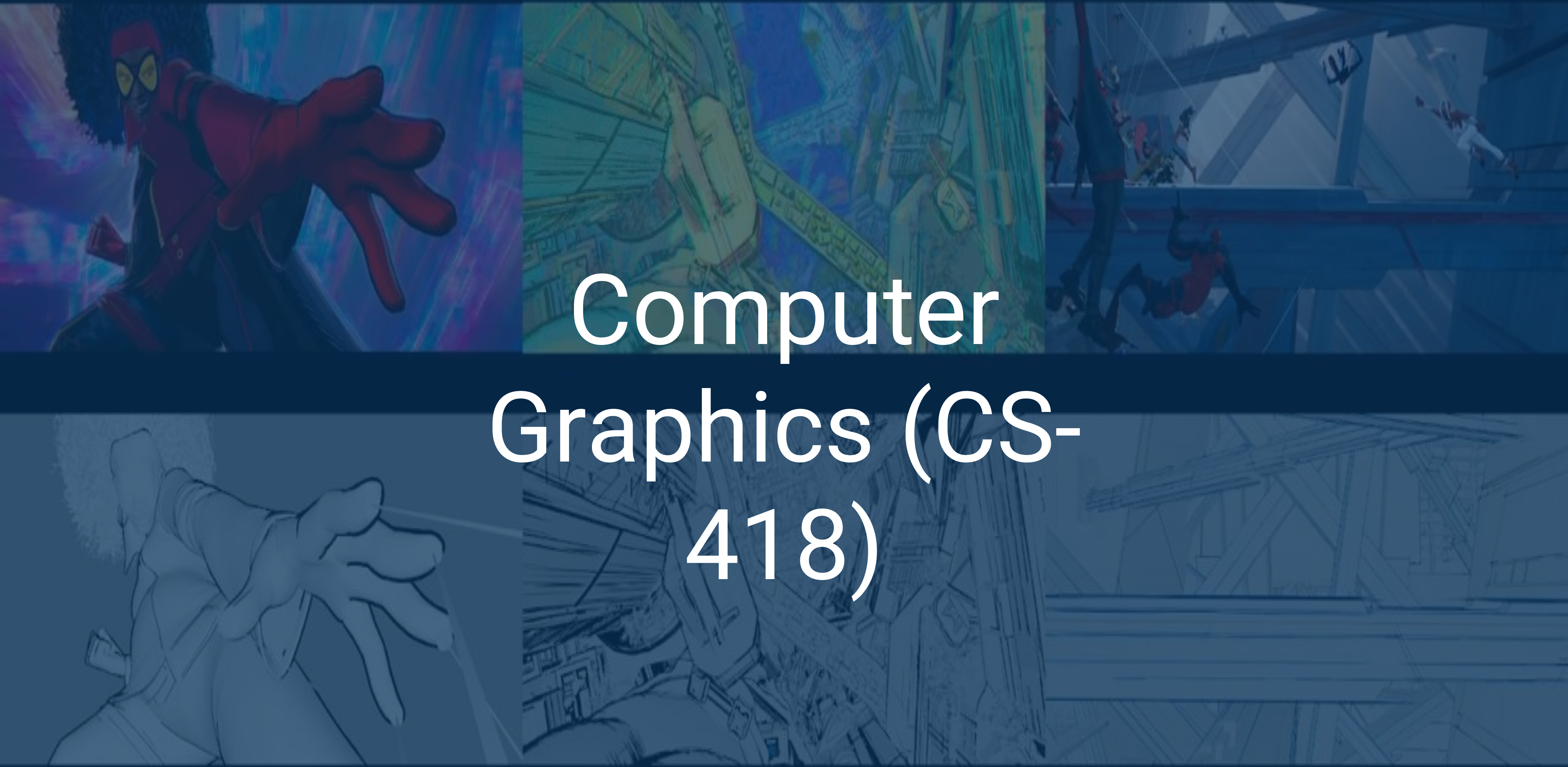


# Computer Graphics (CS-418)

# Course Structure

| | |
|---|---|
| • Lectures | 16 Weeks |
| • Assignment(2) | 06% |
| • Mid Term | 30% |
| • Final | 50% |
| • Project | 08% |
| • Quizzes(2) | 06% |

# Course Website

- All class material will be available  on the Google Classroom
- Lecture notes, handouts, papers to read, homework, project announcements, etc.

- Important: Check the GCR for every course announcements

# Course Material

- Books to read:
  - Fundamentals of Computer Graphics, 5th Edition,Steve Marschner and Peter Shirley, A K Peters/CRC Press, 2015., ISBN-10: 0136053580.
  - Fundamentals of Computer Graphics, 3rd Edition, A K Peters, 2009.
  - Computer Graphics: Principles and Practice, 3rd Edition, Addison Wesley, 2013.
  - Physically based rendering: from theory to implementation, 2nd Edition Matt Pharr and Greg Humphreys, 2010.

# The Project

- Done in teams of 2 students.
  - You can either make any game using CG pipeline.

    OR
  - You can design your own geometrical model using the CG pipeline.

**Projects will be reviewed and then approved**

# What this course is about?

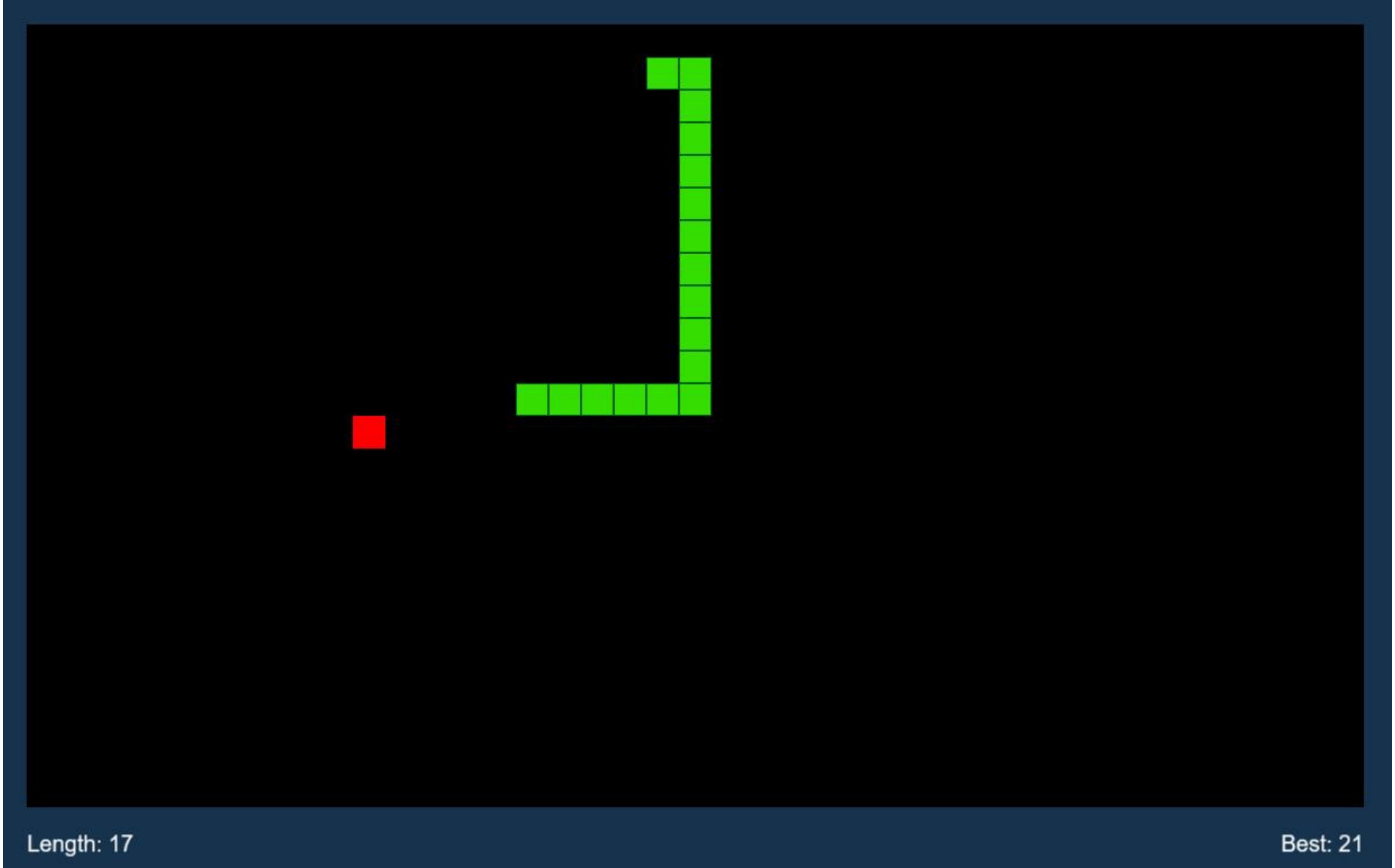
- A computer graphics course typically covers a range of topics that introduce you to how visual content is generated, manipulated, and displayed using computers.
- Learn basics of graphics systems, math (linear algebra, geometry), and coordinate systems.
- Study rendering techniques like rasterization, ray tracing, shading, and texturing.
- Explore 2D/3D modelling, transformations, curves, surfaces, and animation principles.
- Work with tools like OpenGL, Blender, or Unity and complete hands-on projects (e.g., games, 3D models).

