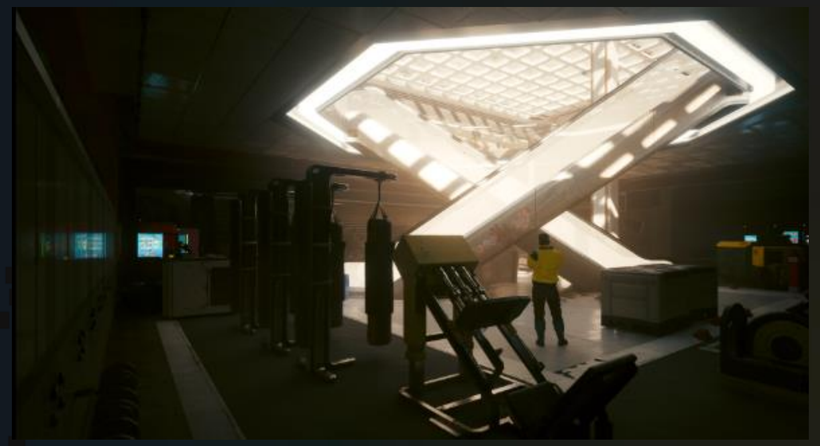
# Real-Time Ray Tracing

## The future of Gaming Graphics


**Group Members:**
**Ayiza Waqar(44529)**
**Samreen Bibi(46484)**


# Overview



1 Introduction to Real-Time Tracing

2 Background And Evolution

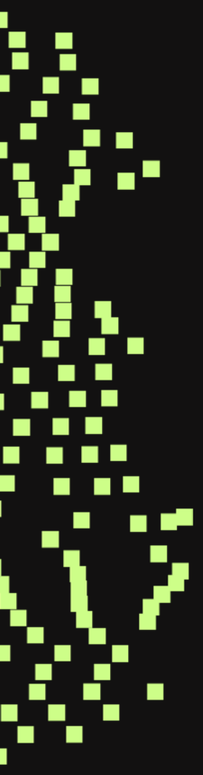
3 Challenges & Real-World Applications

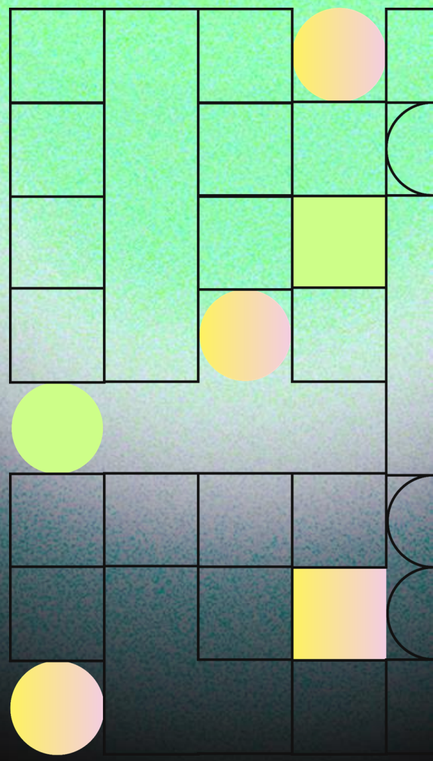
# Introduction

**What is ray graphics?**
Graphics technique that simulates **real light behavior**
shadows, reflections, refraction, and global illumination.
Traditionally used in Hollywood rendering farms; now entering **real-time gaming**.

**Why is it important today?**
Modern games demand realism.
GPUs now include dedicated **ray-tracing hardware**.
AI upscalers & denoisers make ray tracing fast enough for gameplay.

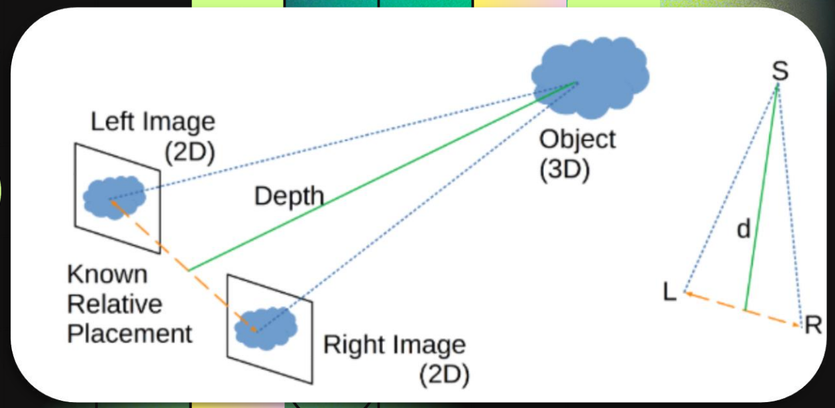
**Link to Parallel Computing:**
**“Ray tracing is connected to parallel computing because each ray, each pixel, and each light interaction can run on thousands of GPU cores simultaneously.”**





## Core Concepts

# Background




**Key Terms**
**Ray:** A virtual line simulating light.
**Bounce:** Reflection/refraction path.
**BVH (Bounding Volume Hierarchy):** Tree structure speeding up intersection tests.
**Global Illumination:** Light bouncing off surfaces realistically.

