TÉCNICO LISBOA

COMPUTER GRAPHICS FOR GAMES

P2 2025-26

Carlos Martinho


Epic Games
Student Showcase 2024

“WHAT I CANNOT CREATE, I DO NOT UNDERSTAND.”

Richard Feynman

OUTLINE

- Course introduction
- Importance of Computer Graphics
- Brief history of Computer Graphics
- Quarter planning

GOALS

GOALS

- Develop a real-time graphics engine ‘from scratch’ using modern 3D graphics programming with GPU.
(*technical challenge*)

- Understand the theory and the algorithms underlying modern 3D graphics systems.
(*intellectual challenge*)

- Development will be in Modern C++ / OpenGL / GLSL.

- Help creating a strong portfolio.

PREREQUISITES

PREREQUISITES

- Linear algebra (review in the context of Computer Graphics and addenda next lectures).
- Proficient object programming skills. Good knowledge of C++ and STL will help.
- Knowledge of Computer Graphics (e.g. image formats, raster algorithms, etc.) may help.

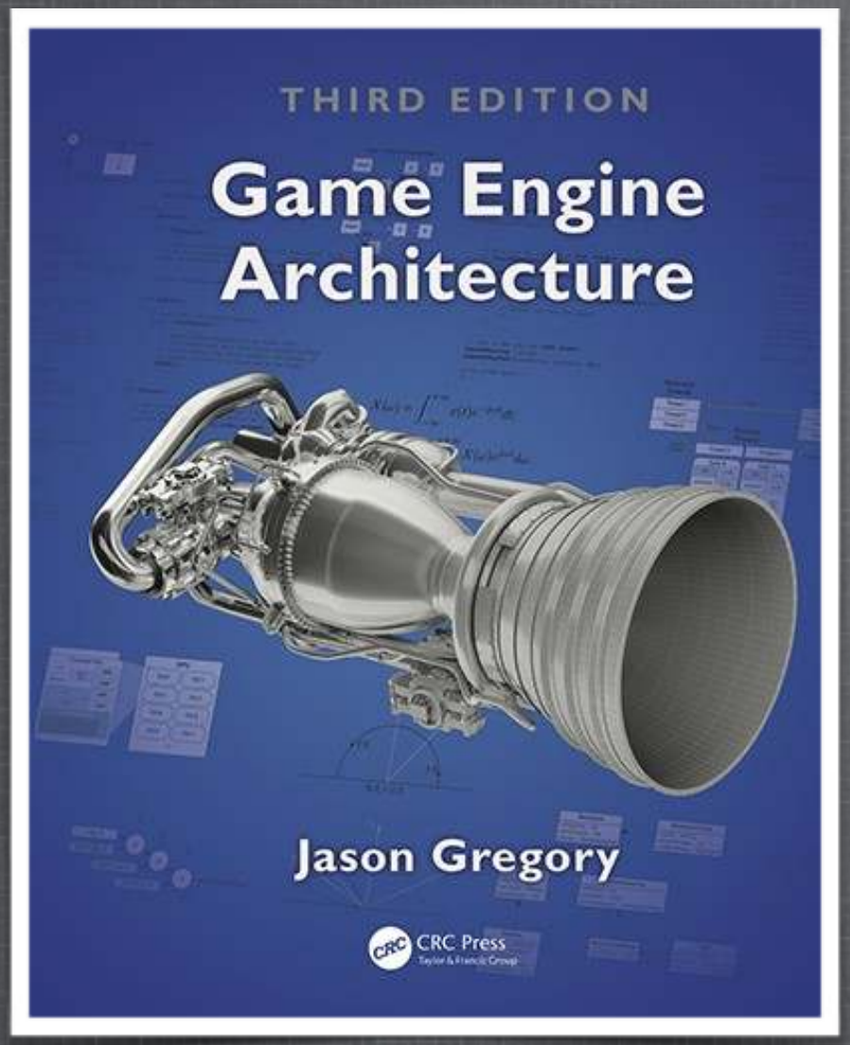
GRADING

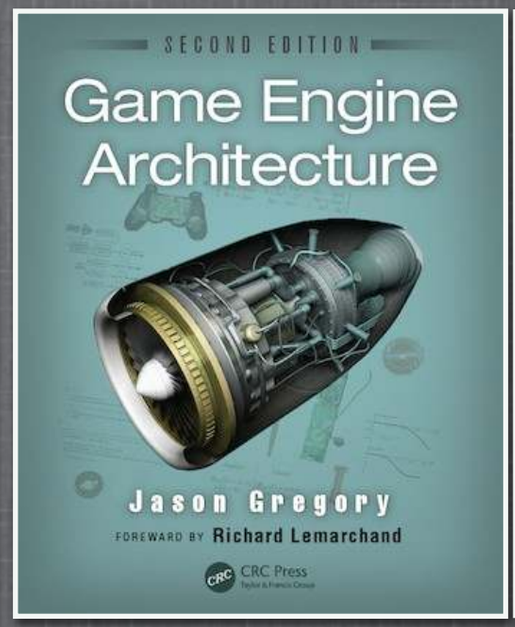
BASE PROJECT
(T2, 40%, >9.5)

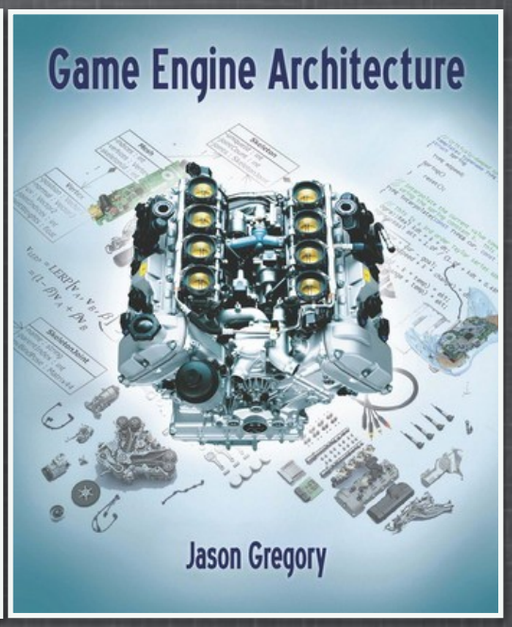
+

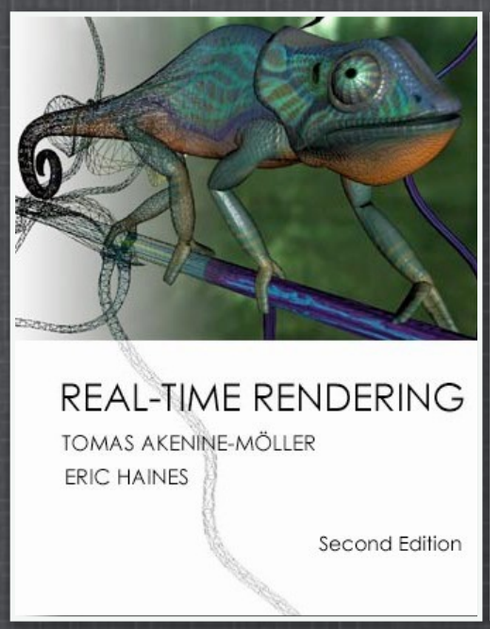
SHADER PROJECT
(T1, 60%, >9.5)

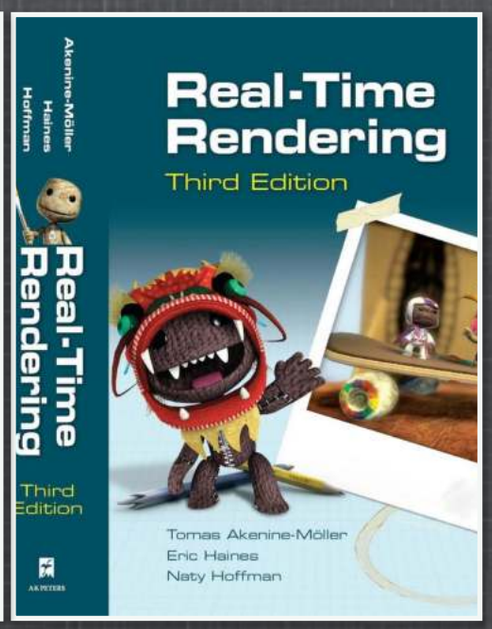
BIBLIOGRAPHY

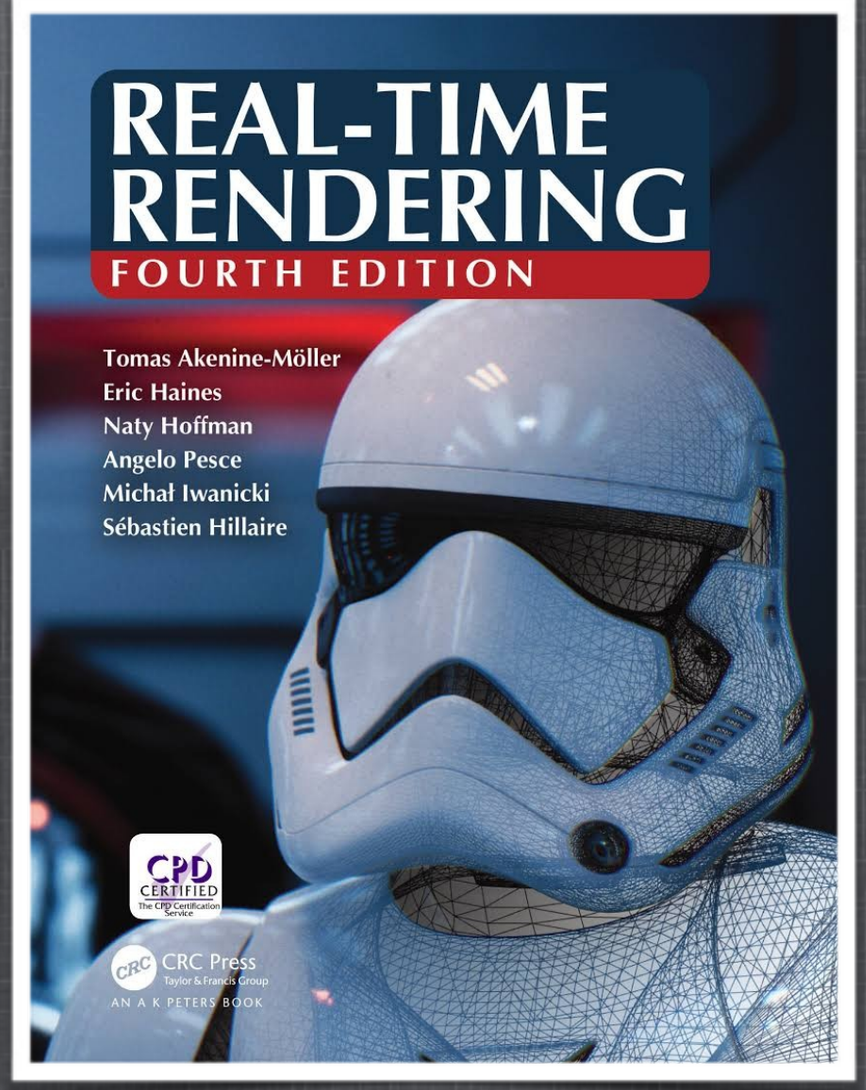

THIRD EDITION
Game Engine Architecture
Jason Gregory
CRC Press
Taylor & Francis Group


SECOND EDITION
Game Engine Architecture
Jason Gregory
FOREWARD BY Richard Lemarchand
CRC Press