# CGI 3D Making of: "The Incredible Hulk - Making Of" - by Robert Kuczera

THE
CG
BROS
HD

# Dinosaur Making (CG pipeline)

# Why Graphics?



Length: 17

Best: 21

# Things to Note…

- Everything bad but
  - was playable!
  - was in real time

*"The Legend of Zelda: Tears of the Kingdom," was developed using a custom engine by Nintendo.*

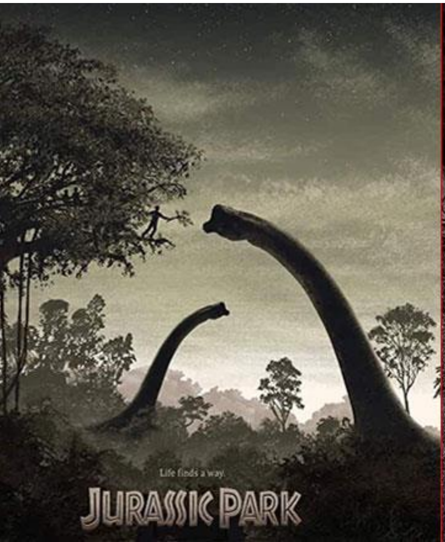
# What we want; May be

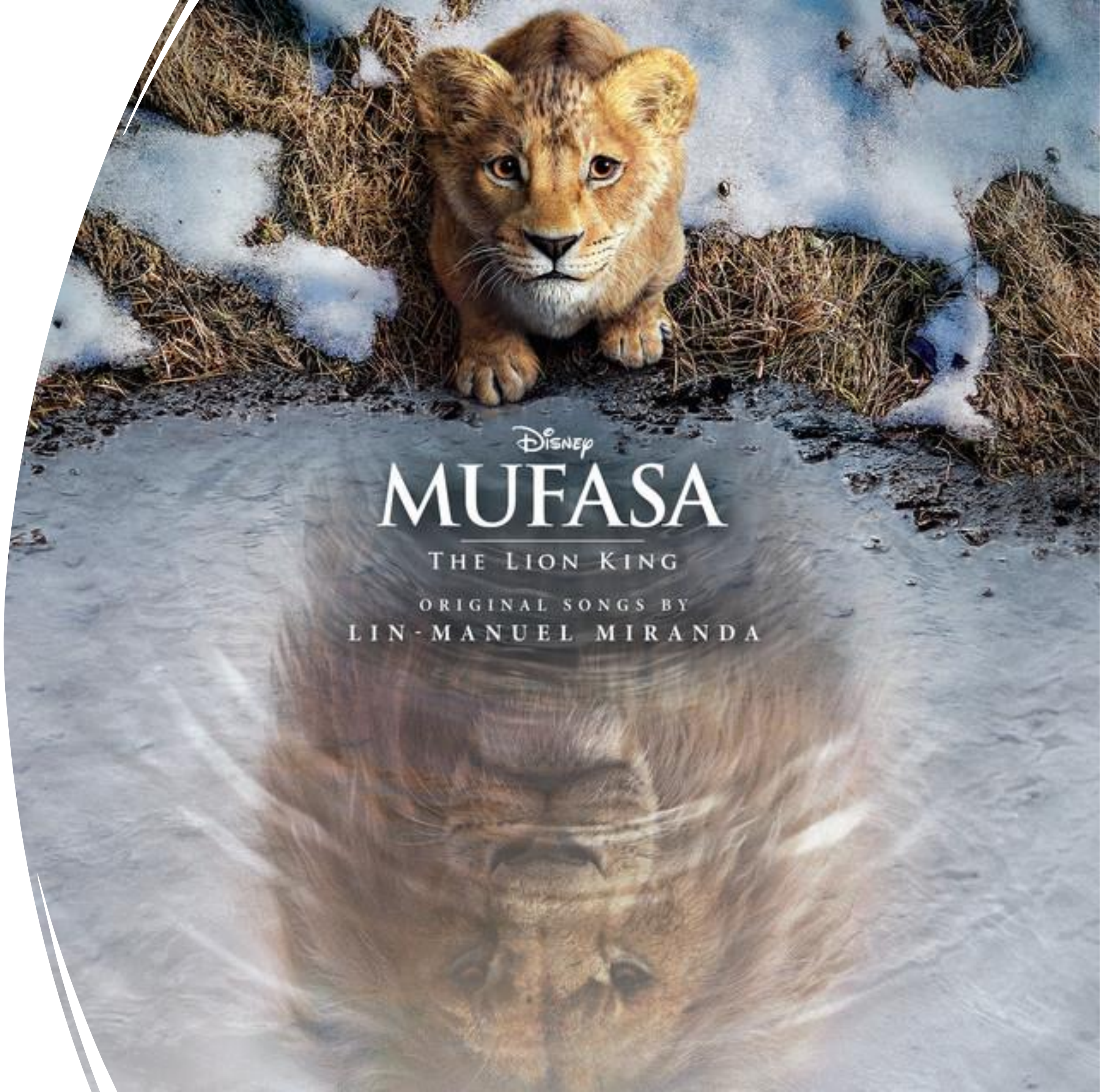
# Graphics on Mobile

"Zen Garden," released on iOS 8 in 2014, was developed using **Unreal Engine 4**. It was created as a tech demo to showcase the graphical capabilities of Unreal Engine 4 combined with Apple's Metal API, which allowed for more efficient GPU processing on iOS devices.

# Movies

- Mufasa: The Lion King," released in December 2024, is a photorealistic CGI-animated film produced by Walt Disney Pictures.