**Real-Life Analogy**
Think of shining a **laser pointer** into a room:
It hits surfaces
Bounces
Creates reflections, shadows
Ray tracing does the same , but for **every pixel on the screen**.

# How It Works

- **Primary ray** fired from camera for each pixel
- Ray hits object ↠ check material (metal, glass, wood)
- Spawn **secondary rays** (reflection/refraction)
- Rays accumulate light contributions
- GPU **denoiser** + **upscaler** cleans the image
- Final frame is displayed

**Where Parallelism Happens**
Millions of rays calculated **simultaneously**
Thousands of GPU cores run intersection tests in parallel
BVH traversal is parallel on warps/wavefronts
AI denoising uses tensor cores in parallel.

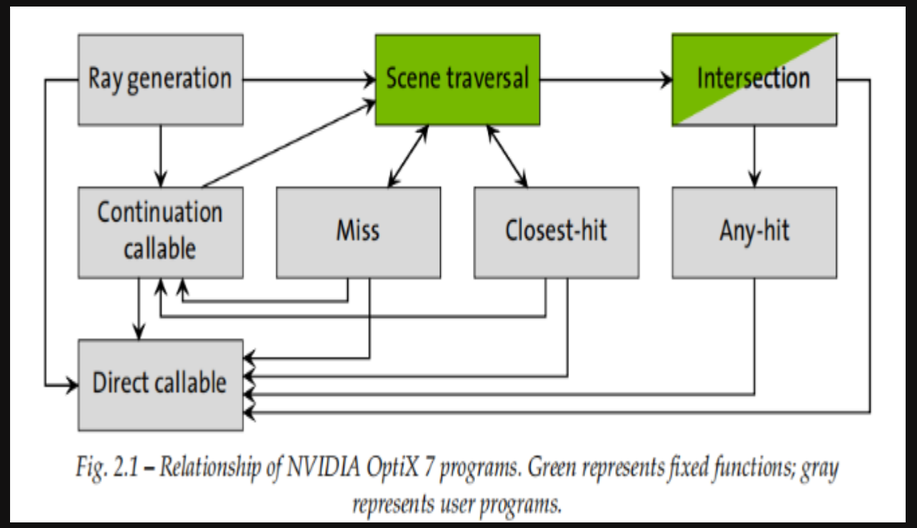
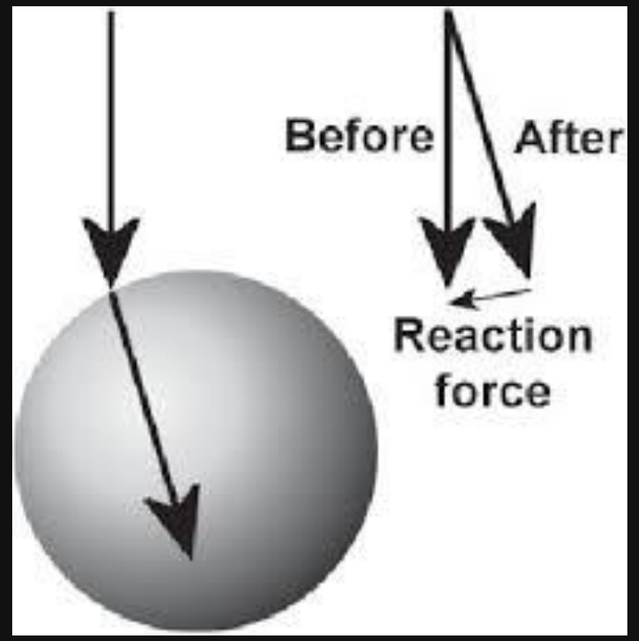


*Fig. 2.1 – Relationship of NVIDIA OptiX 7 programs. Green represents fixed functions; gray represents user programs.*