Game Engine Architecture
Jason Gregory


REAL-TIME RENDERING
TOMAS AKENINE-MÖLLER
ERIC HAINES
Second Edition

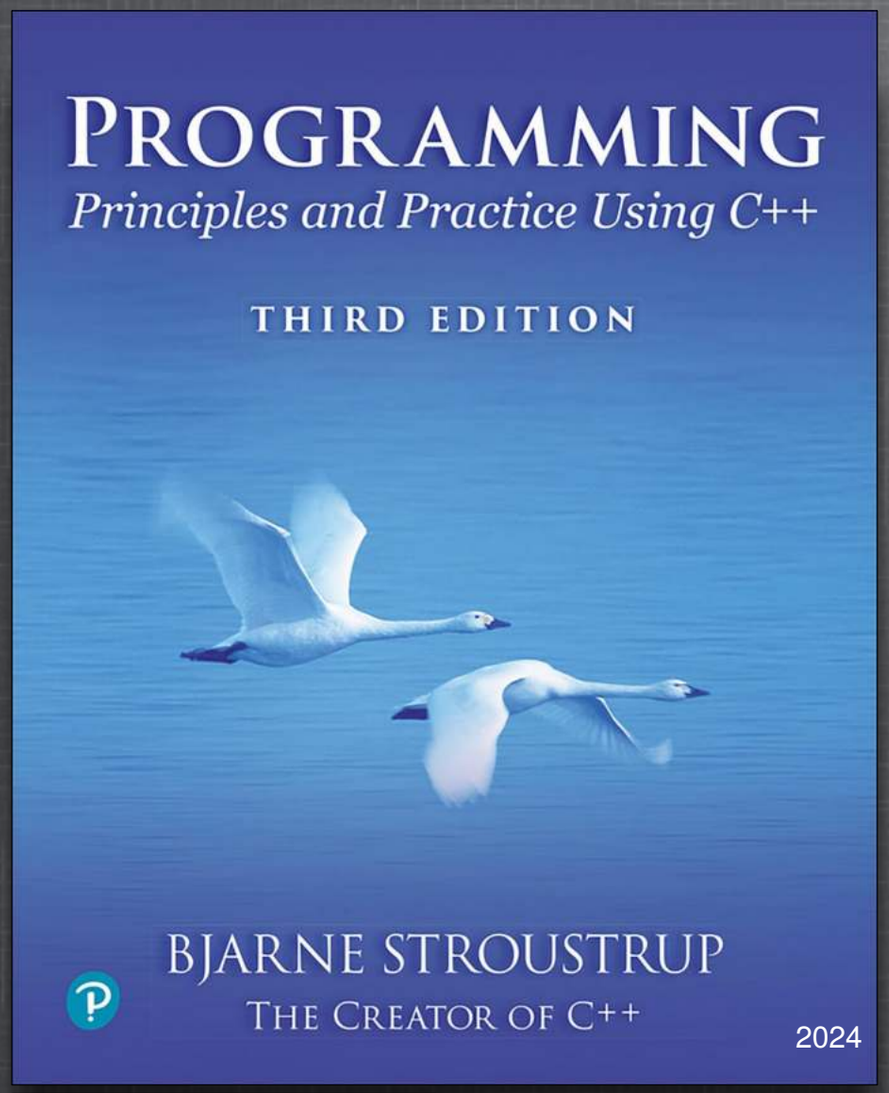

Akenine-Möller
Haines
Hoffman
Real-Time Rendering
Third Edition
Real-Time Rendering
Third Edition
Tomas Akenine-Möller
Eric Haines
Naty Hoffman
A K PETERS

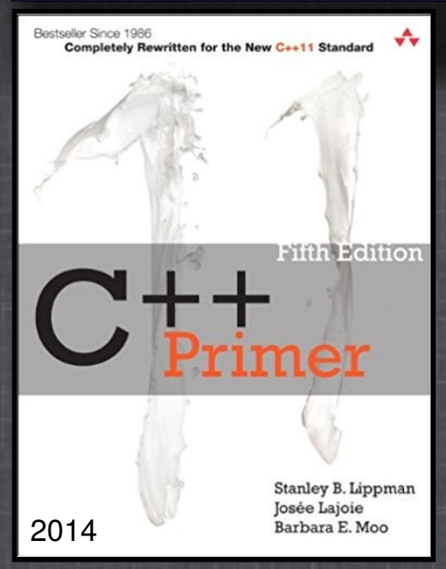

REAL-TIME RENDERING
FOURTH EDITION
Tomas Akenine-Möller
Eric Haines
Naty Hoffman
Angelo Pesce
Michał Iwanicki
Sébastien Hillaire
CPD CERTIFIED
The CPD Certification Service
CRC Press
Taylor & Francis Group
AN A K PETERS BOOK

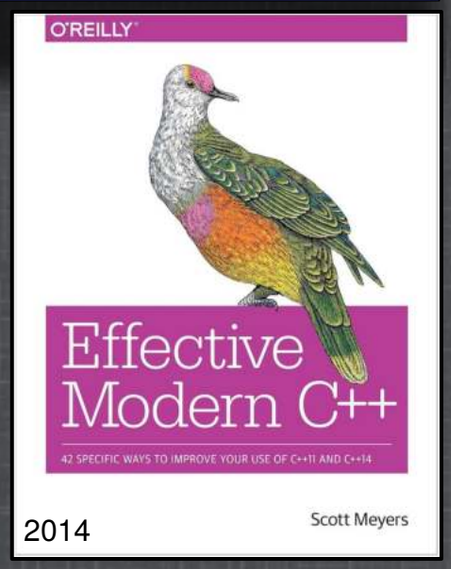
2004-2007

2002-2009

2010-2016

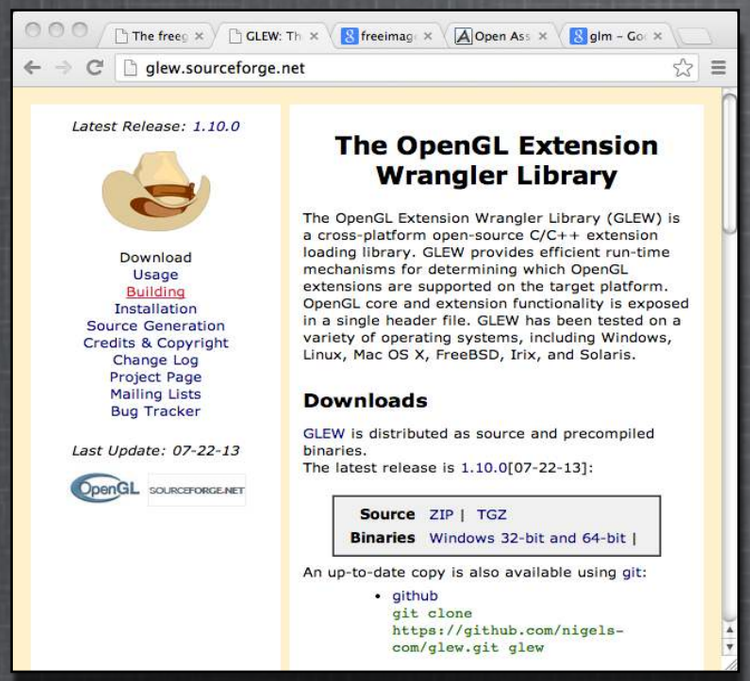

PROGRAMMING
Principles and Practice Using C++
THIRD EDITION
BJARNE STROUSTRUP
THE CREATOR OF C++
2024

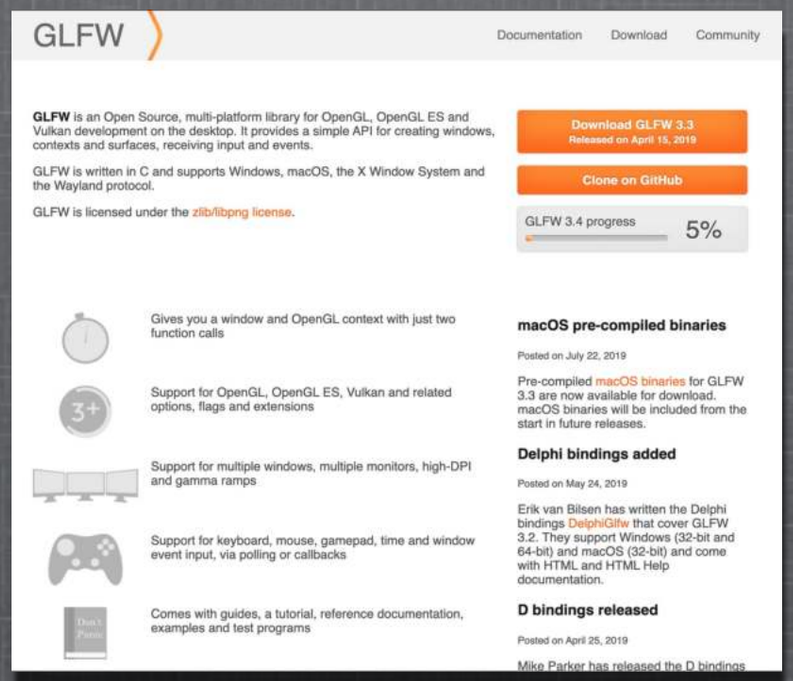

Bestseller Since 1986
Completely Rewritten for the New C++11 Standard
Fifth Edition
C++ Primer
Stanley B. Lippman
Josée Lajoie
Barbara E. Moo
2014

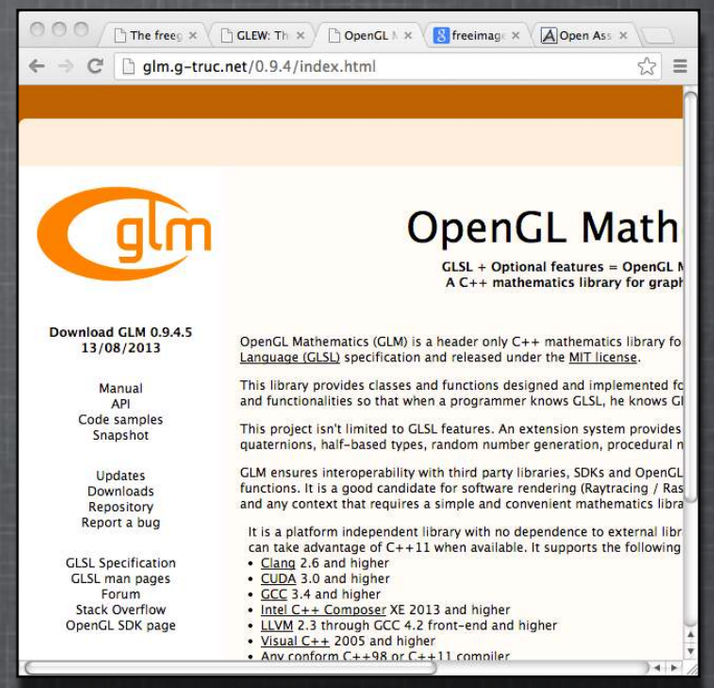

O'REILLY
Effective Modern C++
42 SPECIFIC WAYS TO IMPROVE YOUR USE OF C++11 AND C++14
Scott Meyers
2014