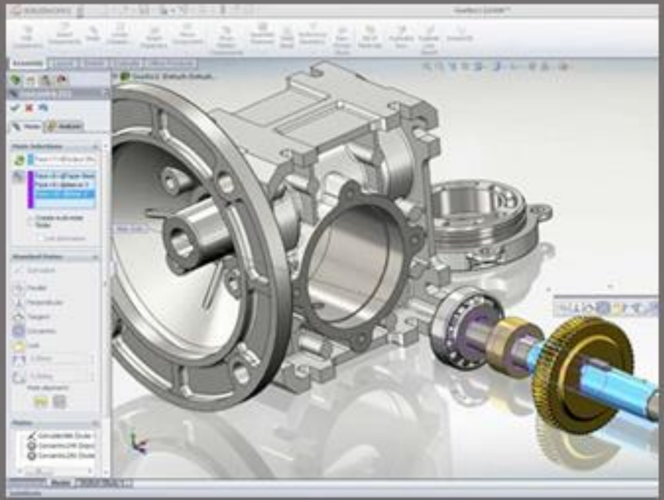
# Virtual Reality

Johnny English 3

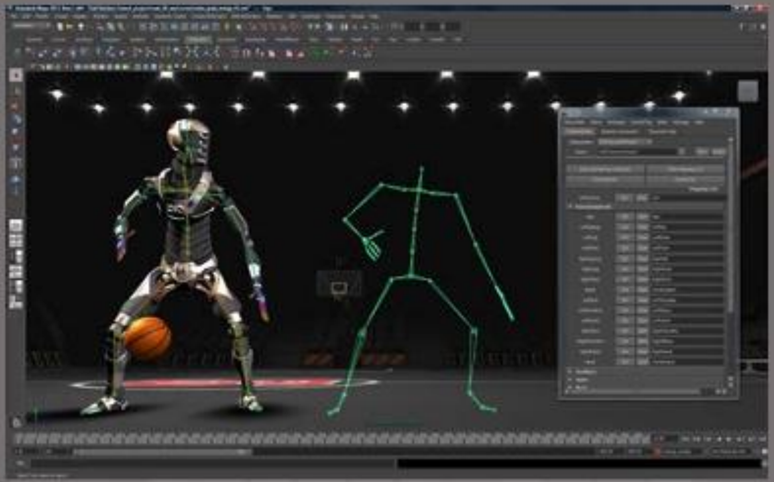
Trade show in Shenzhen,China

# Virtual Reality

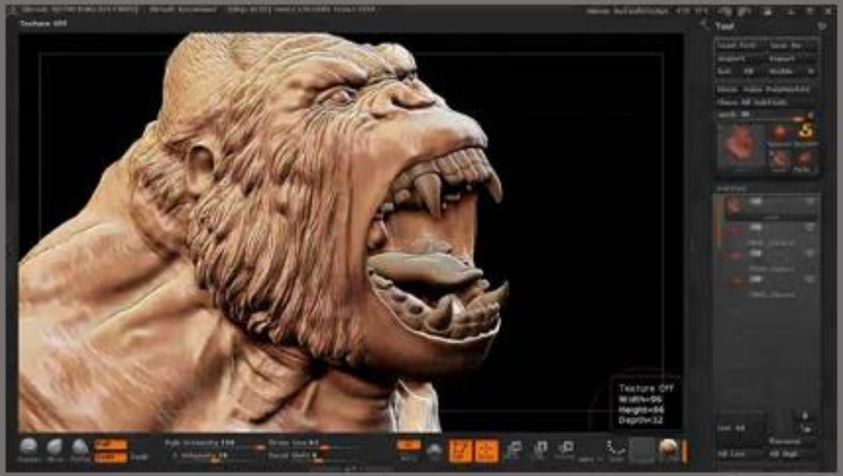
# CADs, Animators, Modelers



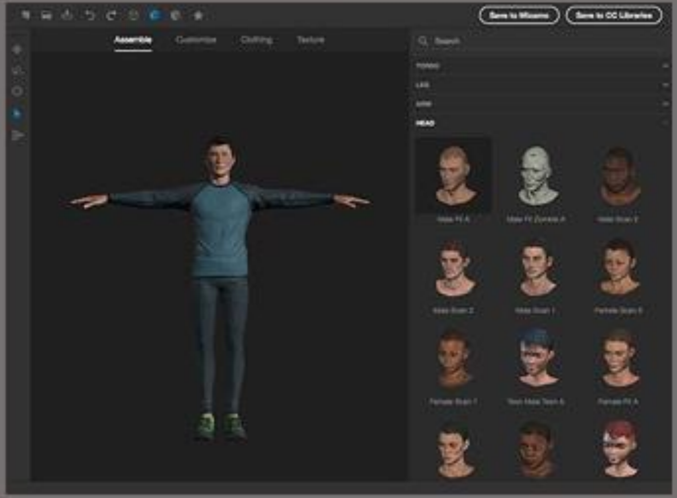
Solidworks, Dassault Systemes



Maya, Autodesk



ZBrush, Pixologic



Fuse, Adobe

# Hollywood Visual Effects

- One often cannot film various real-world situations required in order to tell a story
  - Some times the situation is too dangerous, impractical, expensive, or rare
  - Other times the situation doesn’t actually exist, except in an alternative reality

# Visual Effects: Liquid

Battleship

The Day After Tomorrow

© 2003 Warner Bros. Pictures. All Rights Reserved.

Terminator 2

# Visual Effects: Gasses

Harry Potter and the Order of the Phoenix

Terminator 3

Star Wars Episode III

# Visual Effects: Solid

Super 8

2012

TWO MINUTE
PAPERS
WITH KÁROLY ZSOLNAI-FEHÉR (KZF)
SIMULATING LIQUID-HAIR INTERACTIONS

Disclaimer: I was not part of this research project, I am merely providing commentary on this work.