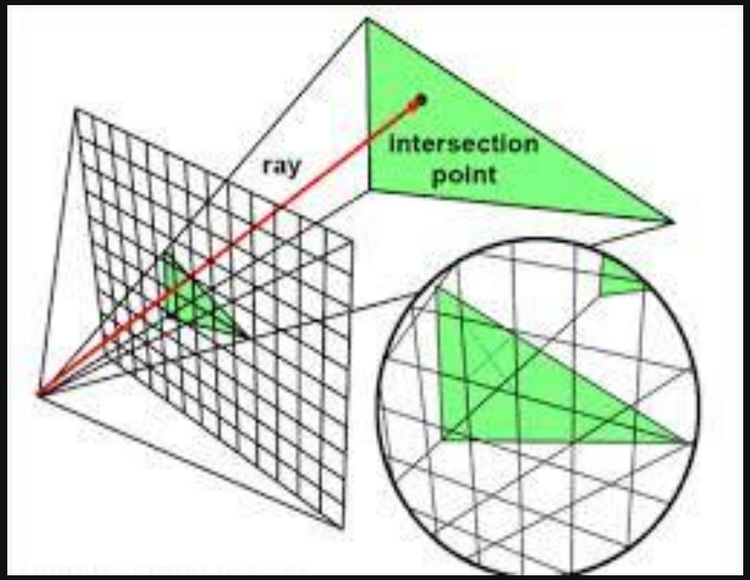
# Working-Light Ray Simulation

- Ray tracing imitates how real-world light behaves.
- Light rays travel through the scene and interact with objects.
- Rays bounce or pass through surfaces(refraction).
- Each bounce changes the ray’s color and intensity.
- These interactions create realistic shadows and highlights.
- Final pixel color is computed based on all ray contributions.



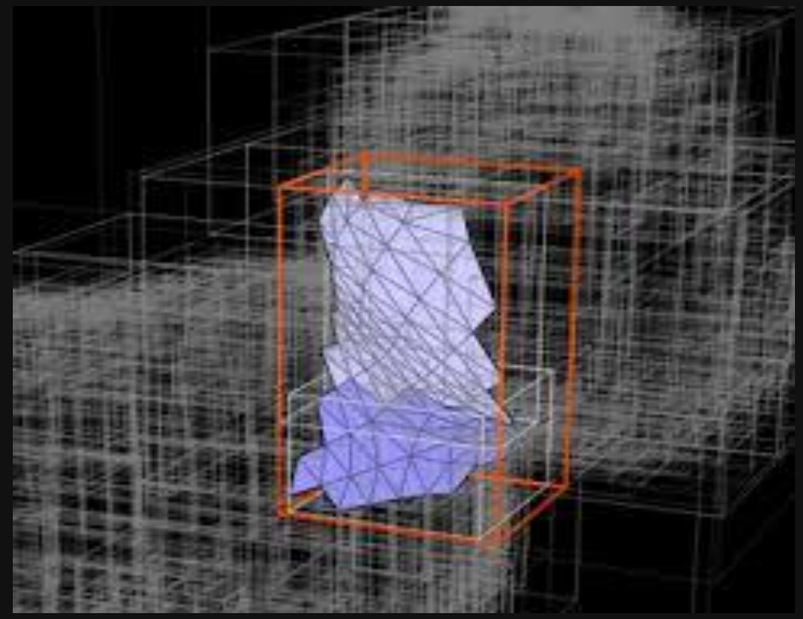
# Working-Hybrid Rendering Pipeline

- Games don’t use full ray tracing alone (it’s too slow).
- Rasterization handles fast geometry drawing.
- Ray tracing handles lighting, shadows, and reflections.
- Both techniques run together for better performance.
- Hybrid rendering gives speed + realism at the same time.
- Final image is produced by combining raster + ray-traced effects.



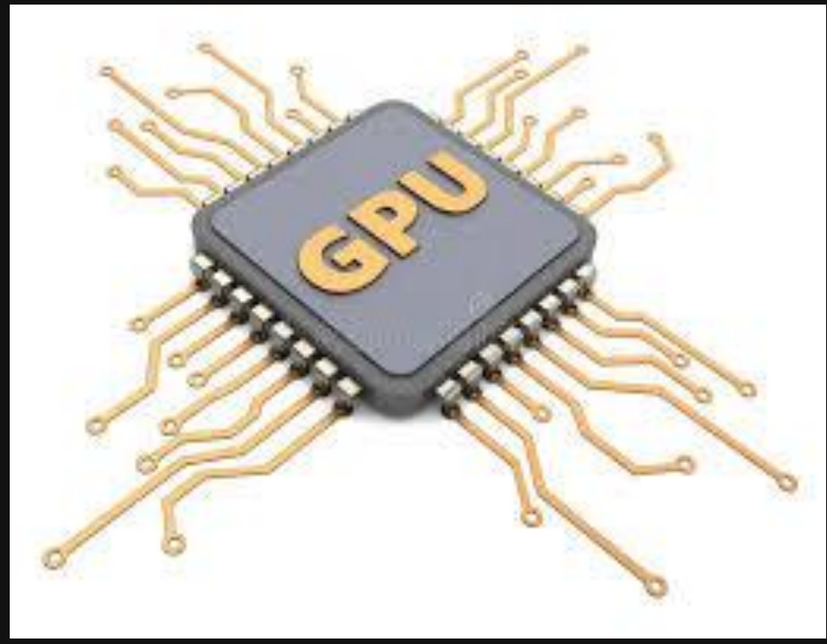
# Working-Acceleration Structures (BVH)

- The scene is divided into a tree-like structure called BVH.
- Objects are grouped inside bounding boxes.
- Rays only check collisions with relevant boxes, not every object.
- This reduces unnecessary calculations and boosts speed.
- BVH is the key optimization for real-time ray tracing.
- Faster traversal = smoother, more efficient rendering.