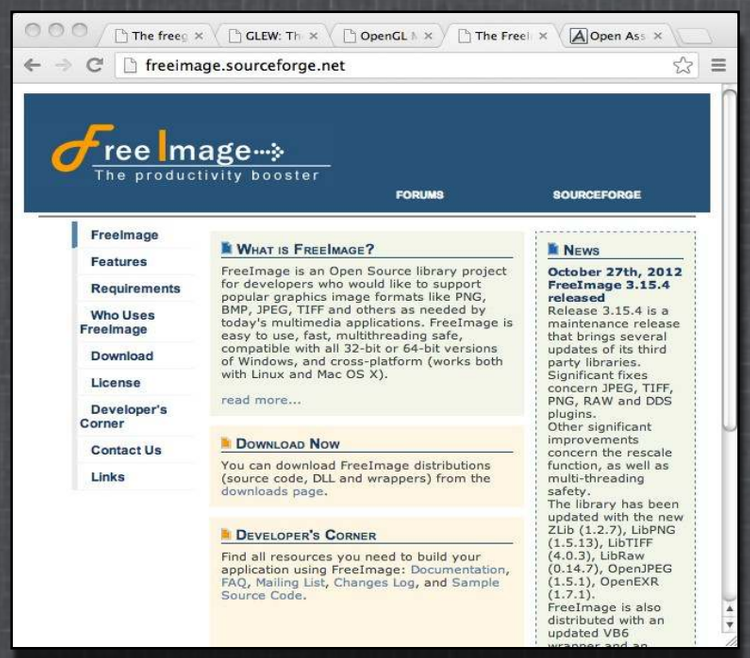

cppreference.com
Create account
Page
Discussion
Standard revision: Diff
C++ reference
C++11, C++14, C++17, C++20, C++23, C++26 | Compiler support C++11, C++14, C++17, C++20,
Language
Keywords – Preprocessor
ASCII chart
Basic concepts
Comments
Names (lookup)
Types (fundamental types)
The main function
Expressions
Value categories
Evaluation order
Operators (precedence)
Conversions – Literals
Statements
if – switch
for – range-for (C++11)
while – do-while
Declarations – Initialization
Functions – Overloading
Classes (unions)
Templates – Exceptions
Freestanding implementations
Standard library (headers)
Named requirements
Feature test macros (C++20)
Language – Standard library
Language support library
Program utilities
Signals – Non-local jumps
Basic memory management
Variadic functions
source_location (C++20)
Coroutine support (C++20)
Comparison utilities (C++20)
Type support – type_info
numeric_limits – exception
initializer_list (C++11)
Concepts library (C++20)
Diagnostics library
Assertions – System error (C++11)
Exception types – Error numbers
basic_stacktrace (C++23)
Debugging support (C++26)
Memory management library
Allocators – Smart pointers
Memory resources (C++17)
Metaprogramming library (C++11)
Type traits – ratio
integer_sequence (C++14)
General utilities library
Function objects – hash (C++11)
Swap – Type operations (C++11)
Integer comparison (C++20)
pair – tuple (C++11)
optional (C++17)
expected (C++23)
variant (C++17) – any (C++17)
bitset – Bit manipulation (C++20)
Containers library
vector – deque – array (C++11)
list – forward_list (C++11)
map – multimap – set – multiset
unordered_map (C++11)
unordered_multimap (C++11)
unordered_set (C++11)
unordered_multiset (C++11)
Container adaptors
span (C++20) – mdspan (C++23)
Iterators library
Ranges library (C++20)
Range factories – Range adaptors
generator (C++23)
Algorithms library
Numeric algorithms
Execution policies (C++17)
Constrained algorithms (C++20)
Strings
Text pro
Numeric
Date an
Input/ou
Concurr
Executio
Technical specifications
Standard library extensions (library fundamentals TS)
resource_adaptor – invocation_type
Standard library extensions v2 (library fundamentals TS v2)
propagate_const – ostream_joiner – randint
observer_ptr – Detection idiom
Standard library extensions v3 (library fundamentals TS v3)
scope_exit – scope_fail – scope_success – unique_resource
Parallelis
simd
Concurr
Transact
Reflecti
External Links – Non-ANSI/ISO Libraries – Index – std Symbol Index