https://www.cs.columbia.edu/cg/liquidhair/main_opt.pdf

# Visual Effects: CG Creatures

**Sméagol/Gollum, The Lord of the Rings**

**Anna  and Olaf, Ferozen**

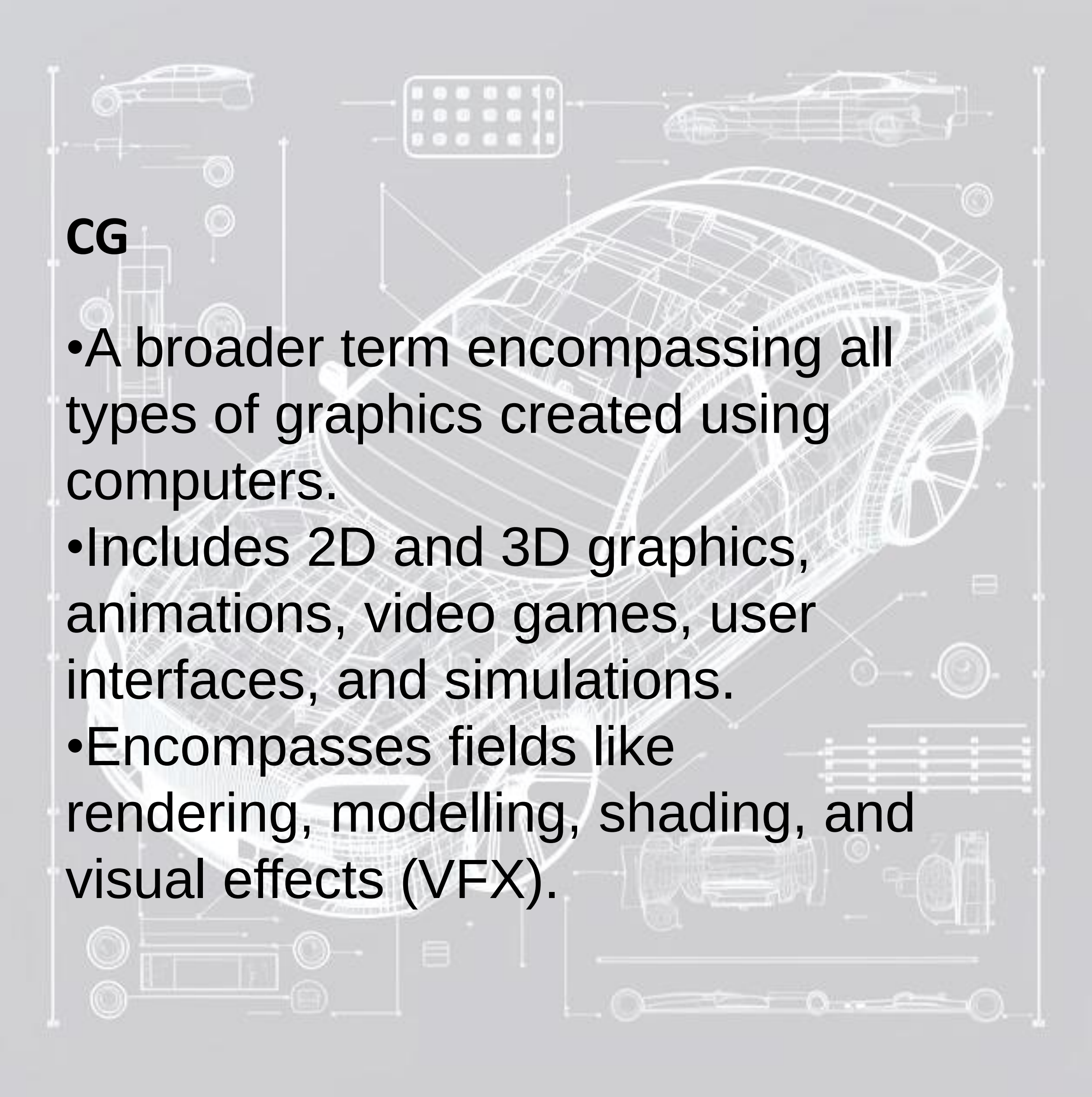
# Motion Capture

# Computer Graphics Is a Humongous Field

# Computer Graphics

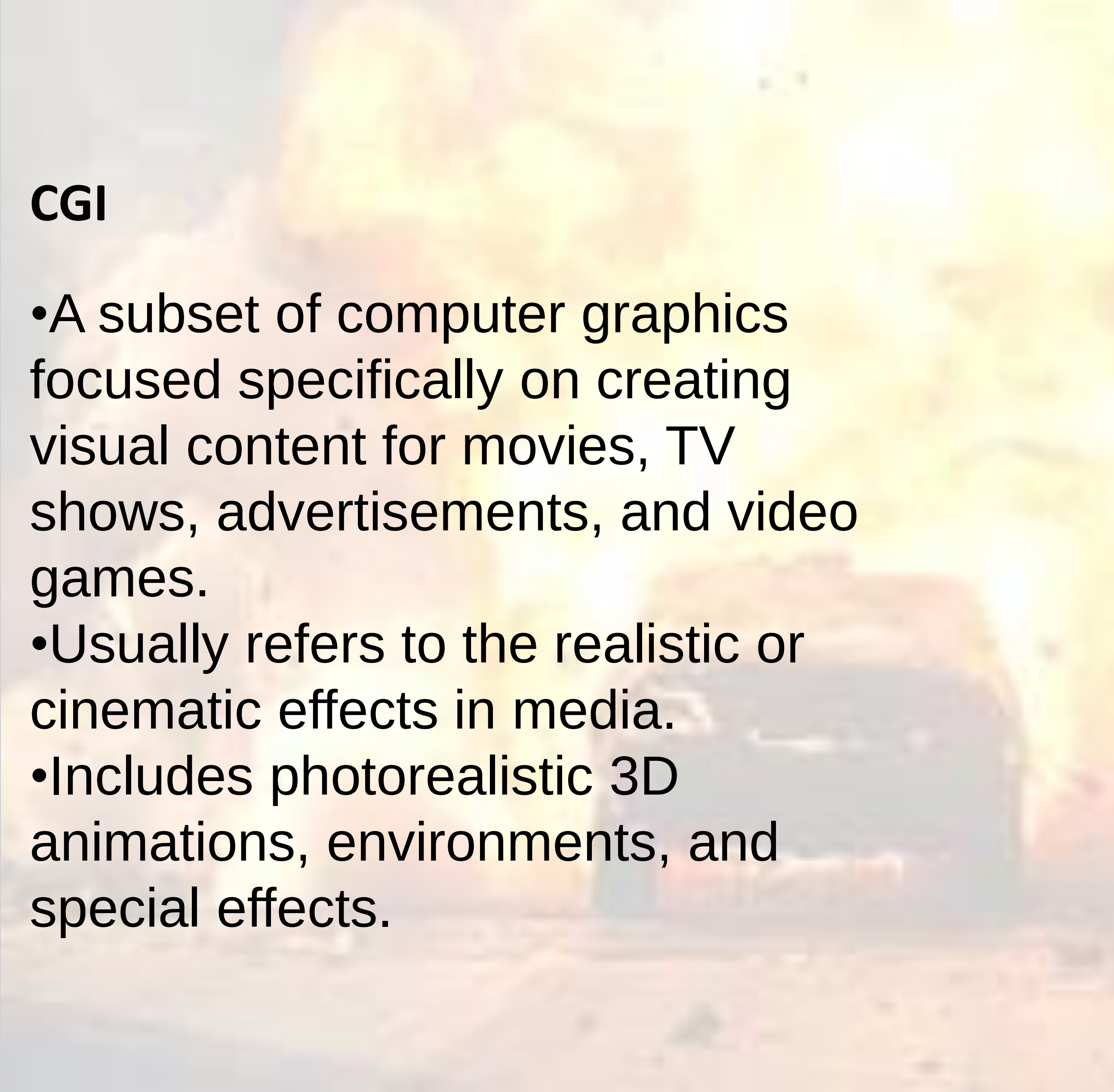
- CG spans multiple industries and disciplines, including:

1. **Entertainment:** 3D animation, CGI in movies, gaming, and virtual reality.
2. **Design and Art:** Digital illustration, architecture visualization, and product design.
3. **Simulations:** Scientific visualization, medical imaging, and engineering simulations.
4. **Human-Computer Interaction:** User interfaces and interactive media.
5. **Education and Training:** Virtual labs, e-learning, and simulations.
6. **Research:** Development of new algorithms, rendering techniques, and real-time graphics.



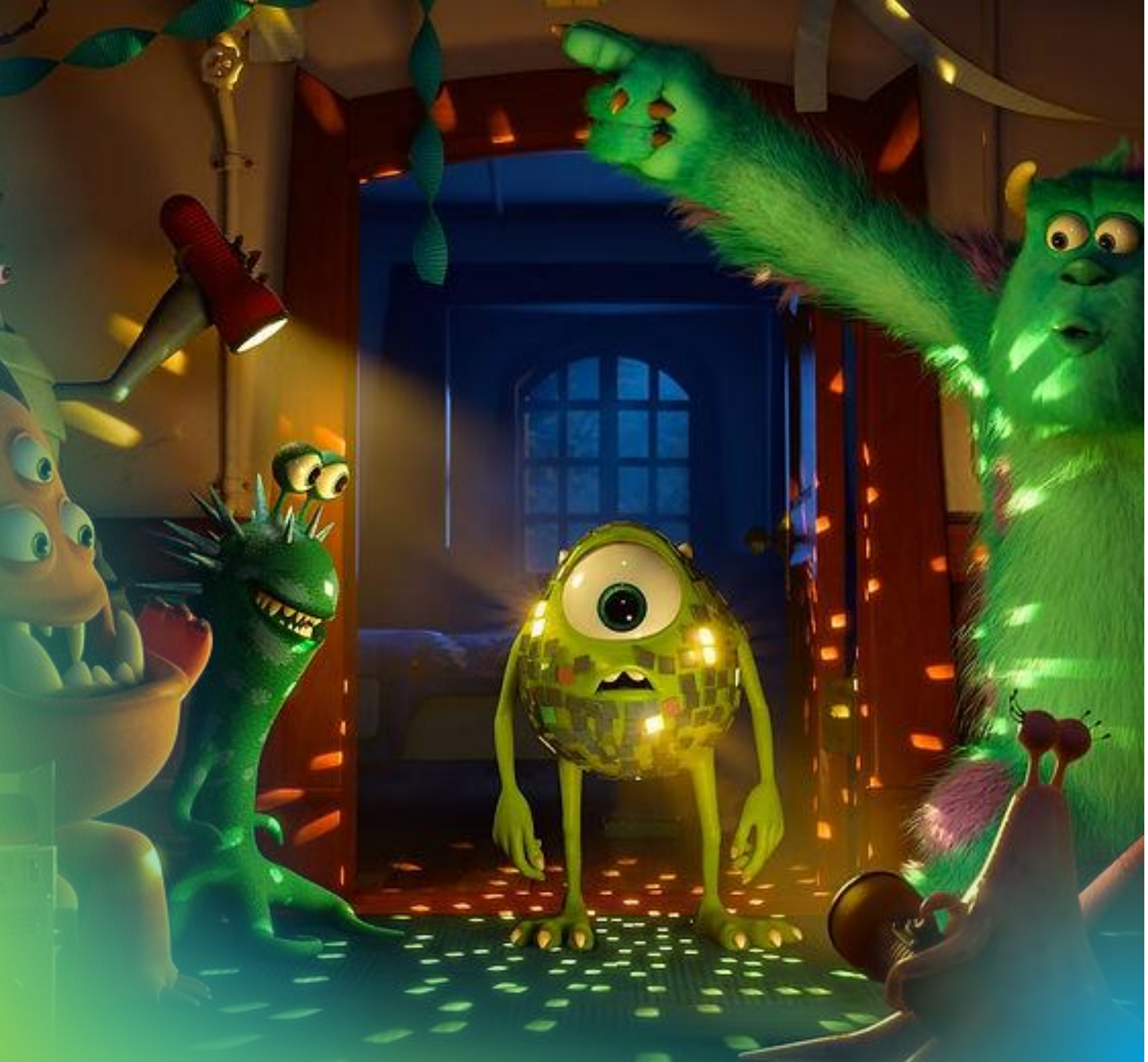
## CG

- A broader term encompassing all types of graphics created using computers.
- Includes 2D and 3D graphics, animations, video games, user interfaces, and simulations.
- Encompasses fields like rendering, modelling, shading, and visual effects (VFX).



## CGI

- A subset of computer graphics focused specifically on creating visual content for movies, TV shows, advertisements, and video games.
- Usually refers to the realistic or cinematic effects in media.
- Includes photorealistic 3D animations, environments, and special effects.



How do you make this picture?

# How do you make this picture?

- Modeling
  - Geometry
  - Material
  - Lighting
- Animation
  - Make it move
- Rendering
  - Draw the picture.

# Pipeline of Computer Graphics

**Modeling**: This is the process of creating a 3D representation of objects. It involves defining the shape, texture, and structure of the objects in a scene. This can be done using various modeling tools (e.g., Blender, Maya) and techniques like polygonal modeling..

**Transformation**: This stage involves applying transformations (such as translation, scaling, and rotation) to the objects in the scene to position them properly in world space. The object coordinates are transformed into world coordinates.

**Lighting**: Lighting determines how objects in the scene are illuminated. It involves setting up light sources, defining their properties (such as intensity, color, and position), and applying the appropriate lighting models (like Phong, Lambertian, etc.) to simulate real-world lighting effects.