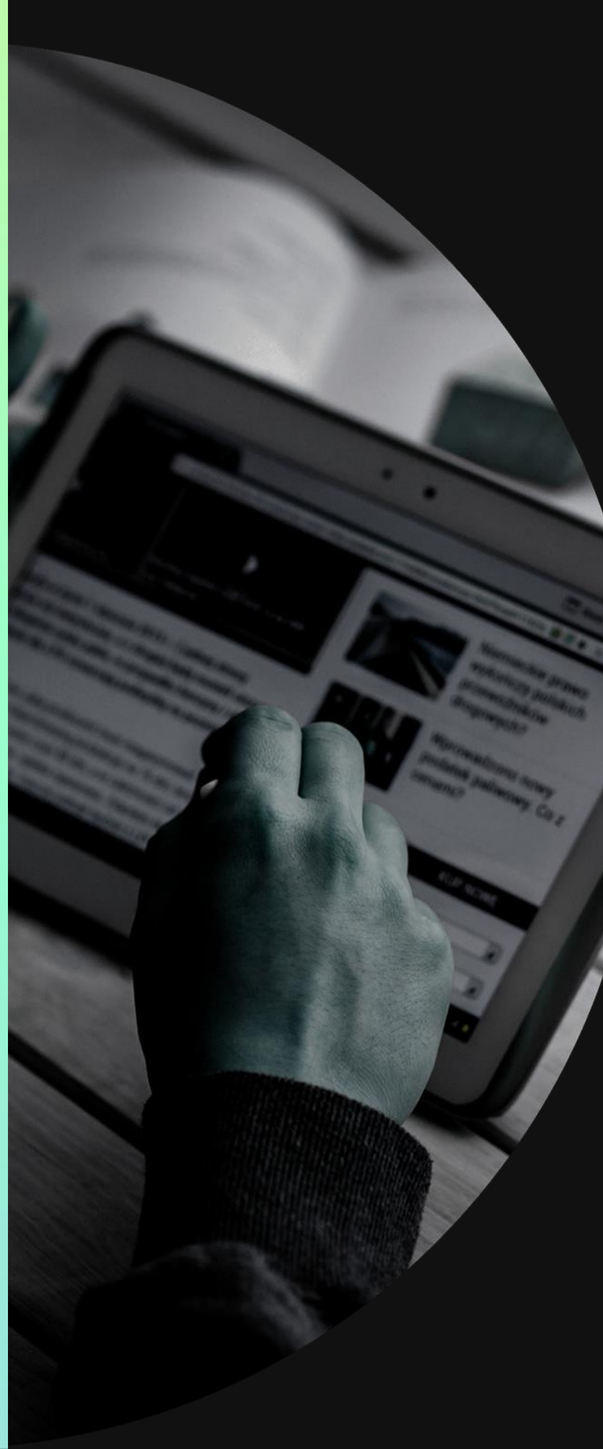


# Working GPU RT Cores (Hardware Support)

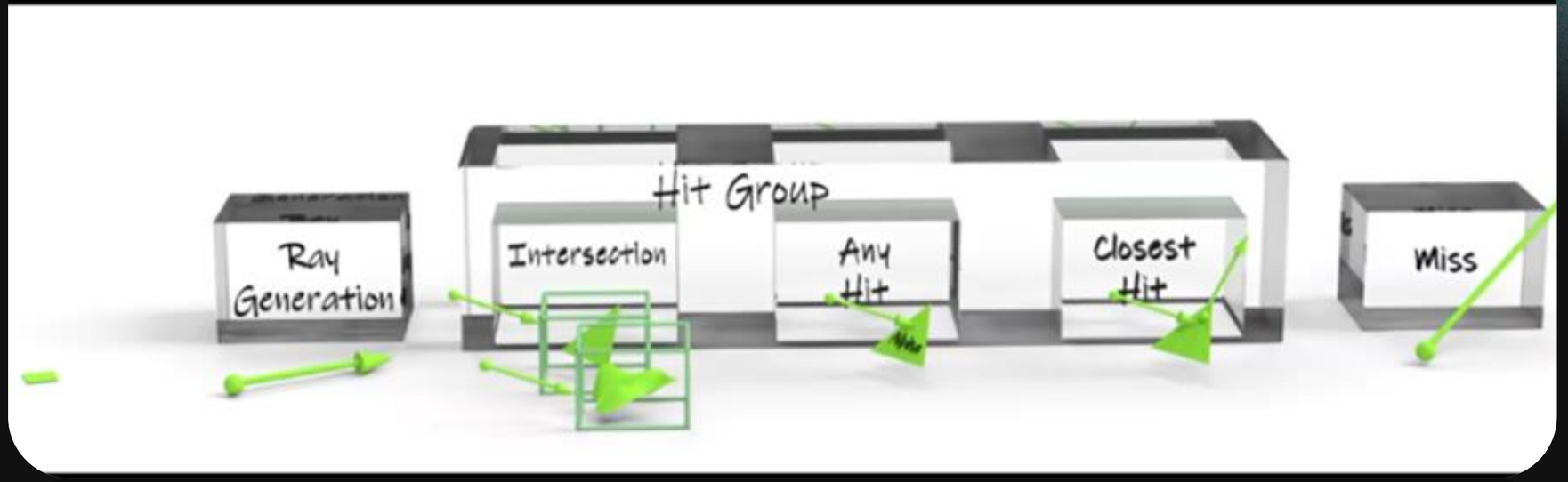
- Modern GPUs like NVIDIA RTX contain dedicated RT cores.
- These cores are built specifically for ray-tracing calculations.
- Each ray is processed in parallel for maximum speed.
- This reduces the load on normal GPU shaders.
- Hardware acceleration makes real-time ray tracing possible.
- Result: smoother gameplay with realistic lighting effects.