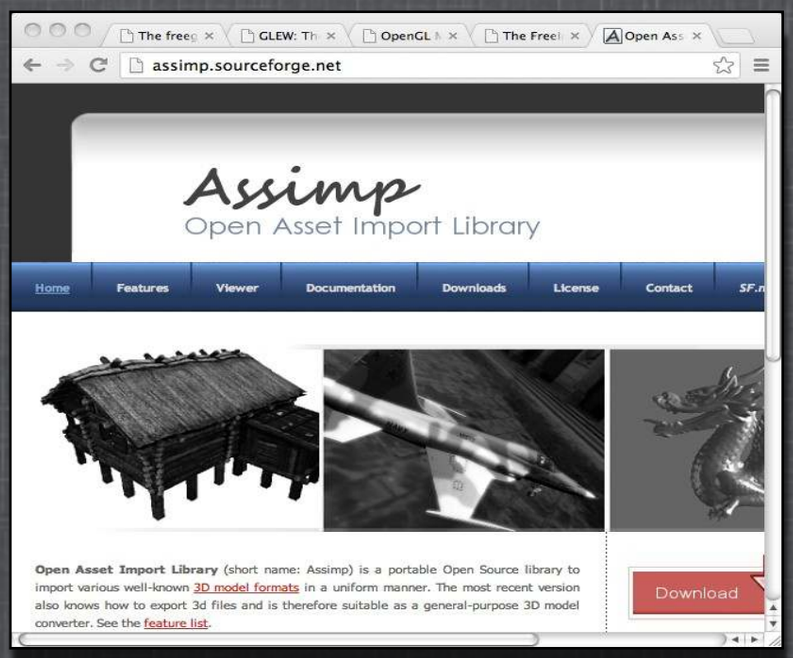
ONLINE LIBRARIES

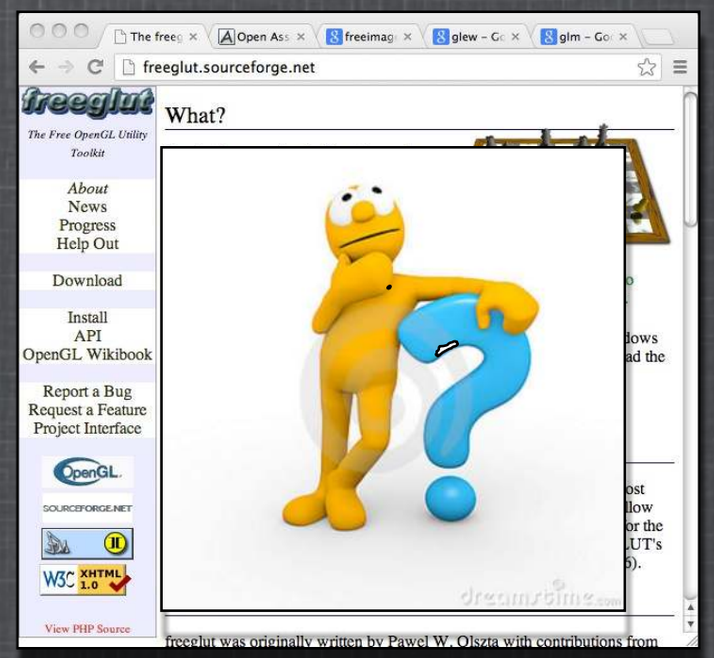




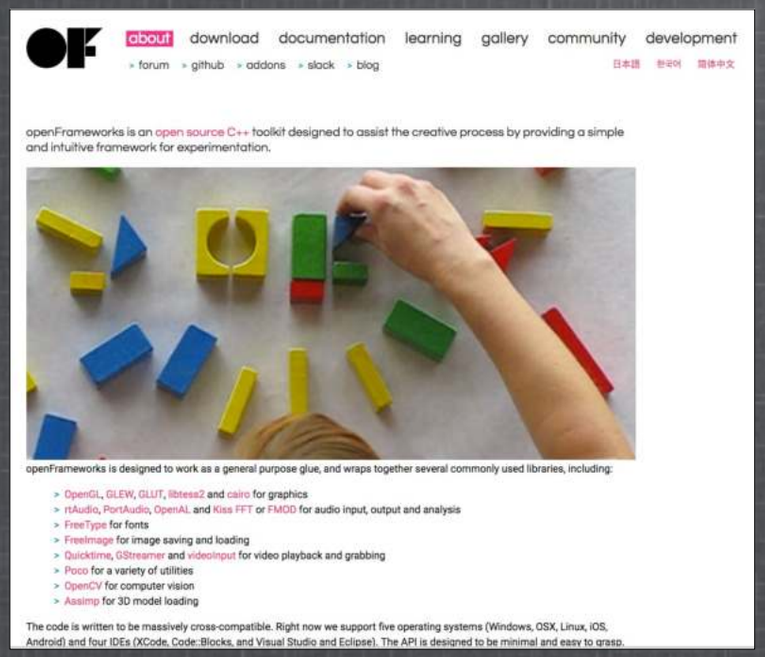




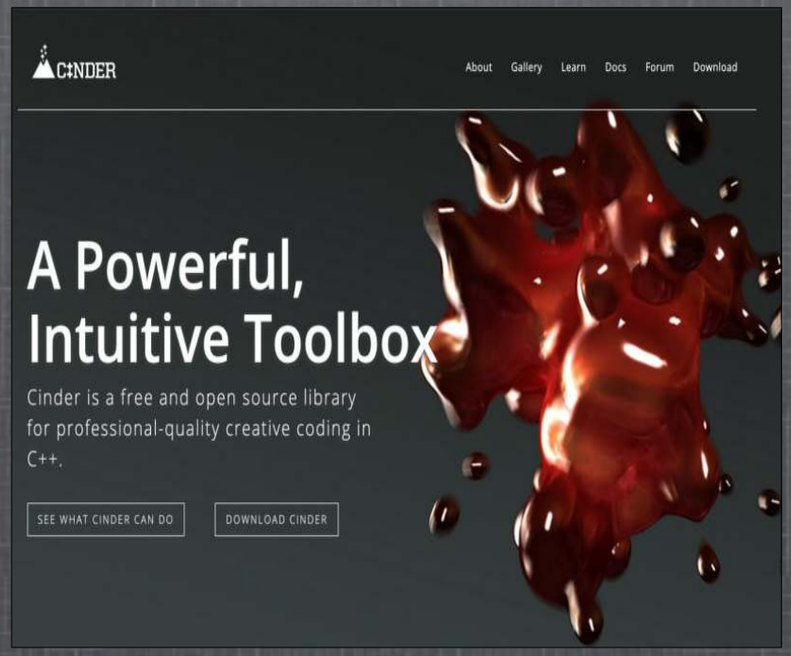




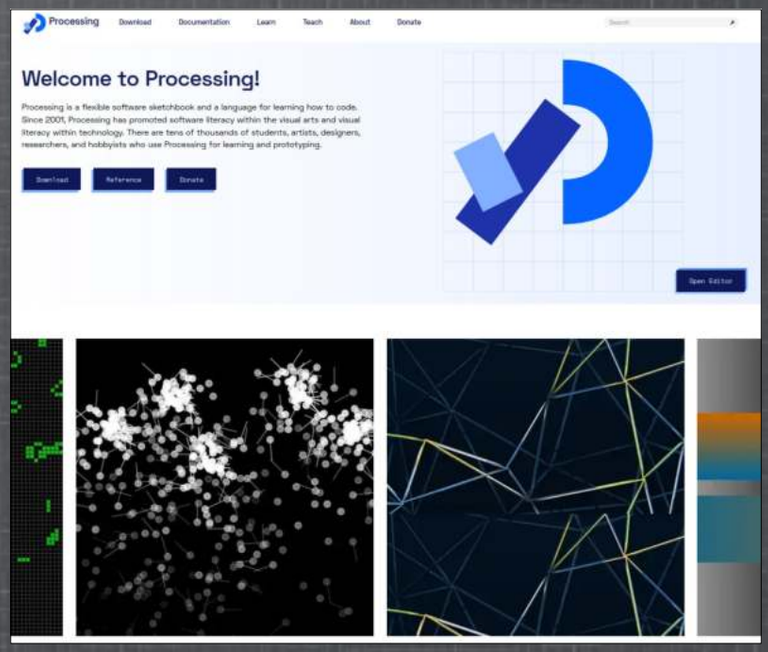
CREATIVE CODING



OPEN FRAMEWORKS



CINDER



PROCESSING

IMPORTANCE OF COMPUTER GRAPHICS

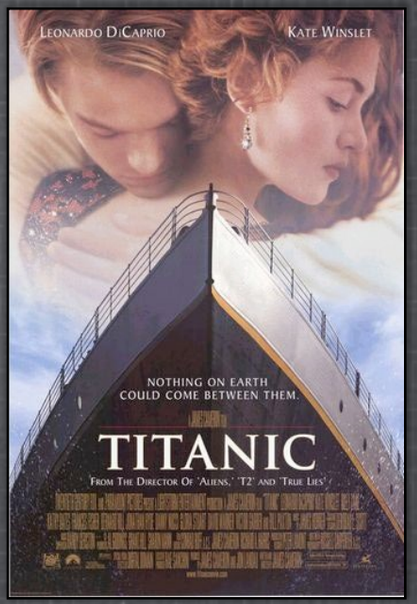
COMPUTER GRAPHICS IN MOVIES

(1982)

(1991)

(1993)

(1995)

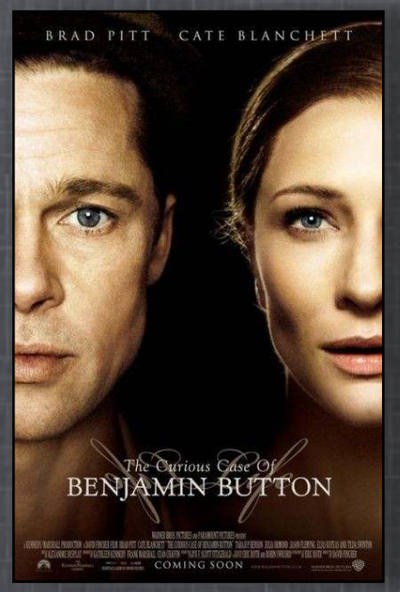


(1997)

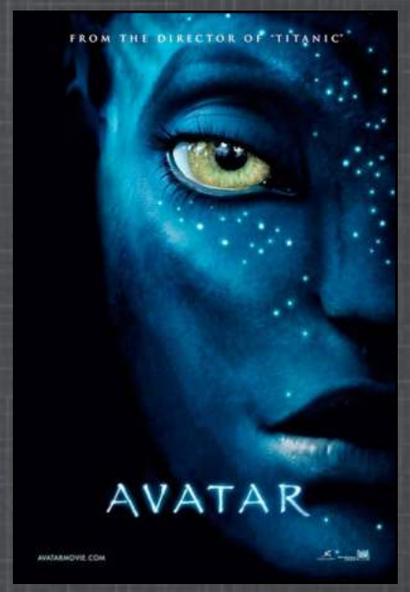
(1999,2002,2005)

(1999)

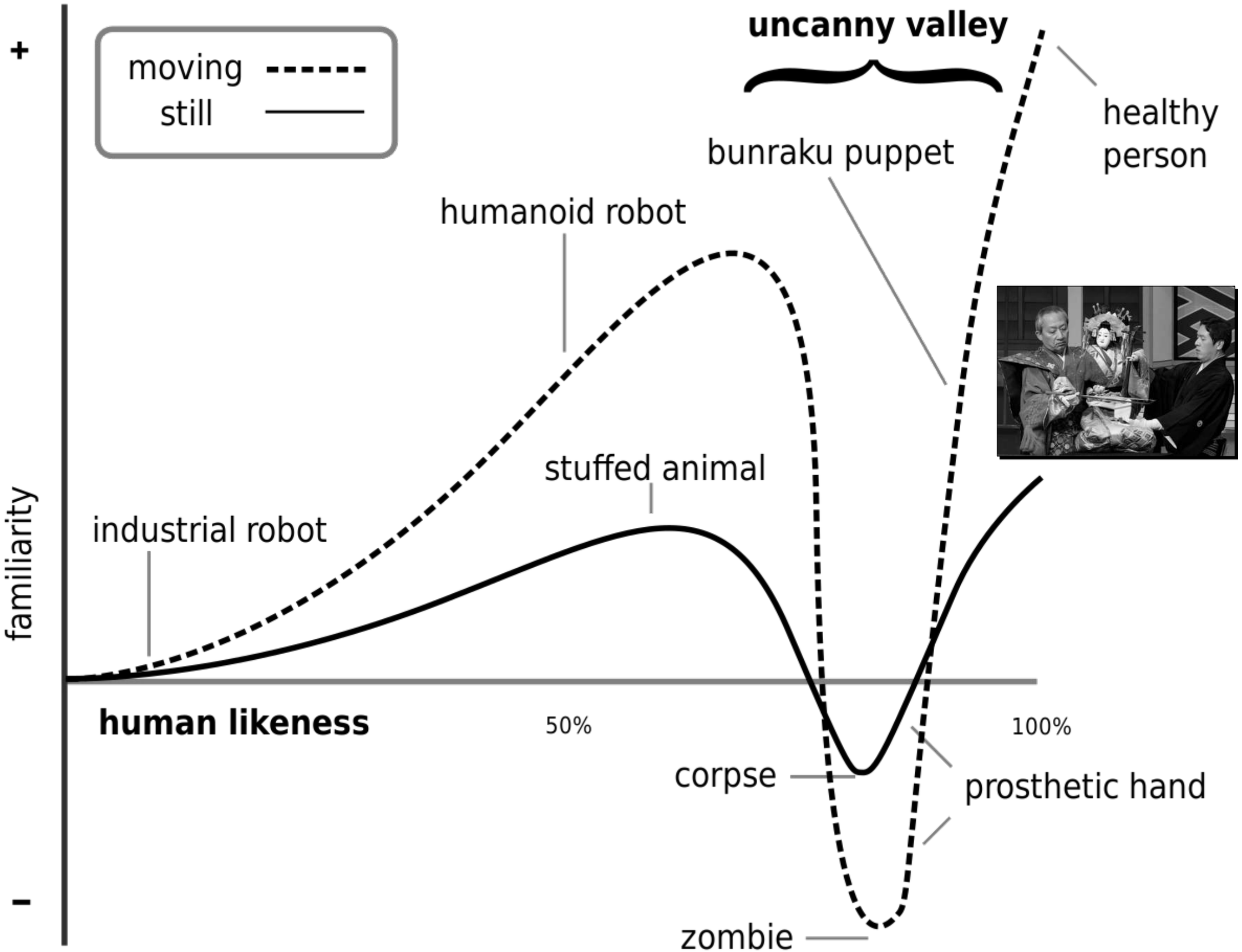
(2001, 2002, 2003)



(2008)



(2009)


+
familiarity
−
moving
still
uncanny valley
healthy person
bunraku puppet
humanoid robot
stuffed animal
industrial robot
human likeness
50%
100%
corpse
prosthetic hand
zombie

COMPUTER GRAPHICS IN GAMES

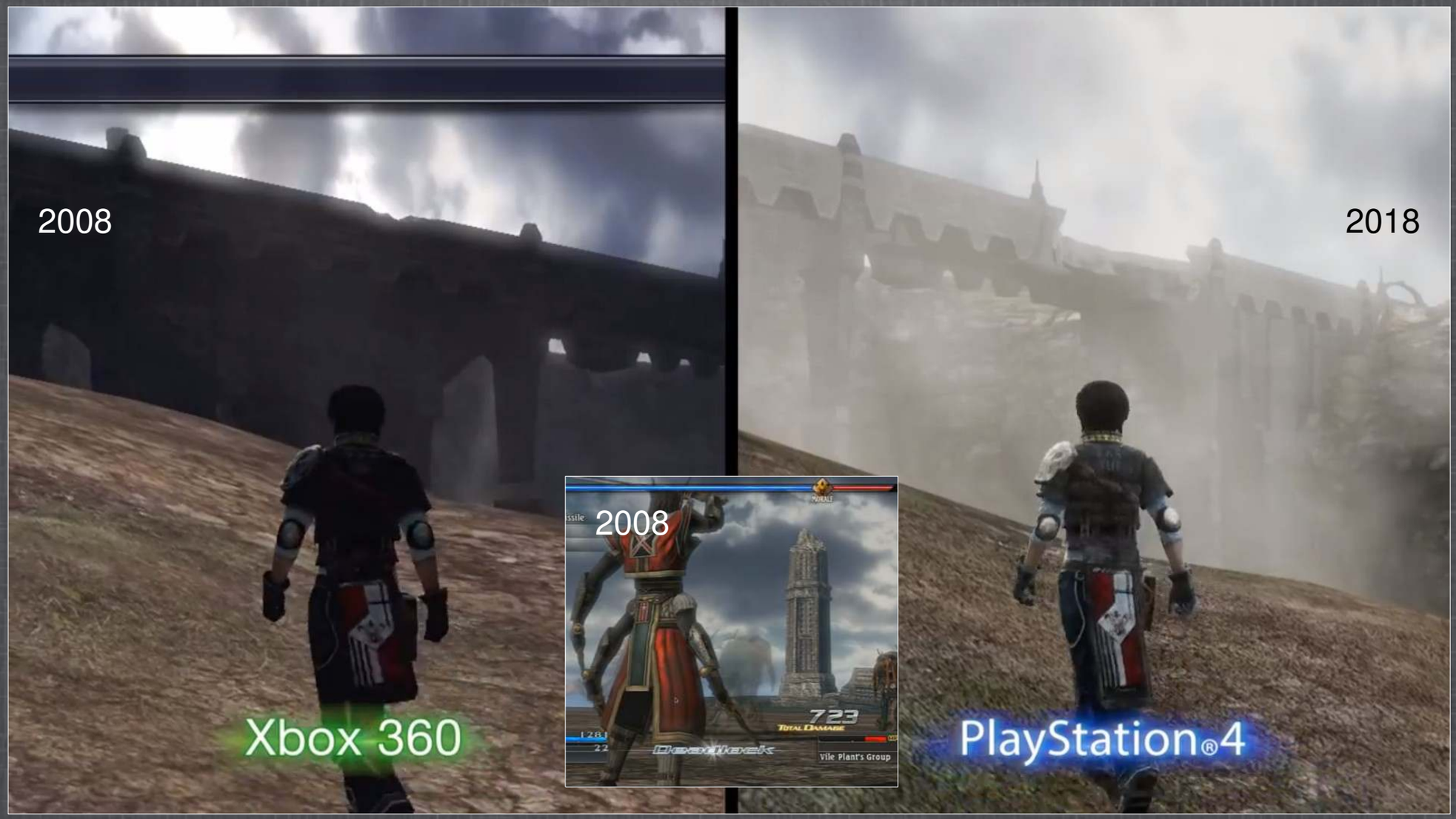
DIGITALFOUNDRY 2017
(KINGDOM HEARTS III, SQUARE ENIX, 2019)


