detection
- → shadows

Research activities

- → methods to register both the real world along with virtual scenes and keep them registered in real time.

→ developing some new display technologies. to merge real world and virtual objects.

Performance Issues

→ update rate for generating the augmenting image.

→ Accuracy of the registration of the real and virtual image.

Failures in Registration

→ Noise
- position and pose of camera w.r.t scene

→ Image distortions

→ Time delays
- In calculating the camera position.

Note : Registration is a process which blends virtual objects generated by computer with

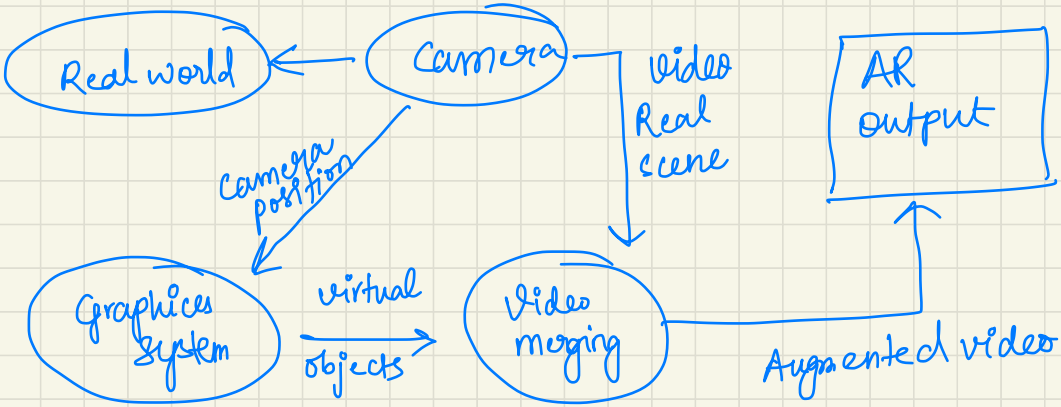
real world image caught by camera.

Display Technologies

Monitor Based AR

→ available easily.

→ We can use our cellphones as a window to see the AR scene



Examples:- → VAR system in football games

→ AD campaigns, etc.

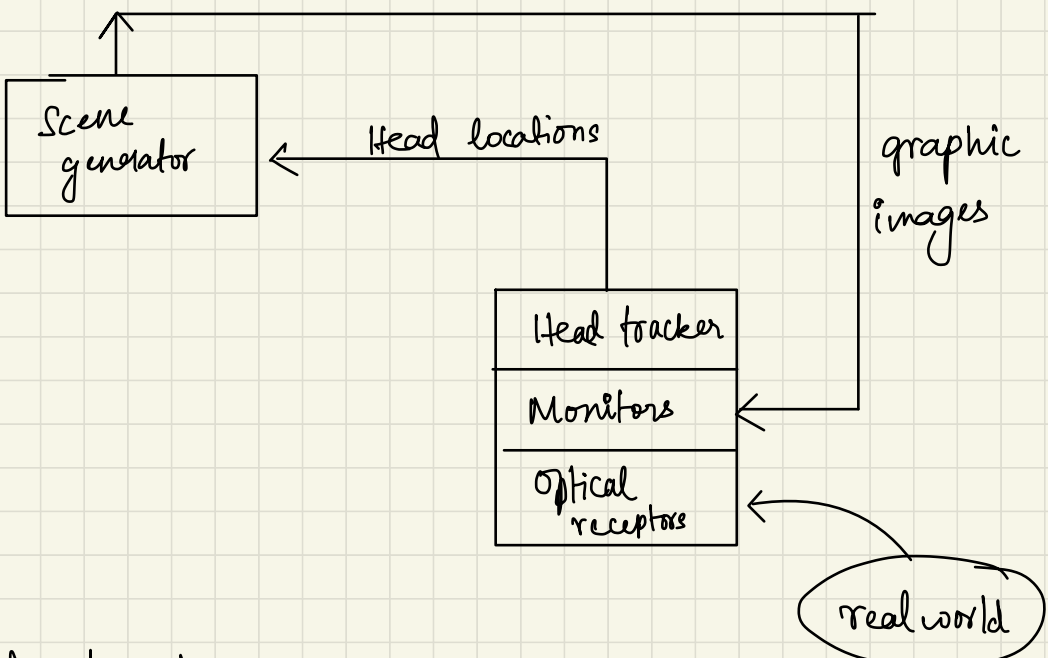
Advantages of Monitor Displays

→ consumer friendly equipment

→ less cost & most practical

HMD Based AR

Optical see through HMD (Head Mounted devices)