# Architecture Diagram



# RAYTRACING SHADERS



# Hardware Acceleration Replaces Software Emulation

## Turing SM


## Shaders

Launch Ray Probe

Shading

## RT Core

Fetch box
Decode box
Intersection test
Sub-box or tris?

Ray/triangle intersection test

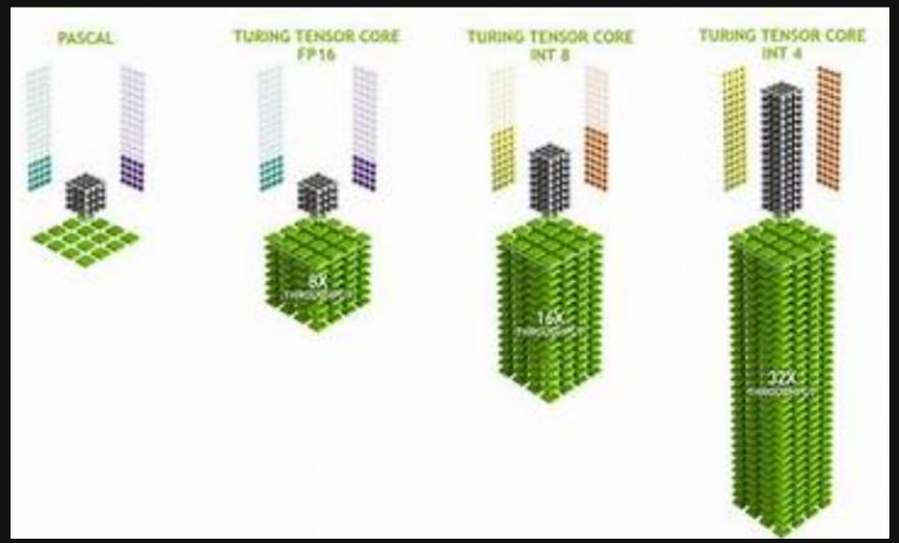
Return hit

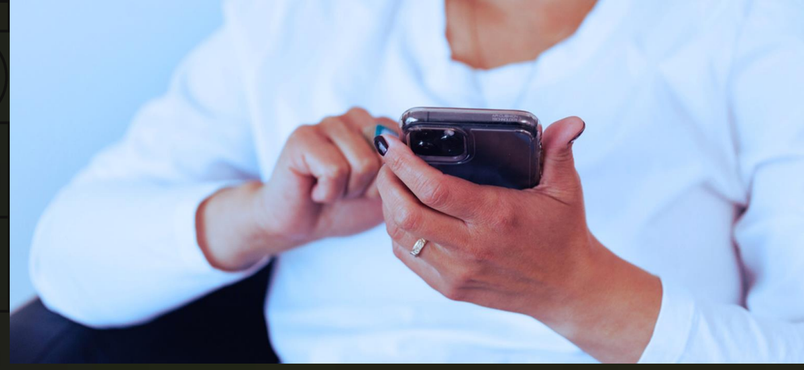
Box Intersection Evaluators

Triangle Intersection Evaluators

# GPU Hardware Blocks Involved

- **RT Cores:** Accelerate ray–triangle intersection
- **Tensor Cores:** AI denoising/super-resolution
- **CUDA/Shader Cores:** Material shading, lighting
- **Texture Units:** Fetch color data
- **Parallel Threads (SIMT):** Execute thousands of ray operations





# Real-World Applications

# Real-World Applications

**1. Gaming Industry**

Used in AAA games for realistic lighting, shadows, reflections (Cyberpunk, Fortnite, GTA VI).

**2. Film & Animation Pre-Visualization**

Directors preview movie-quality lighting in real time during filming and scene setup.