HP
DRILL PUNCH
Attack
Magic
Items
Link
FOCUS
MP
Kingdom Hearts 3

COMPUTER GRAPHICS IN GAMES

BIOSHOCK 2 (2K BOSTON, 2010)

COMPUTER GRAPHICS IN GAMES

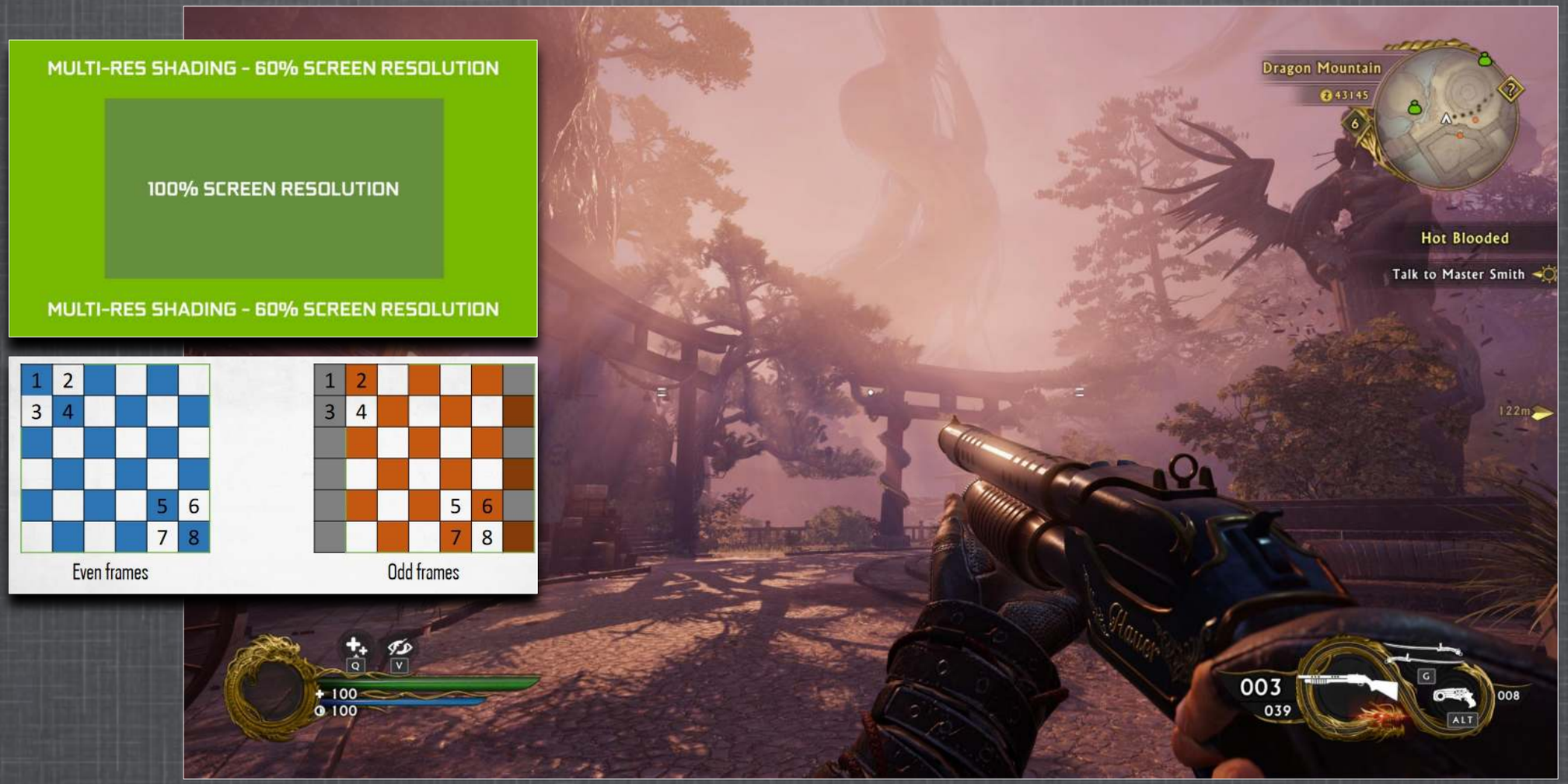


LAST REMNANT, 2018 REMASTER

COMPUTER GRAPHICS IN GAMES

DIGITAL FOUNDRY, 2019
BORDERLANDS 2 AND 3

COMPUTER GRAPHICS IN GAMES

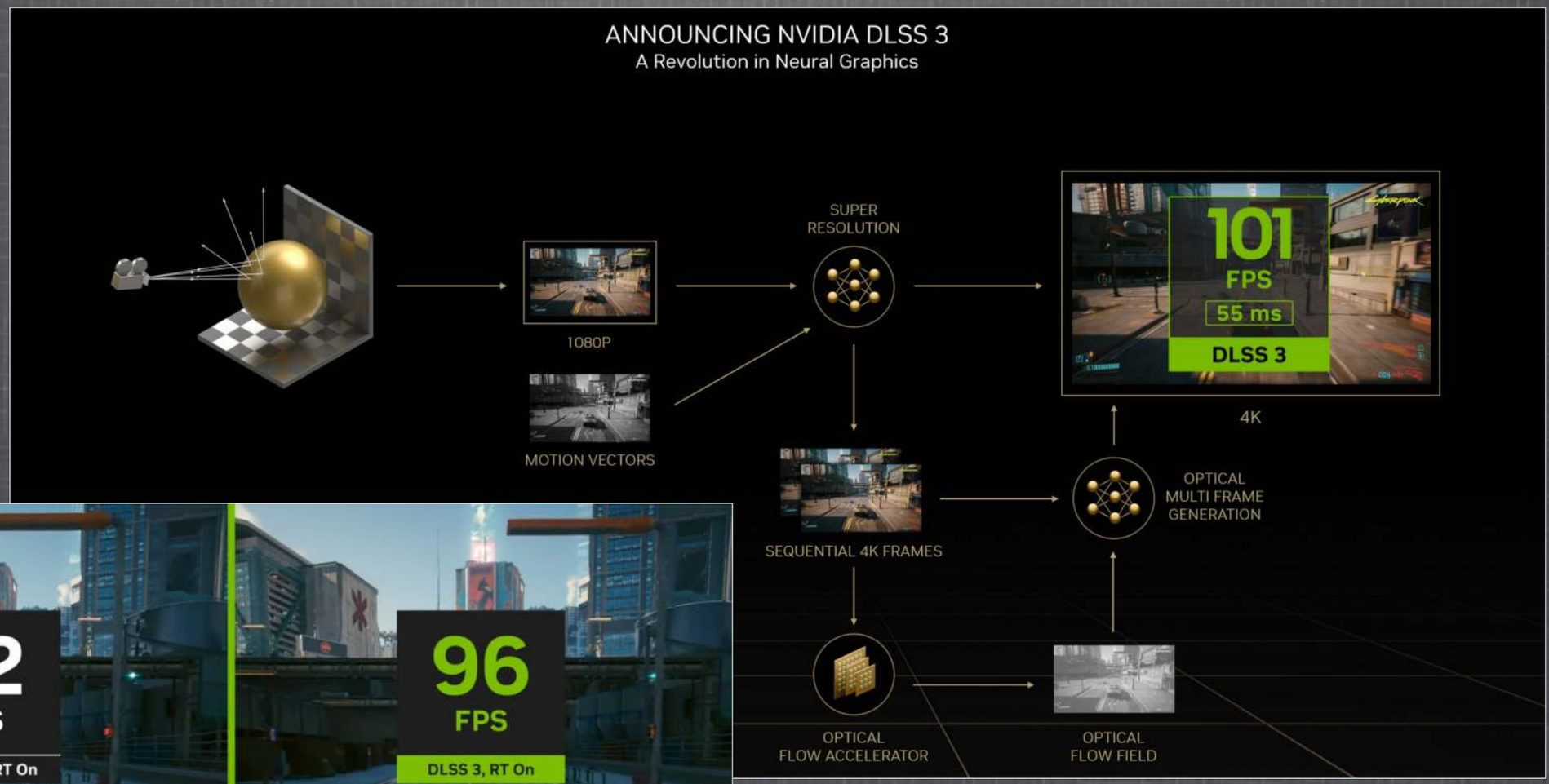


SHADOW WARRIOR 2
MULTI-RESOLUTION MULTI-FRAME SHADING

COMPUTER GRAPHICS IN GAMES

METAL GEAR SOLID 2 – 2021 UPSCALED TRAILER

COMPUTER GRAPHICS IN GAMES

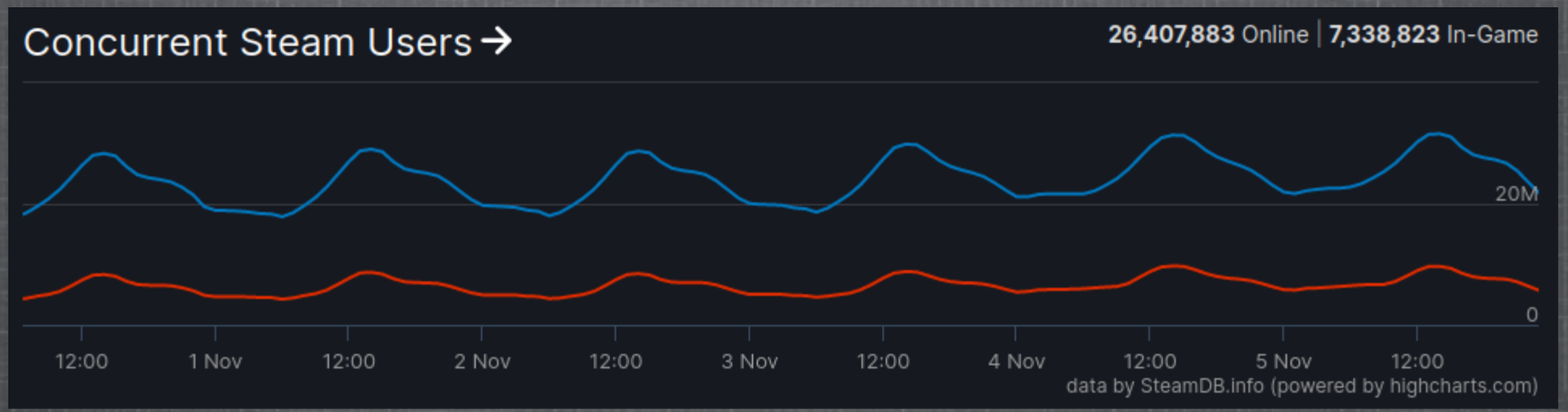


CYBERPUNK 2077 @ GTC 2022

COMPUTER GRAPHICS IN GAMES

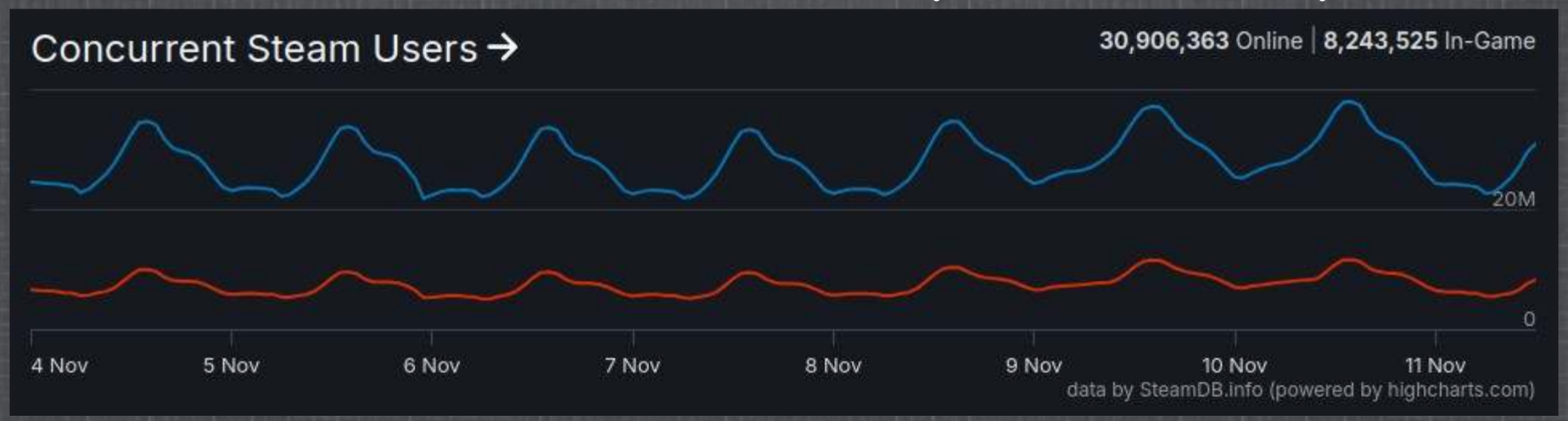
Online Gaming

6 nov 2023 (Steam, 120M+ monthly active users, 62.6M daily active users)