**Camera Setup**: In this stage, the virtual camera’s position, orientation, and field of view are defined. The camera represents the viewpoint through which the scene will be rendered.

**Clipping**: Clipping is the process of removing any geometry that lies outside the camera's view frustum (the 3D pyramid-shaped region that defines what the camera can see). This helps improve performance and ensures that only visible parts of the scene are processed.

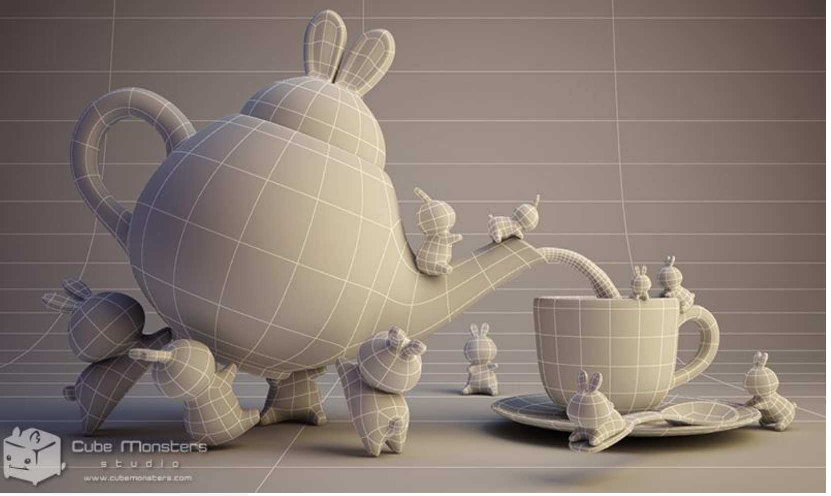
**Rasterization**: This step involves converting the 3D objects into 2D pixels on the screen. The geometric shapes (triangles, polygons) are projected onto a 2D plane using the camera's perspective, and their color and depth are determined based on lighting and material properties.

**Shading**: Shading adjusts how light affects each pixel of the texture..This step involves techniques like flat shading, Gouraud shading, and Phong shading, as well as more advanced methods like bump mapping, normal mapping, and reflection/refraction.

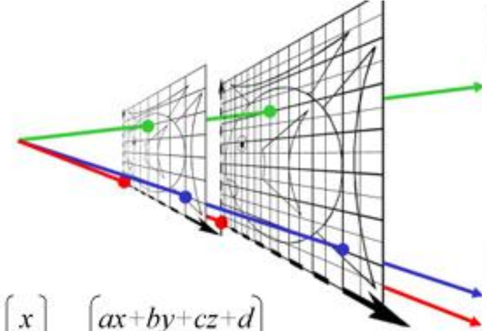
**Texturing**: Texturing involves applying 2D images (textures) to 3D models to give them a more detailed and realistic appearance. The textures are mapped onto the models using UV coordinates, which define how the texture wraps around the 3D geometry.

**Post-Processing**: This stage involves applying additional effects after the image has been rendered. It includes techniques such as anti-aliasing, depth of field, motion blur, color grading, and compositing, all of which enhance the visual quality of the final image.

**Output**: The final image is output to the screen, where it is displayed to the user. For 3D applications, this might involve continuously rendering images to create animations or interactive experiences.

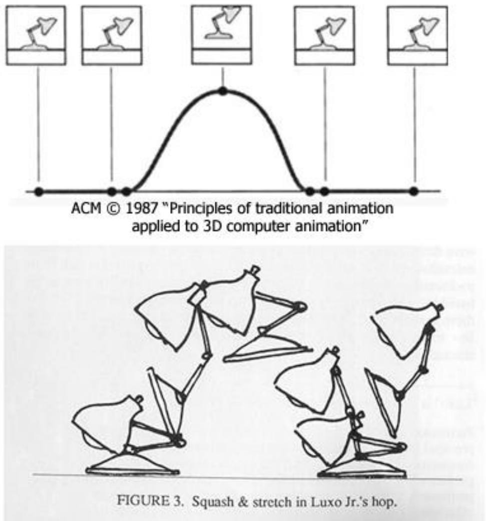


Modelling



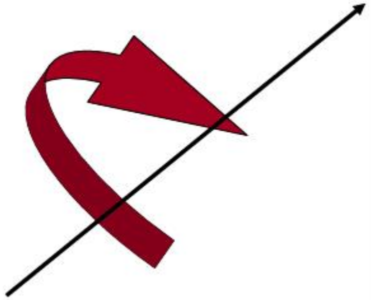
$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} ax+by+cz+d \\ ex+fy+gz+h \\ ix+jy+kz+l \\ 1 \end{bmatrix}$$

Transformation

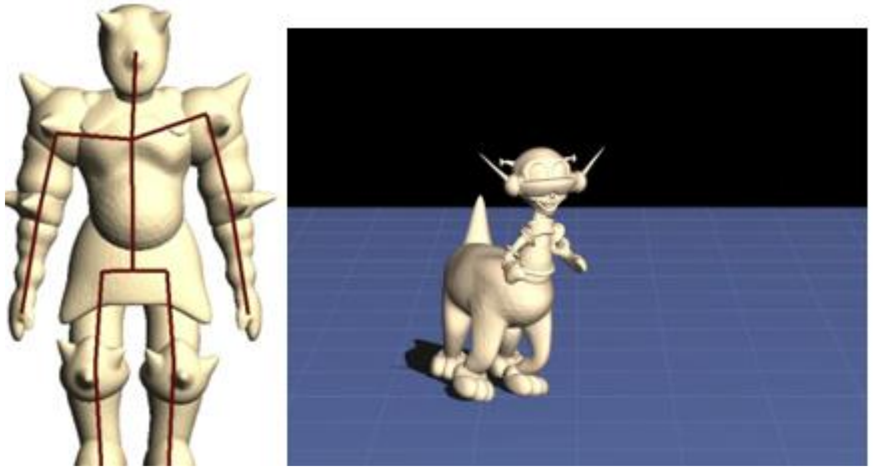


ACM © 1987 "Principles of traditional animation applied to 3D computer animation"

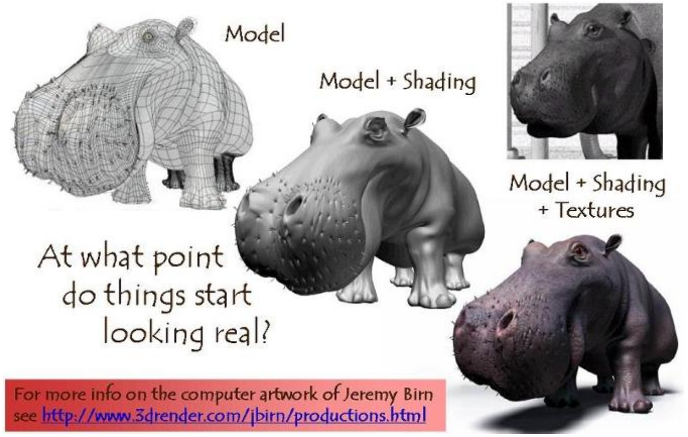
FIGURE 3. Squash & stretch in Luxo Jr.'s hop.