**3. Architecture & Interior Design**

Designers preview true-to-life lighting inside buildings before construction.

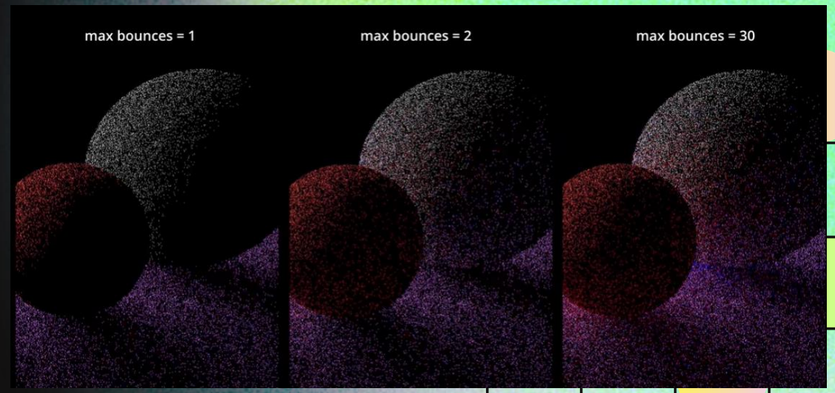
**4. Automotive Industry (Car Design)**

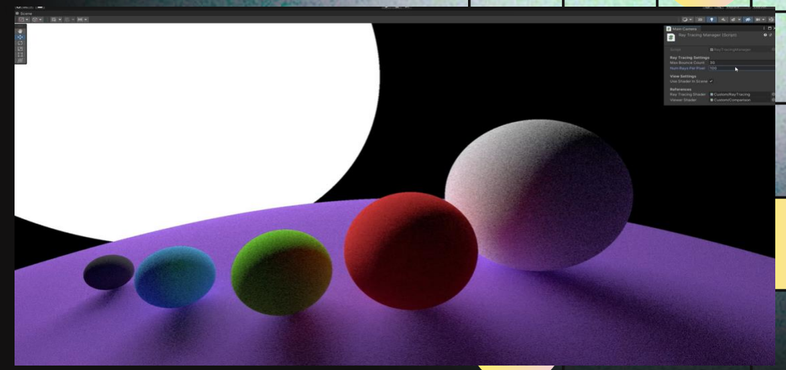
Car companies use real-time ray tracing to visualize paint, metal reflections, and headlight behavior.

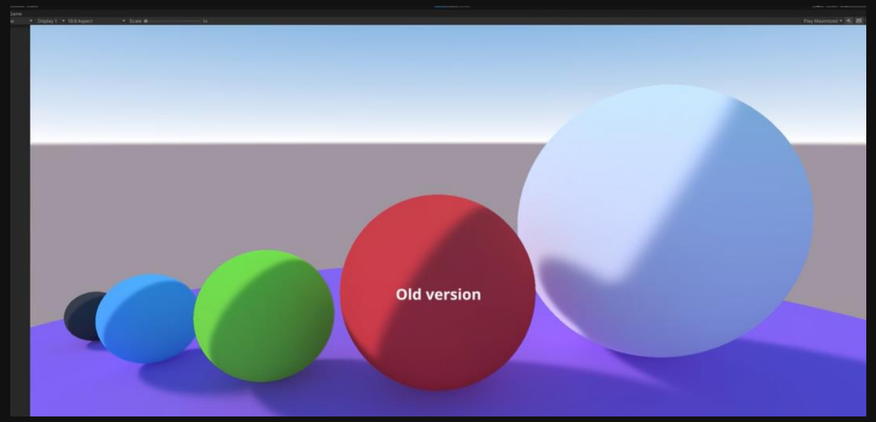
**1. Hyper-Realistic Lighting & Shadows**
- Simulates natural light behavior for lifelike scenes
- Soft, accurate shadows that change with object movement
- Used in games like **Cyberpunk 2077**, **Minecraft RTX**, **Battlefield V**

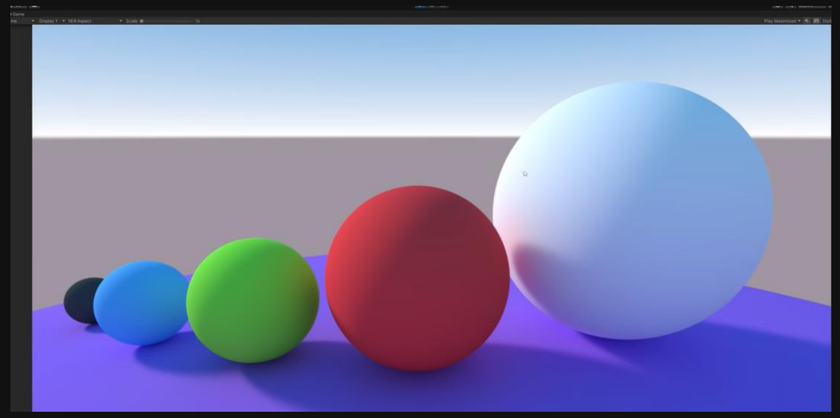
**2. Accurate Reflections & Refractions**
- Mirrors, water, glass, metals show true reflections
- Complex reflective surfaces look far more realistic
- Enhances immersion in open-world & sci-fi games