revenue of US $8.56 billion for 2024

11 nov 2024 (Steam, 132M+ monthly active users, 69M+ daily active users)



projected revenue of US $8.58 billion for 2024

COMPUTER GRAPHICS IN GAMES

		Game title	Change	Publisher
1.		Counter-Strike 2 & GO	-	Valve
2.		Minecraft	-	Mojang Studios
3.		ROBLOX	-	Roblox
4.		Fortnite	↑ 1	Epic Games
5.		League of Legends	↑ 2	Riot Games
6.		Dota 2	↑ 3	Valve
7.		The Sims 4	↑ 1	Electronic Arts
8.		Valorant	↑ 2	Riot Games
9.		PEAK (Aggro Crab)	↓ 3	Aggro Crab Games
10.		HELLDIVERS 2	↑ 4	Sony Interactive Entertainment

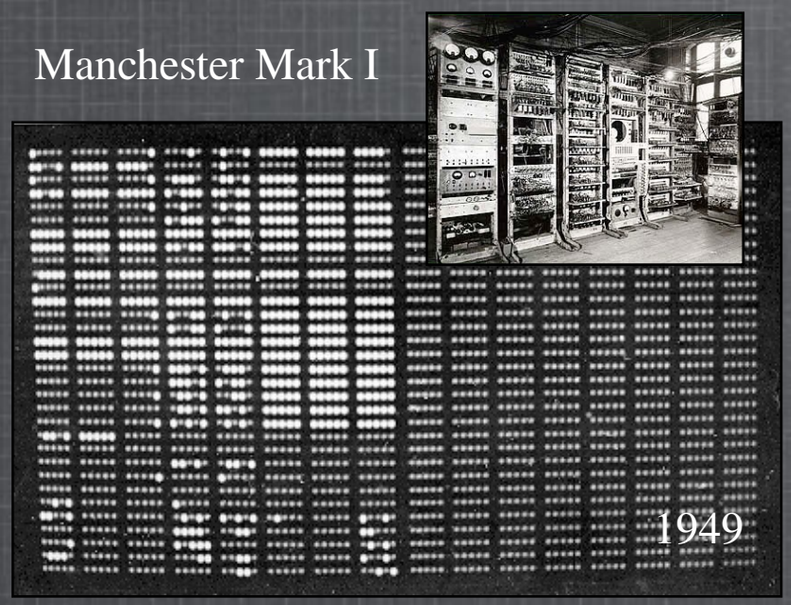
Most popular PC games globally by Monthly Active Users
(Newzoo.com, September 2025)

DIGITAL MEDIA

DIGITAL MEDIA

From *text* to *hypertext*, *image* and *video*

Manchester Mark I



Fiction (1975-77)

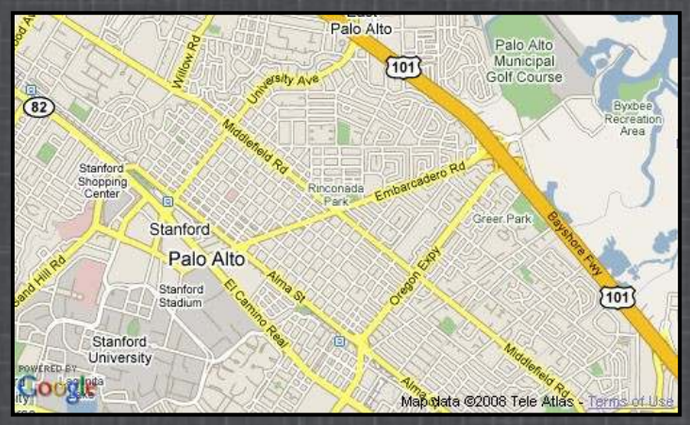
Reality (1999)

flickr · facebook · YouTube · twitch

DIGITAL MEDIA

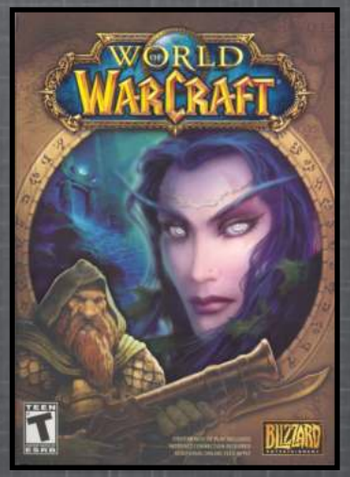
Digital Twins

Google Earth and Google Maps



Home Simulation

Social Spaces



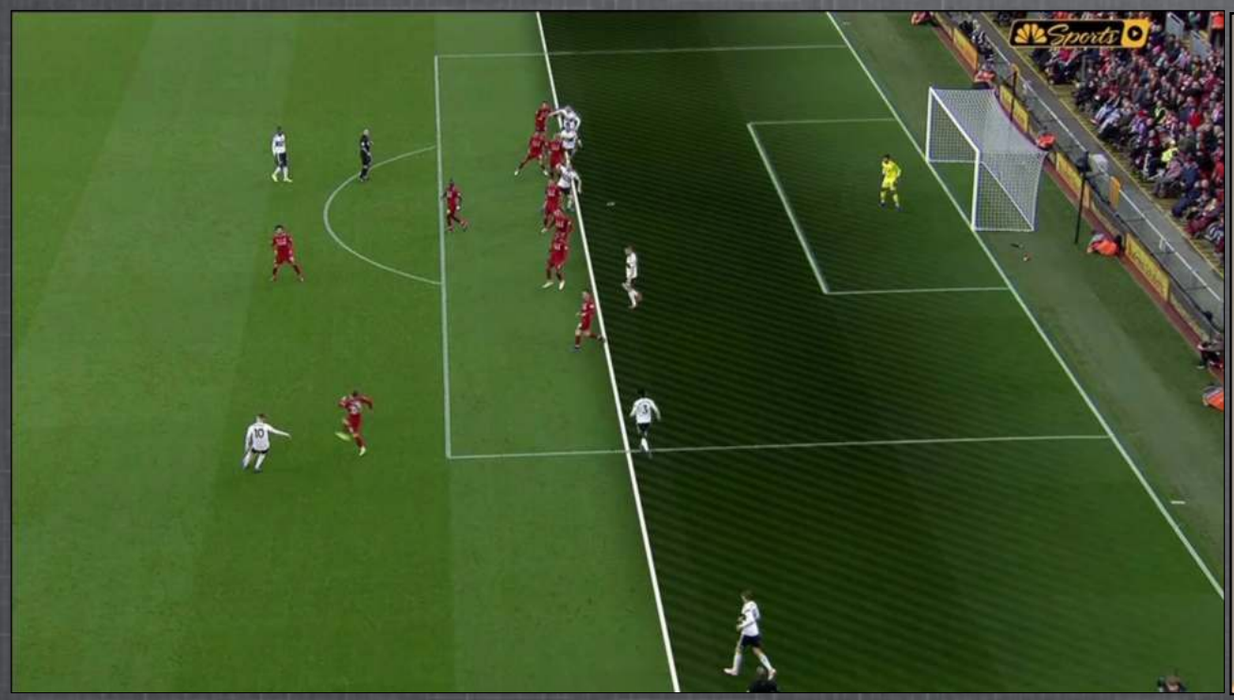
Second Life

Electronic Publishing

DIGITAL MEDIA

Processing Feelings

Information overlay

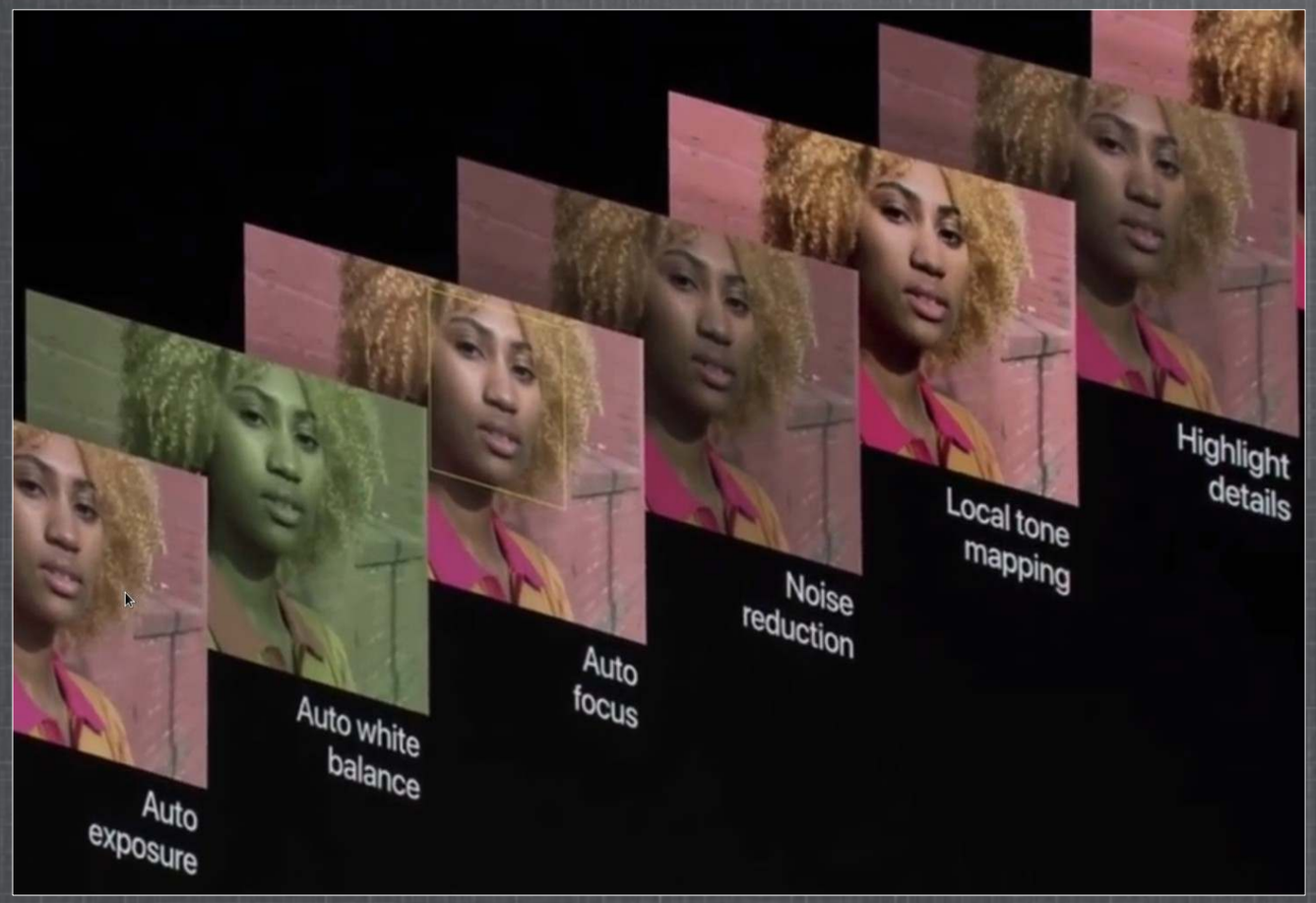


Which one is the original? Are they real?