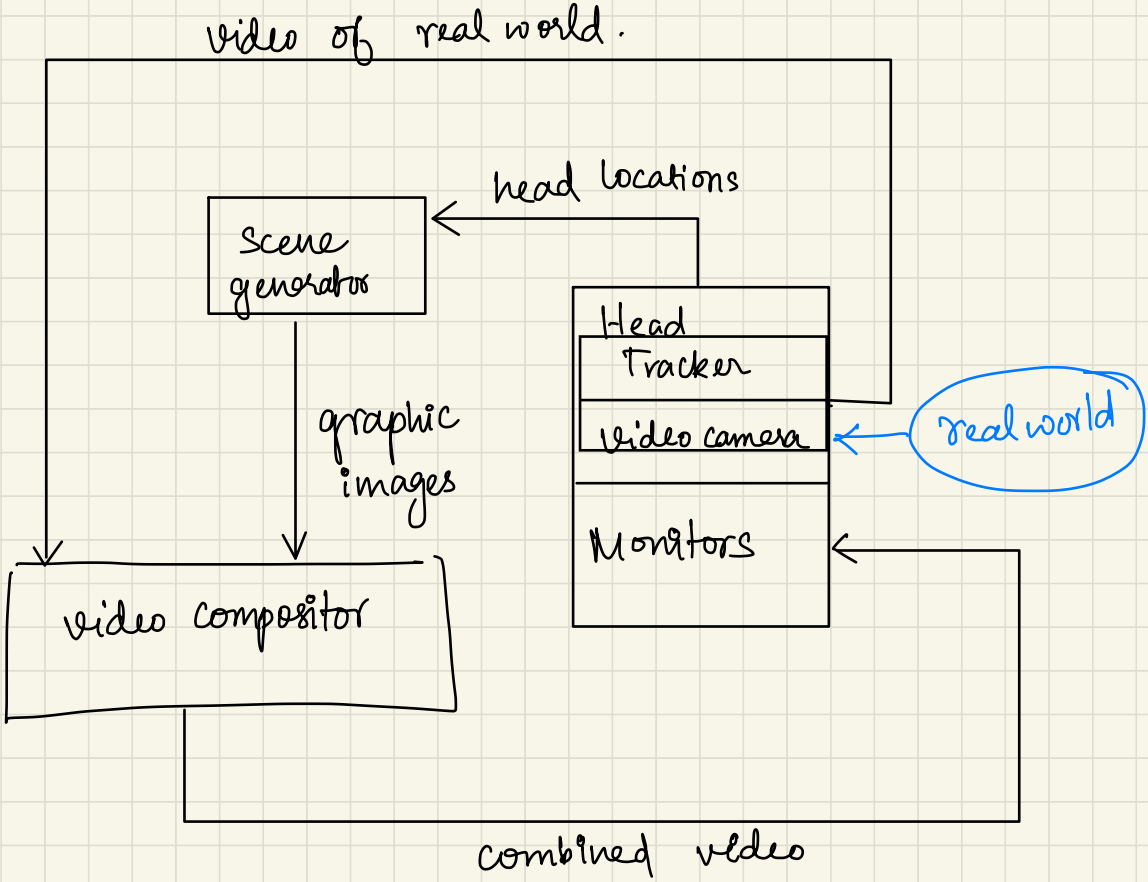


Advantages:-

(i) simplicity

(ii) Resolution, no eye offset.

video see through HMD



Advantages:-

→ flexibility in video composition

→ Real and virtual delays can be matched.

Evaluation Techniques

→ These are required to test the usability and functionalities of an AR or VR system.

→ The evaluation can be carried out in a lab or in field with collaboration of users.

→ It's used to evaluate both design and implementation.

→ It's a part of the design cycle, hence at every stage of the development of the system we have to evaluate the system.

Goals :-

→ assess the functionalities

→ assess the effect the interface will have on user.

→ identify if any problems are there such as design flaws, functionality glitches, etc.

Evaluation Designs

Cognitive Walkthrough

→ This method evaluates design on how well it supports user in learning task.

→ performed by cognitive psycologist (expert)

→ expert walks through the design to identify any flaws using psychological principles.

→ for each walk through, we have to consider:-

(i) what impact will interaction with the system have on user?

(ii) What cognitive processes are required?

(iii) What learning problems may occur?

Heuristic Evaluation

→ proposed by Nielsen and Molich.

→ experts use rules of thumb to measure the usability of UI's in walk throughs and report issues.

Review-based Evaluation

→ It's an expert based evaluation method which relies on experimental results and empirical evidence from the literature in order to support or refute parts of the UI.

AR feedback system

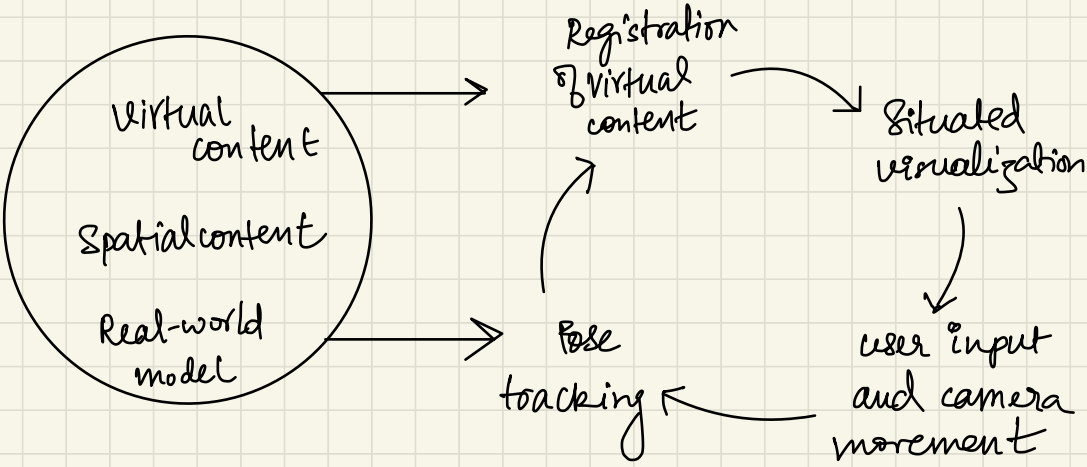
Components of AR :-

(i) **Visual component** :- Responsible for displaying the combined or merged scene to the users.

(ii) **Registration Component** :- It's responsible for combining the real world scenes with the virtual world object.

(iii) **Tracking component** :- It's responsible for tracking the user's head or camera positions in order to update the augmented scene as the user moves.

(iv) **Spatial component** :- Its a storage which keep data of the real world and virtual world.



AR uses a feedback loop b/ln user and computer system.

→ Real-world model serves as a reference to the tracking component, which determines the user's location in the real world.