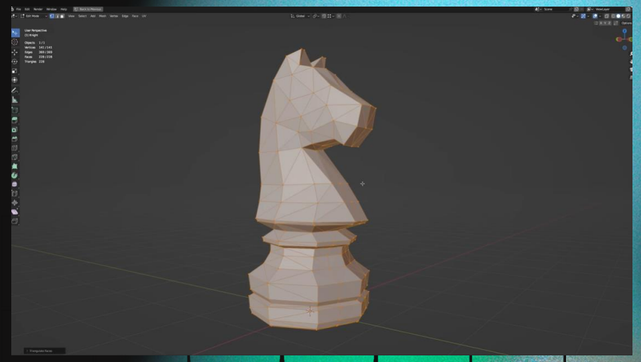
**3. Cinematic Visual Quality**
- Movie-level rendering inside gameplay
- High-fidelity cutscenes and in-game visuals blend seamlessly
- Enables photorealistic characters and environments

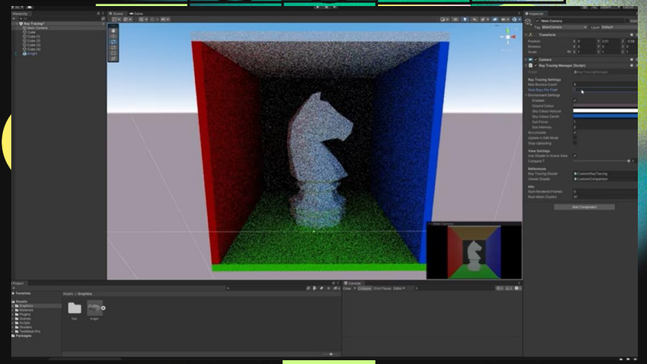


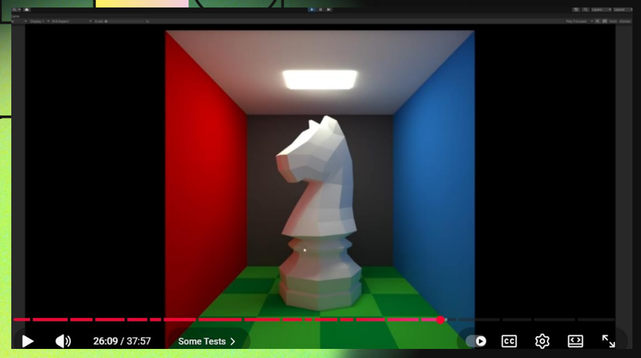












**4. Realistic Ambient Occlusion**
Precise shading in corners, edges, and tight spaces
Adds depth and realism to indoor and architectural scenes
Improves visual storytelling and atmosphere

**5. Enhanced VR & AR Gaming**
Accurate lighting makes VR worlds feel more natural
Depth, shadows, and reflections increase immersion
Essential for training simulations and realistic virtual worlds

**6. Game Development & Level Design**
Designers preview lighting in real-time, reducing rendering time
Faster iteration cycles ↠ more detailed world-building
Helps maintain visual consistency across all environments

**7. Film-Game Convergence**
Ray-traced real-time engines (Unreal Engine, Unity HDRP) used for:
Virtual production
Cinematics
Hybrid movie–game experiences
Reduces the gap between playable and cinematic graphics

```
// Raycast against all meshes and keep info about the closest hit
for (int meshIndex = 0; meshIndex < NumMeshes; meshIndex ++)
{
    MeshInfo meshInfo = AllMeshInfo[meshIndex];
    // Skip the mesh if ray doesn't intersect its bounding box
    if (!RayBoundingBox(ray, meshInfo.boundsMin, meshInfo.boundsMax))
    {
        continue;
    }

    for (int i = 0; i < meshInfo.numTriangles; i ++)
    {
        int triIndex = meshInfo.firstTriangleIndex + i;
        Triangle tri = Triangles[triIndex];
        HitInfo hitInfo = RayTriangle(ray, tri);

        if (hitInfo.didHit && hitInfo.dst < closestHit.dst)
        {
            closestHit = hitInfo;
            closestHit.material = meshInfo.material;
        }
    }
}
```


# 1. AAA Games

- Cyberpunk 2077 (Full RT Overdrive)
- Fortnite RTX
- Metro Exodus
- Spiderman Remastered

# 2.Editing / Film

- Unreal Engine Virtual Production
- Real-time previews for VFX

# 3. Case Study

**Minecraft RTX**

- Originally blocky graphics
- With ray tracing ➳ realistic sunlight, reflections, soft shadows
- Demonstrates how even simple engines transform with RT