Animation



Rigging

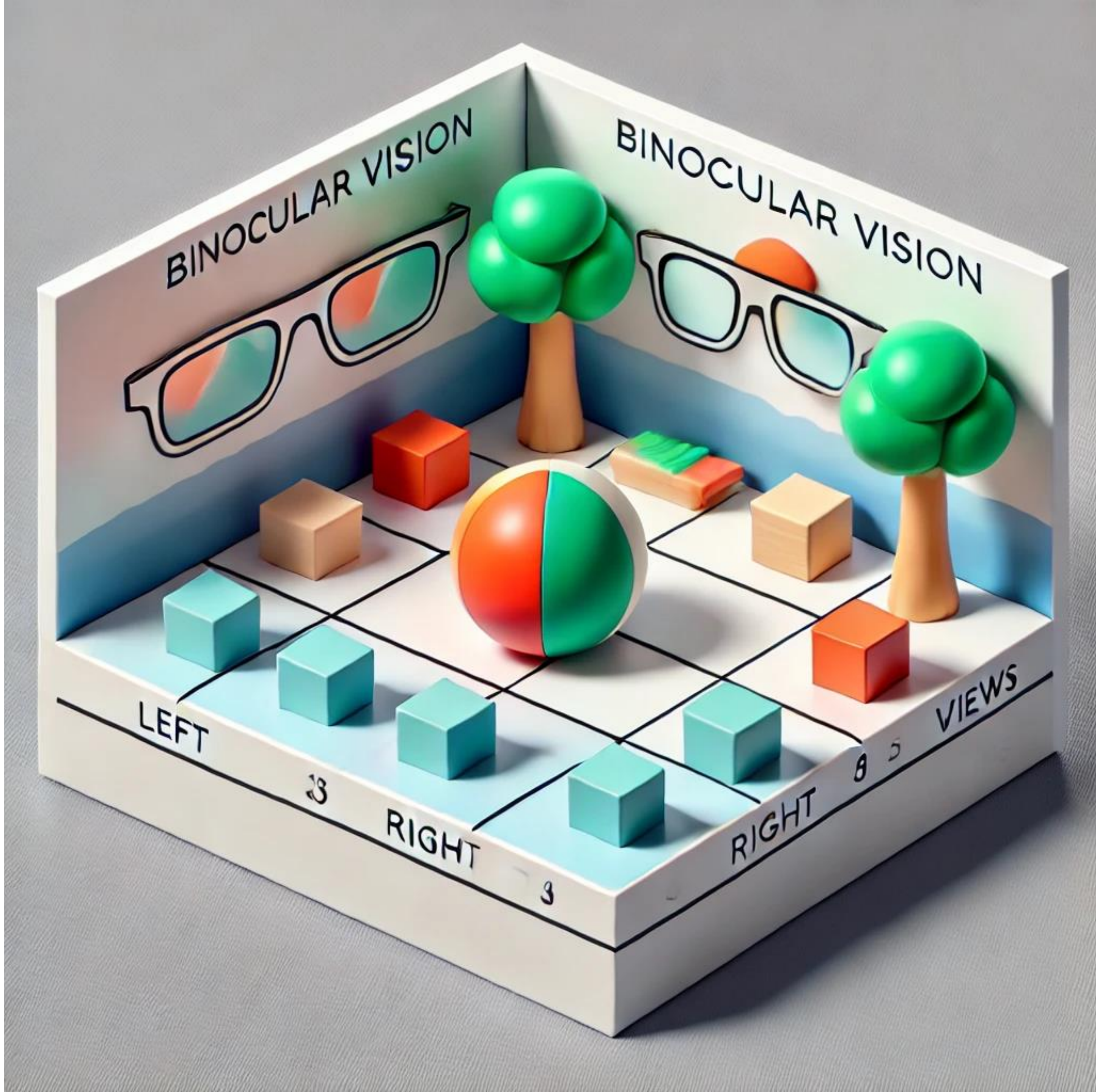

Model
Model + Shading
Model + Shading + Textures
At what point do things start looking real?
For more info on the computer artwork of Jeremy Birn
see http://www.3drender.com/jbirn/productions.html

Texture and Shading

Rendering/Ray tracing

# Into the 3rd Dimension

- Add *depth* dimension
- How do you perceive depth everyday?
  - You have two eyes
  - The image for your left eye is different than the right
  - Brain extracts the differences to understand depth


BINOCULAR VISION
BINOCULAR VISION
LEFT
RIGHT
RIGHT
VIEWS

# But Wait!

- Monitors are one “flat image on a flat surface”
- How do we perceive depth now?
    - Farther objects are smaller (foreshortening)
    - Delicate lighting changes
- We’ll use mathematics to do this for us

# Common Terminology

Rendering: the entire process of drawing an image to the screen

Vertex: a single 3D point (x, y, z)

Edge: a line between two vertices

Face: Most often 3 vertices and their edges

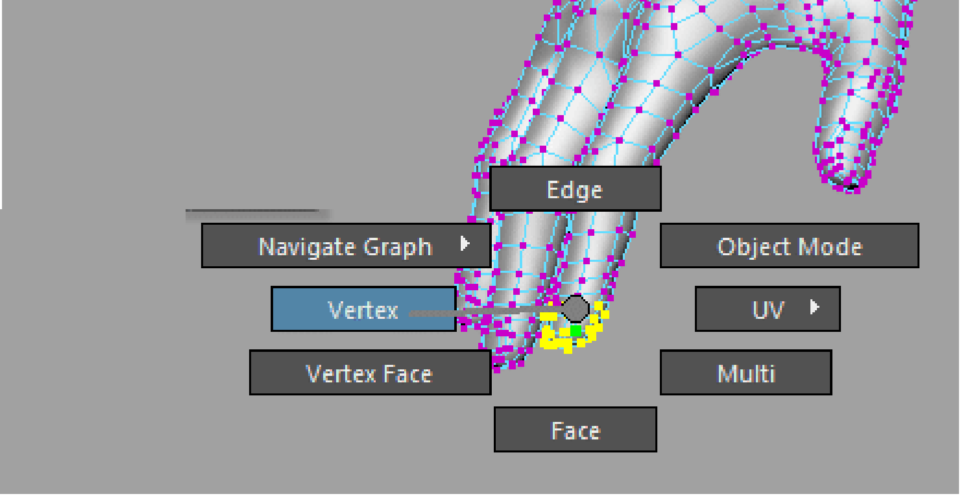
Transformations: moving one or more vertices

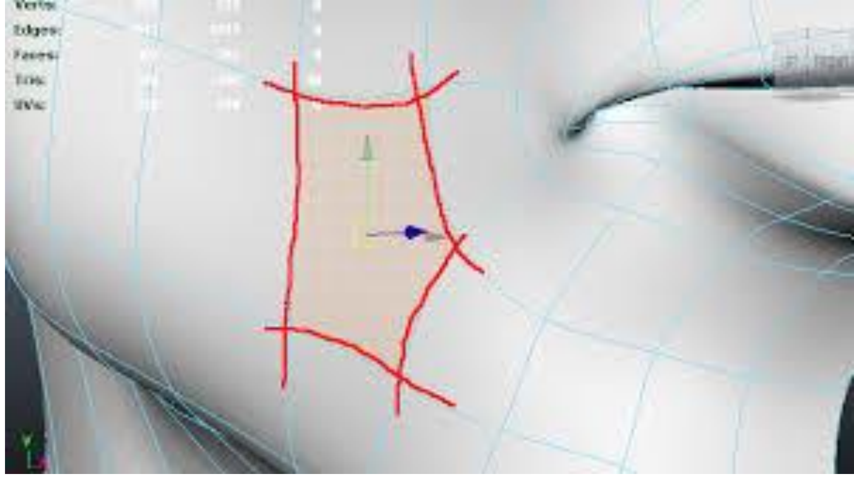
- Translate: pushing vertices along the x, y or z axis
- Rotate: revolving vertices around some 3D point
- Scale: increasing or decreasing the distance of vertices from their center

Model matrix – a mathematical structure for holding transformations (later)

Projection matrix – another, used to get images on the screen (later)

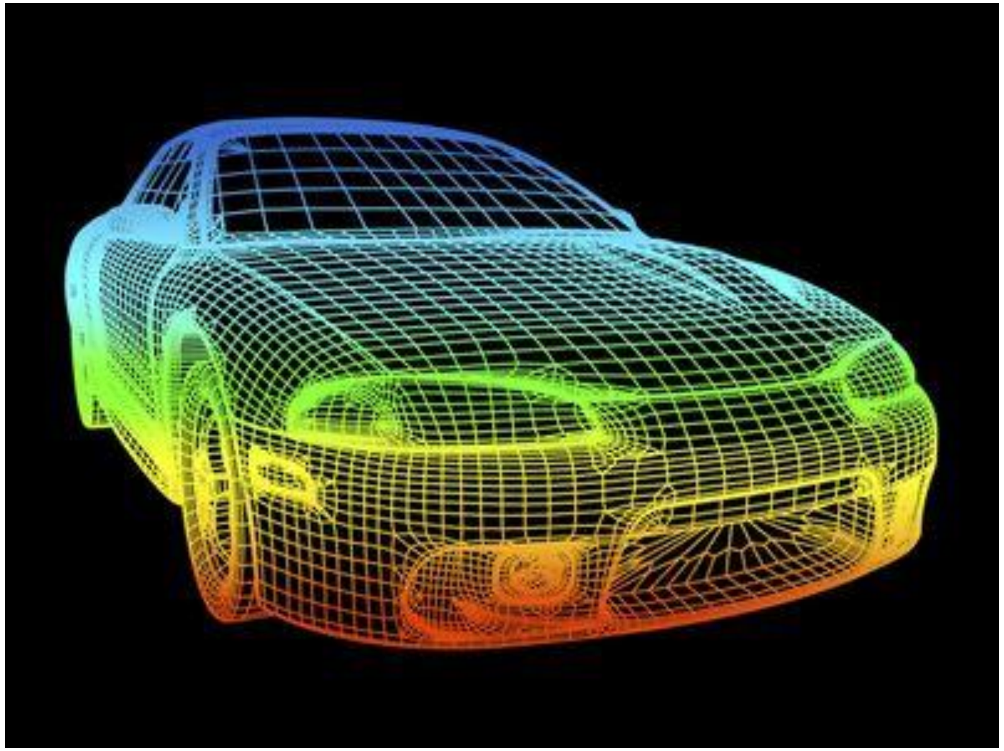
Rasterization – putting the actual pixels on the screen (final phase of rendering)