Composition Enhancement

DIGITAL MEDIA

How real is our memory of a capture?



Apple X s MAX launch (sep 2018)

DIGITAL MEDIA

How real is our memory of a capture?

Lens Flare

DIGITAL MEDIA

Future of Media

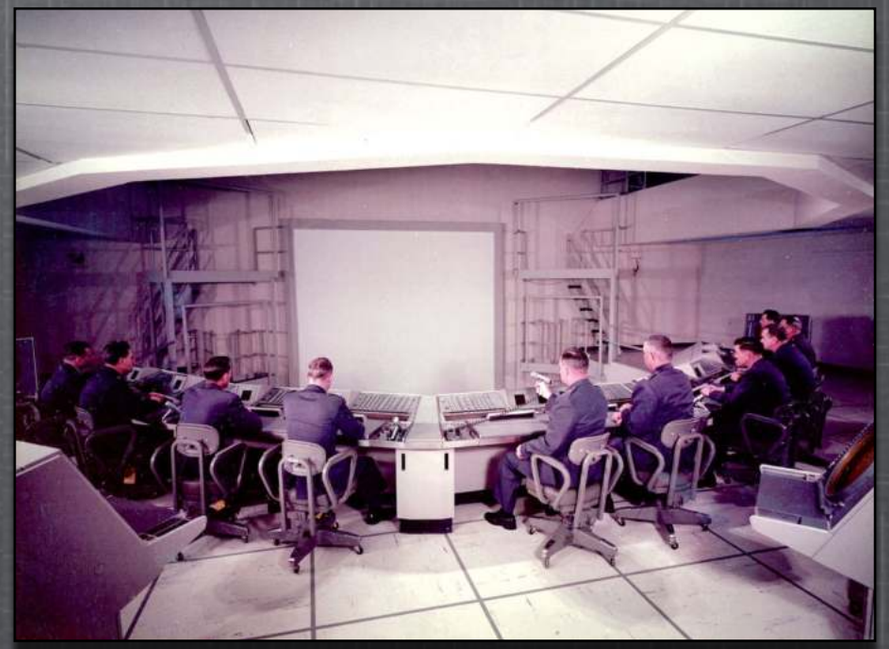
COMPUTER AIDED DESIGN AND VISUALISATION

Unreal Engine 4 - Unreal Paris (January 2015)