# Advantages & Limitations

| Advantages | Limitations |
| --- | --- |
| Ultra-realistic lighting | Very computationally expensive |
| Accurate reflections/shadows | Needs high-end GPUs or cloud servers |
| Film-like global illumination | Requires hybrid rendering for 60+ FPS |
| Works consistently in all scenes (physically correct) | Noise at low sample counts (needs AI denoising) |

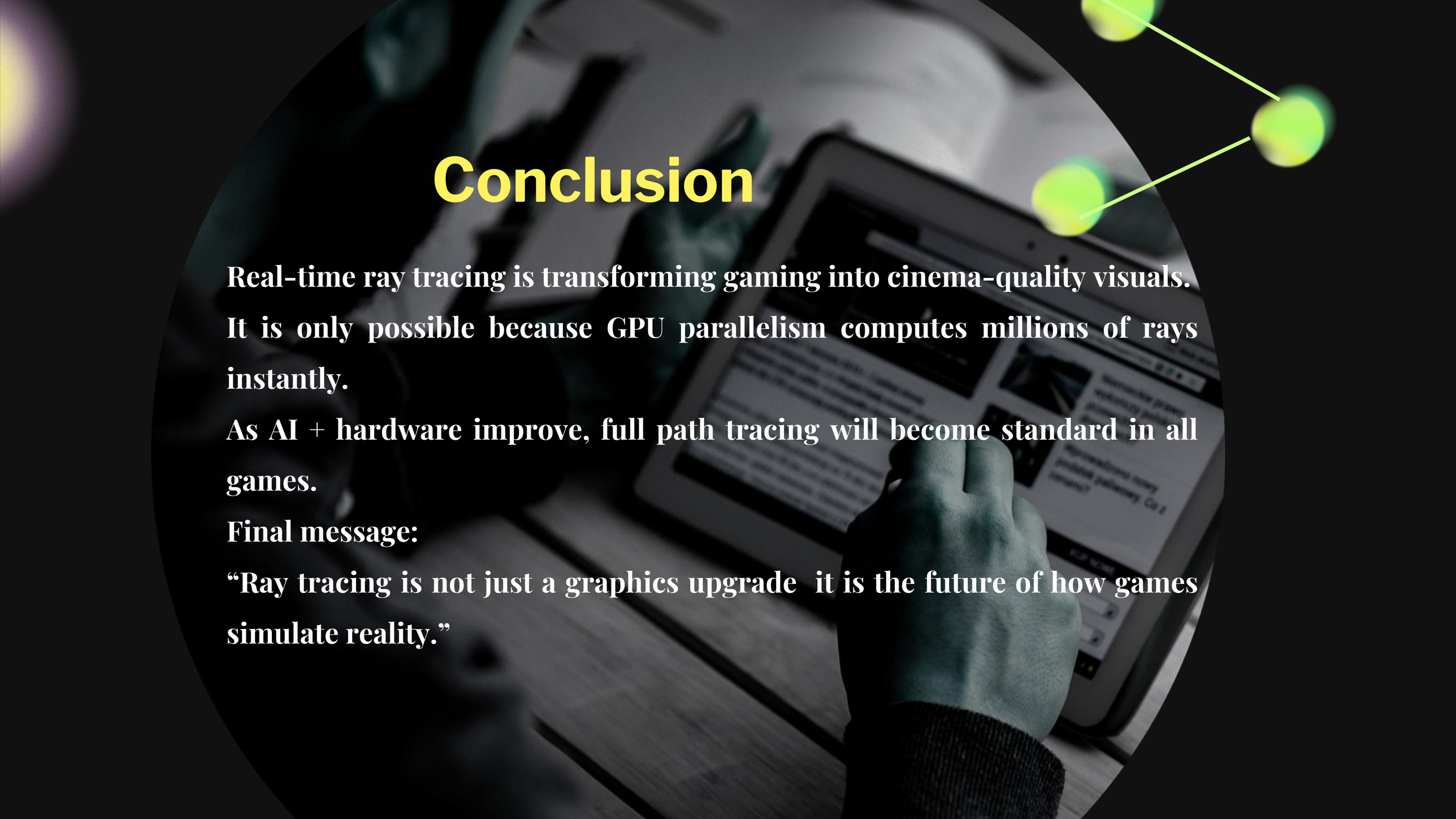
# Future Directions

Full path tracing in all games


More efficient RT hardware

Real-time movie quality graphics


AI that predicts lighting without rays (reduced cost)



# Conclusion

Real-time ray tracing is transforming gaming into cinema-quality visuals.
It is only possible because GPU parallelism computes millions of rays instantly.
As AI + hardware improve, full path tracing will become standard in all games.
Final message:
“Ray tracing is not just a graphics upgrade  it is the future of how games simulate reality.”

# Thank You!

**Do you have any Question?**