Vertex

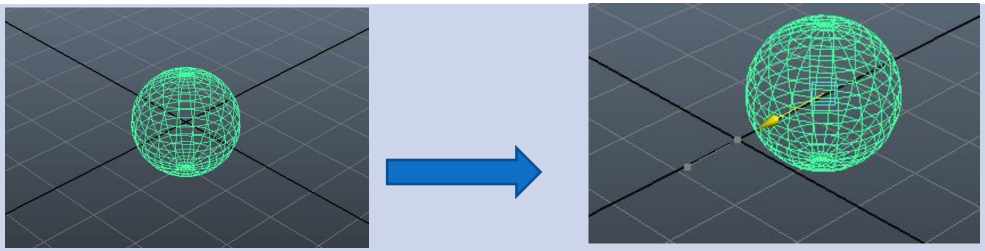
Faces



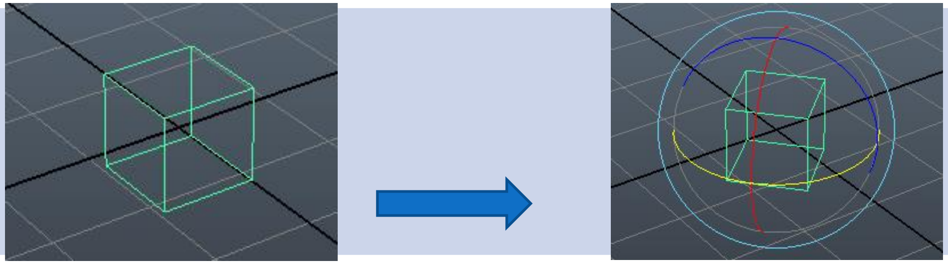
Edges

# Transformations

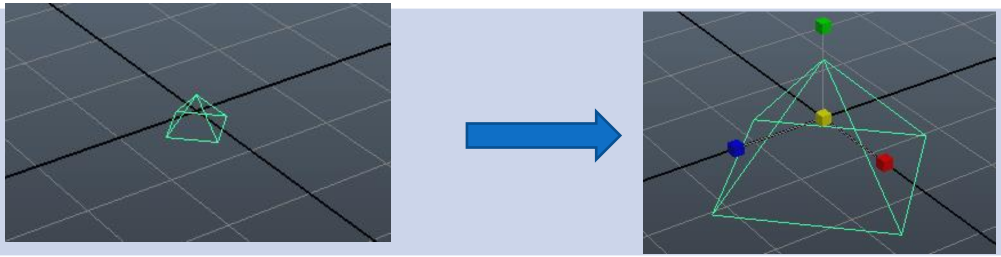
Translate

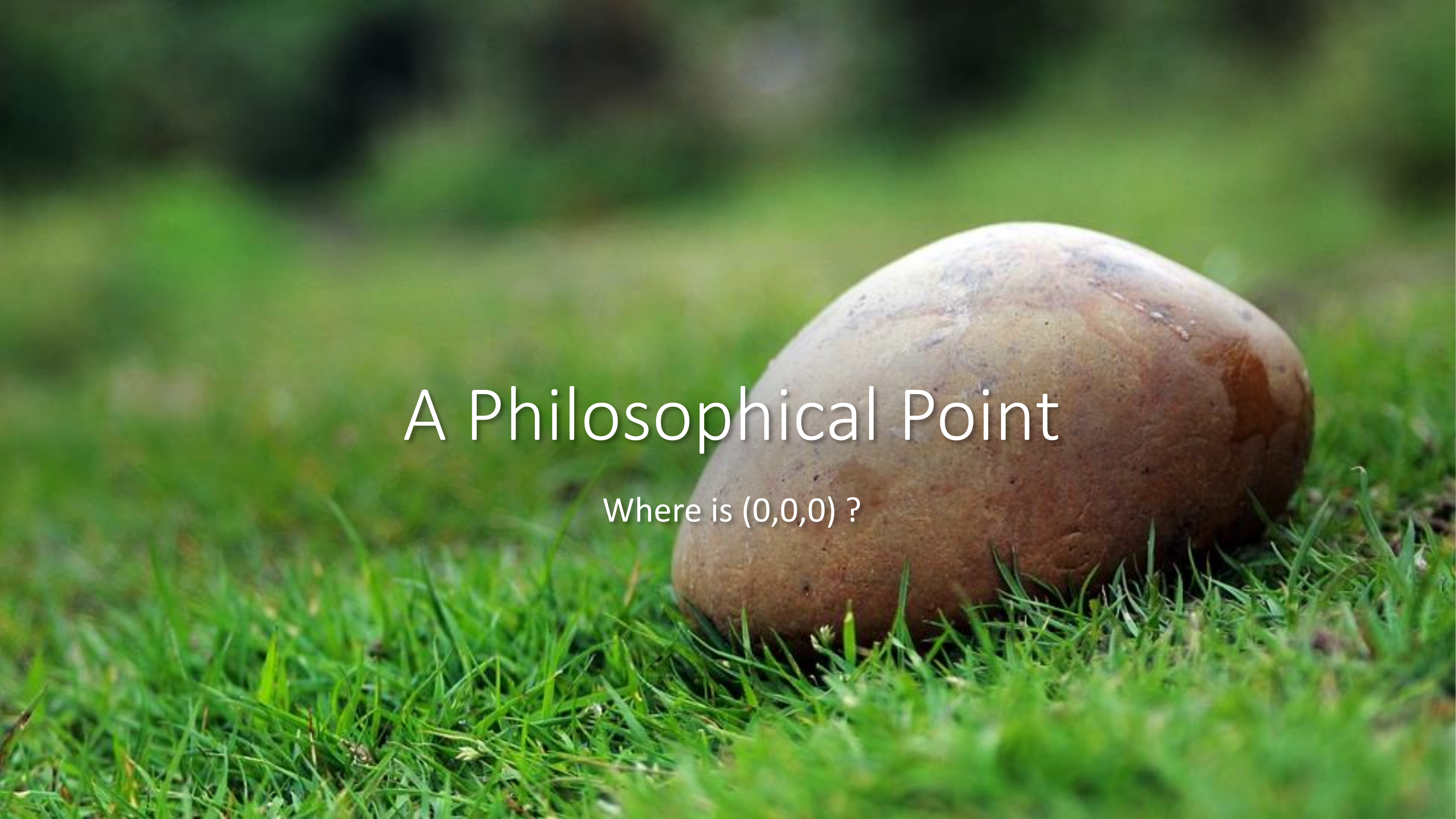


Rotate



Scale

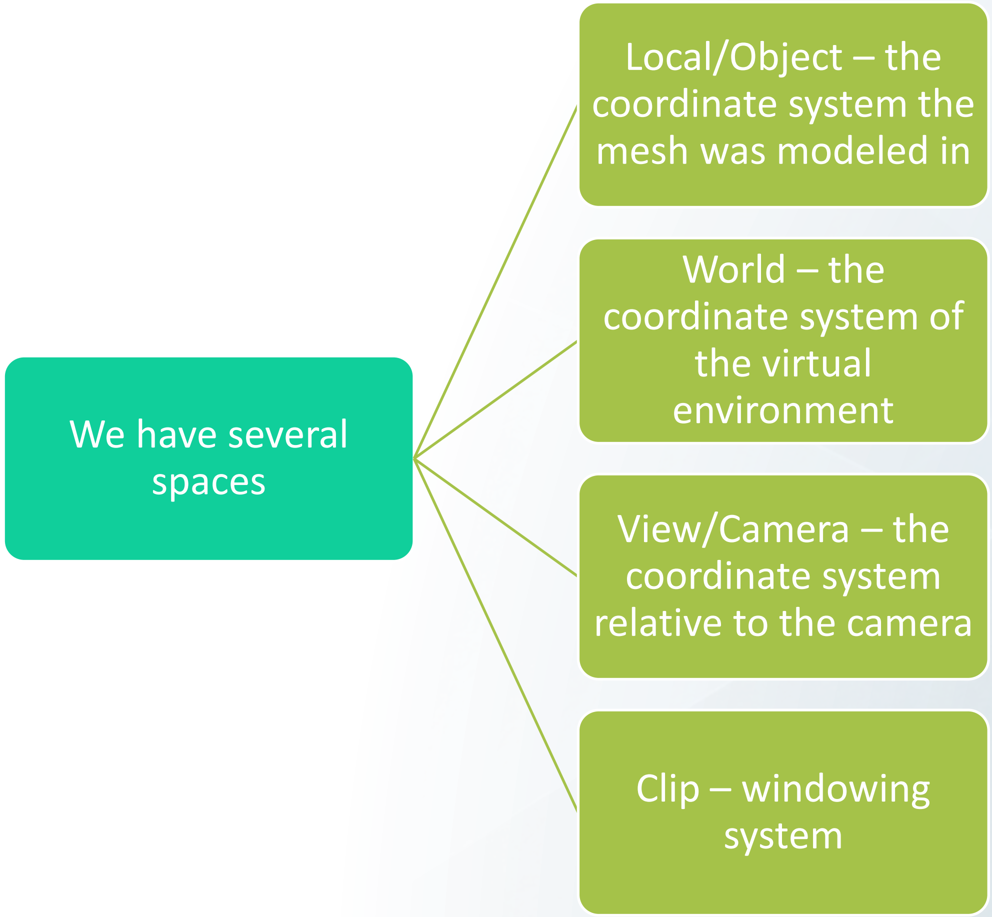




# A Philosophical Point

Where is (0,0,0) ?

# Coordinate Systems



We have several spaces

Local/Object – the coordinate system the mesh was modeled in

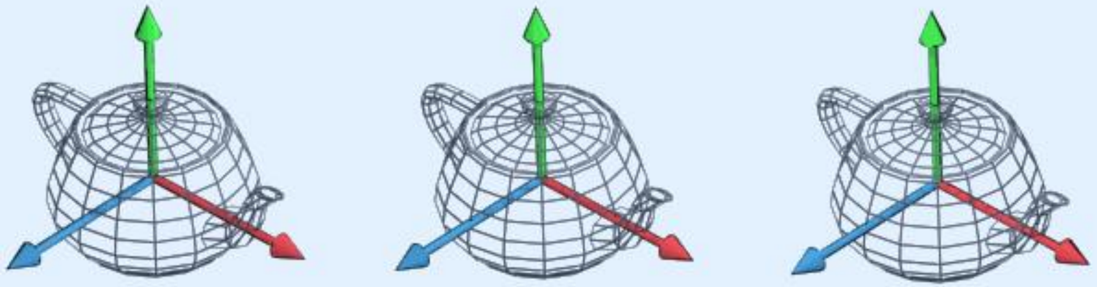
World – the coordinate system of the virtual environment

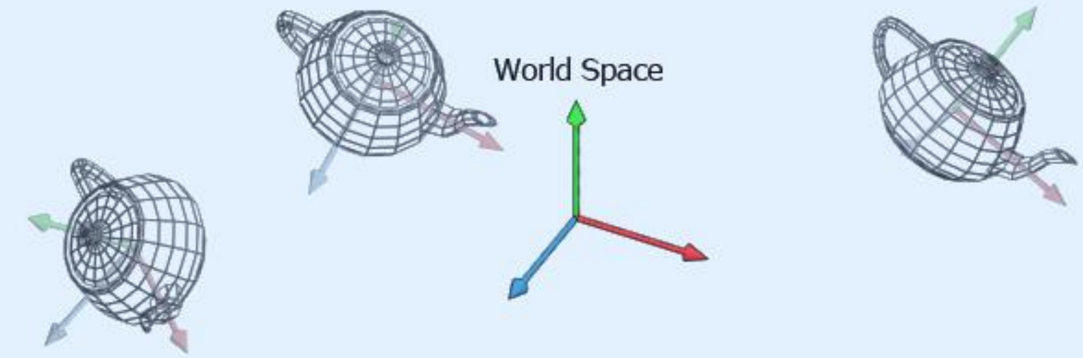
View/Camera – the coordinate system relative to the camera

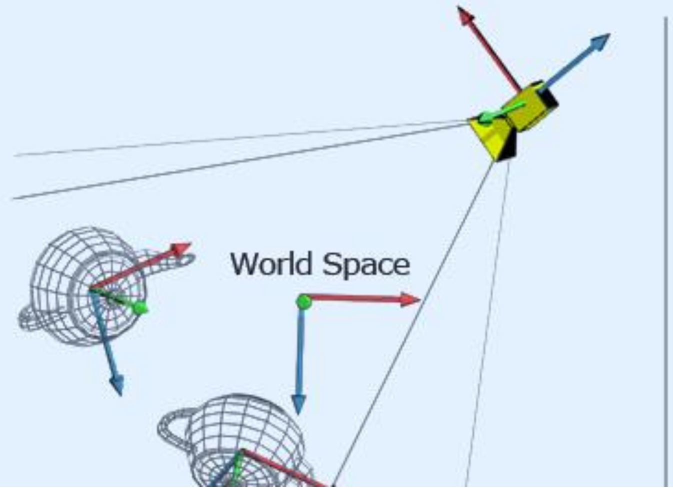
Clip – windowing system

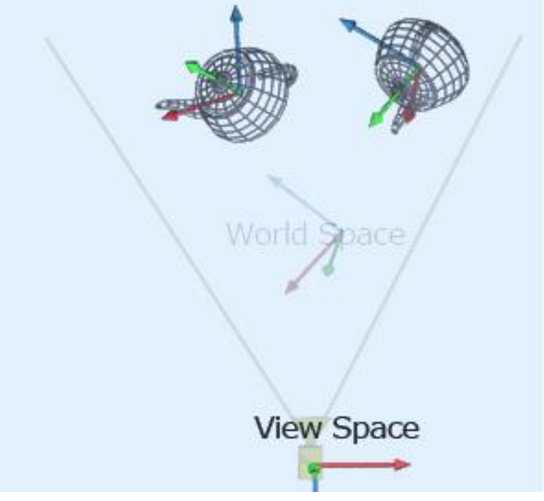
- **Local/Object Space**: The cube is centred at (0,0,0) with vertices defining its shape.
- **World Space**: The cube is translated, rotated, or scaled to a specific position in the scene.
- **View/Camera Space**: The scene is viewed from the camera’s perspective, as if the camera is the centre of the universe.
- **Clip Space**: The cube is projected into a normalized space, making it ready for rasterization and rendering.

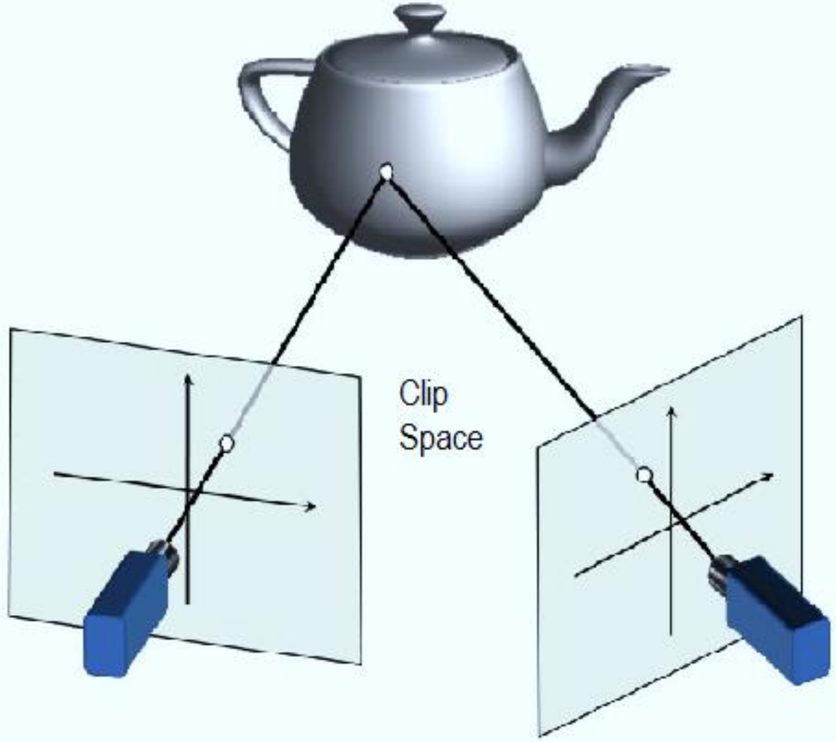
We use mathematics to transform vertices from one space to another




World Space


World Space


World Space
View Space


Clip
Space