BRIEF HISTORY OF COMPUTER GRAPHICS

COMPUTER GRAPHICS

TERM COINED BY WILLIAM FETTER (BOEING) IN 1960

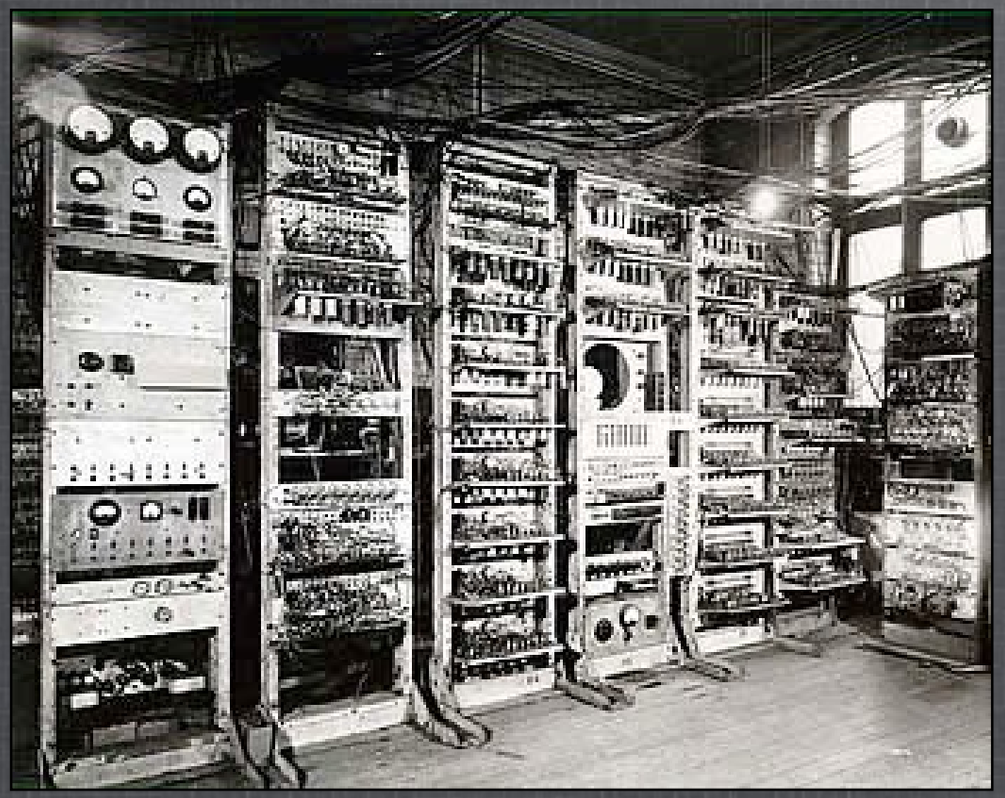


1ST GRAPHICS SYSTEM

USAF SAGE RADAR DATA (MIT) IN MID 1950'S

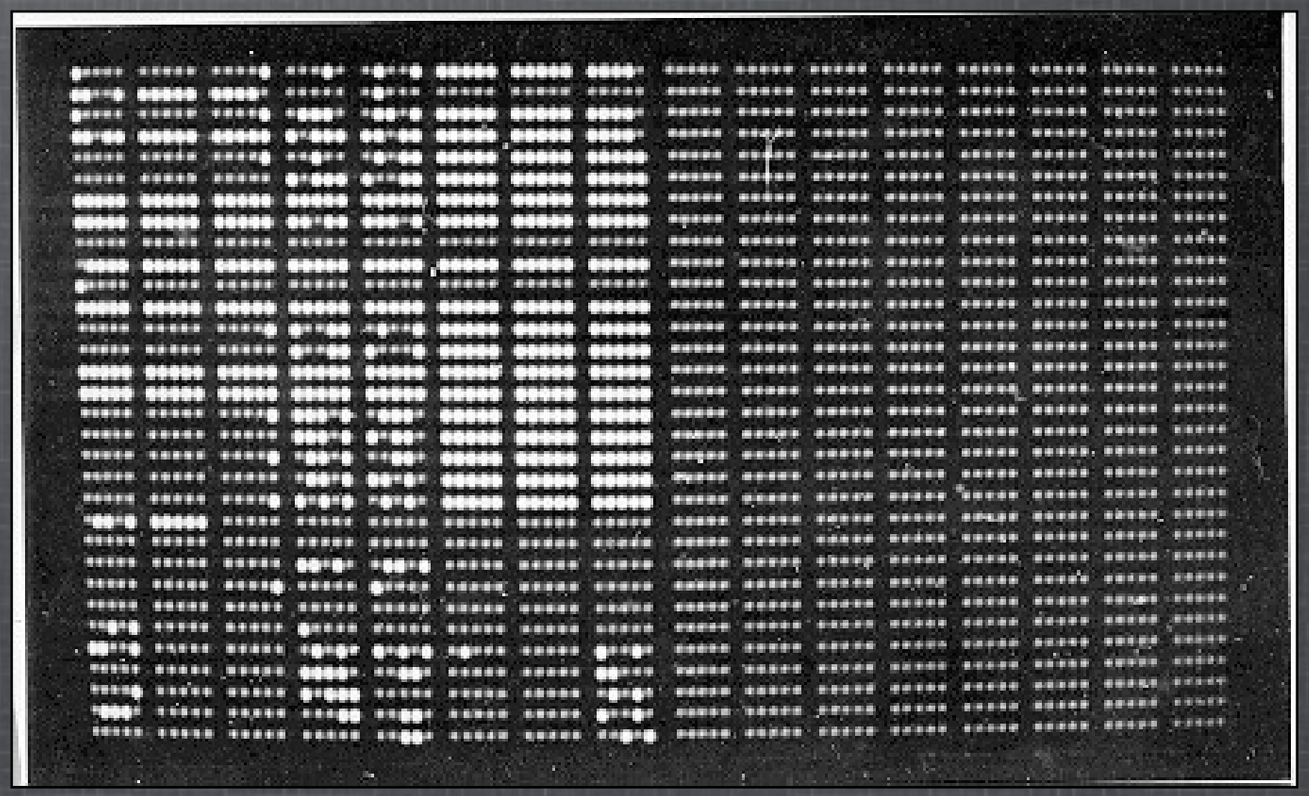
2D COMPUTER GRAPHICS

1ST TEXT SYSTEMS

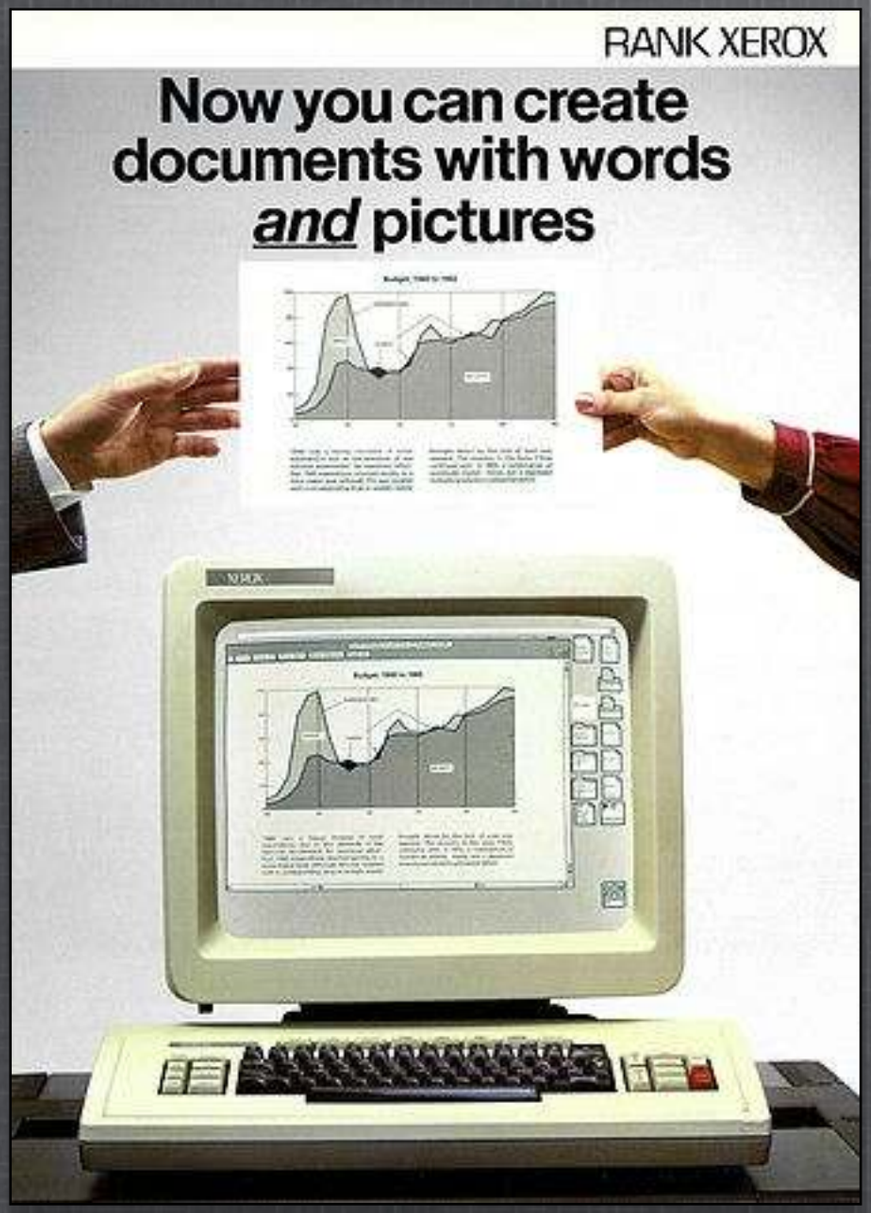


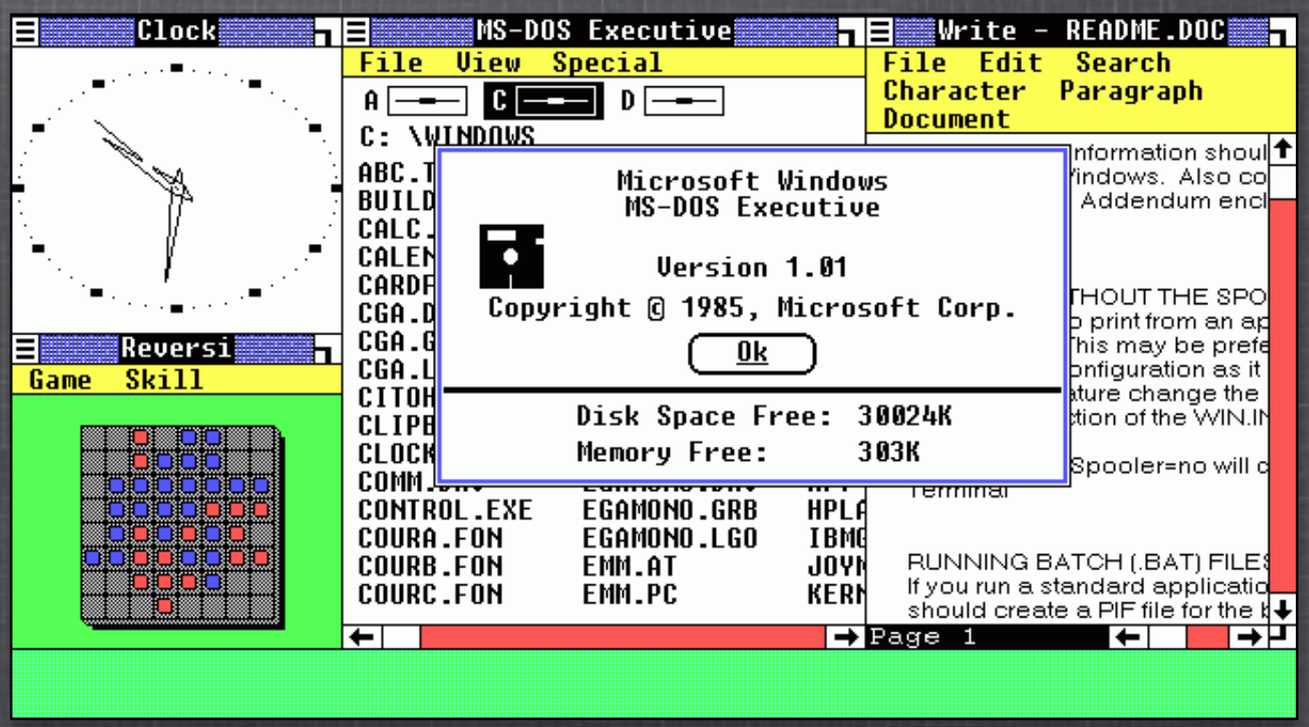
Manchester Mark I

Display

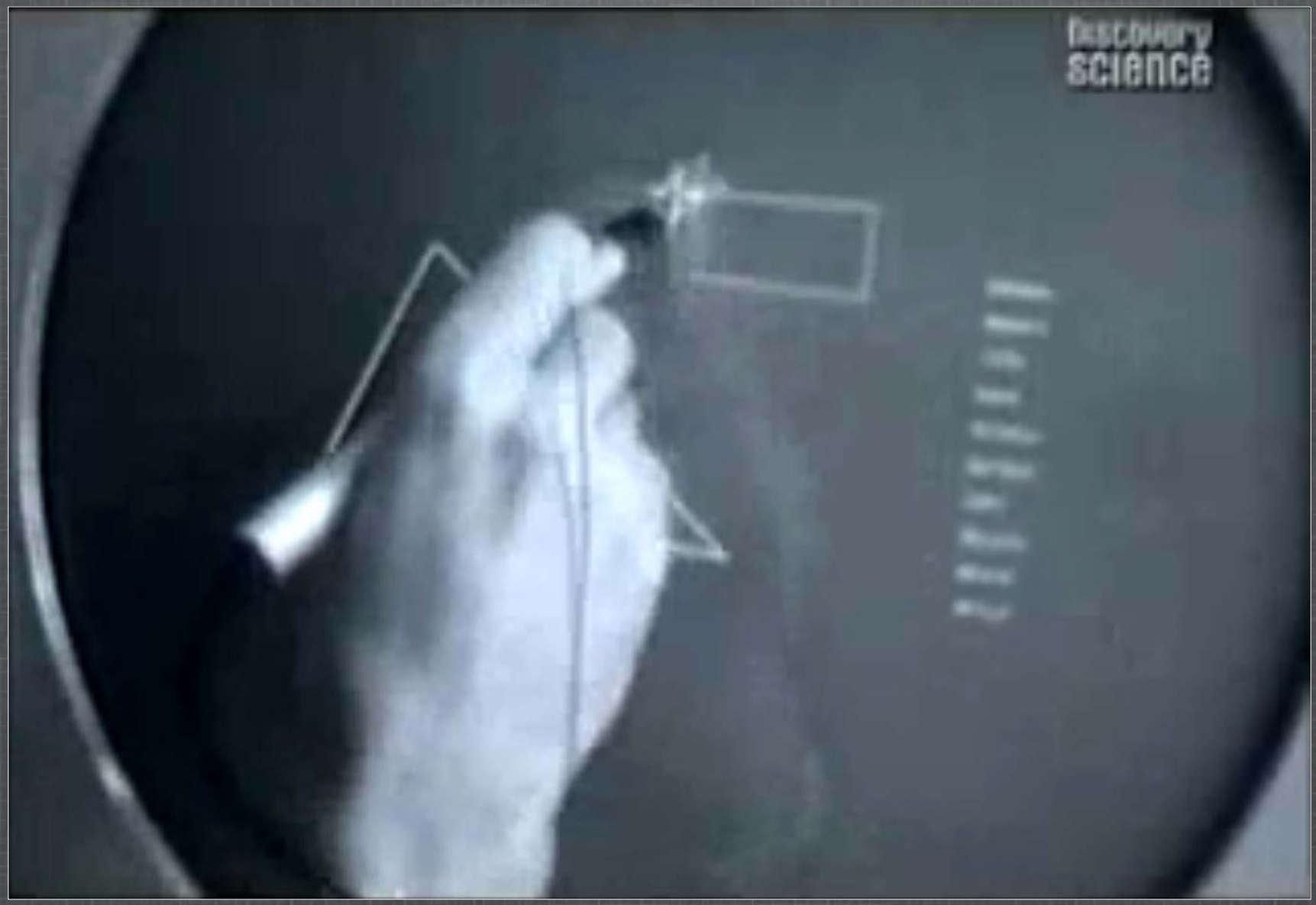


1ST GRAPHICAL USER INTERFACES



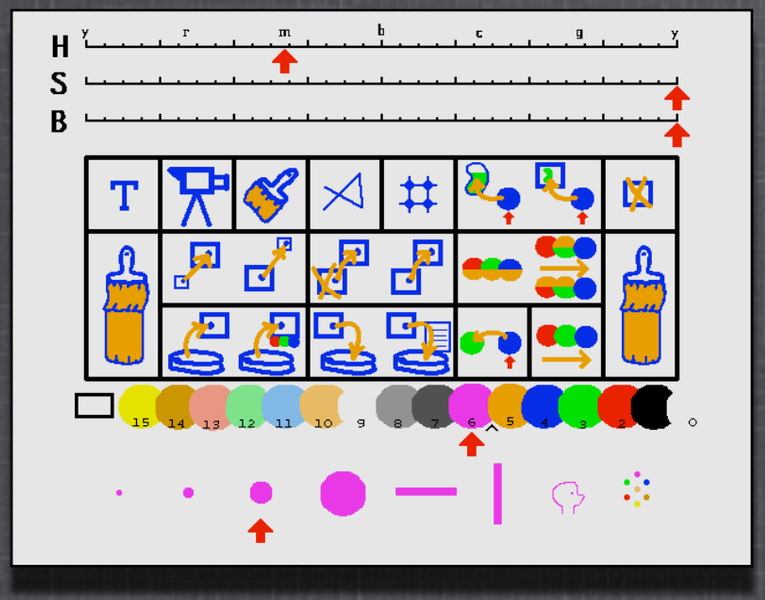


1ST DRAWING SYSTEMS

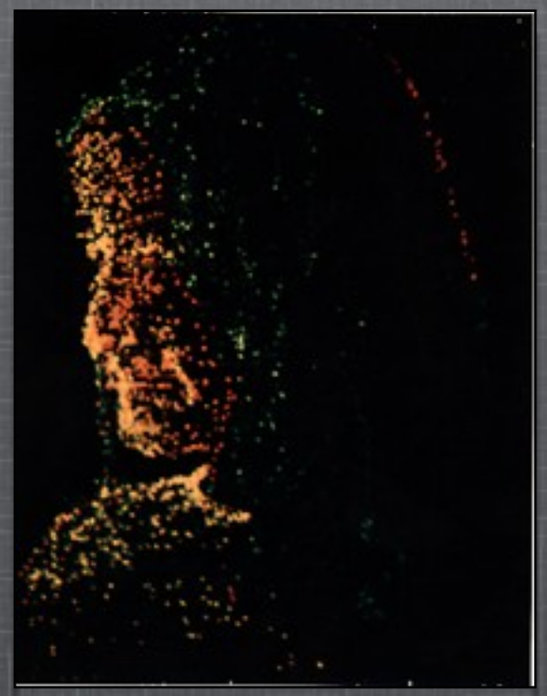


Sketchpad (Sutherland, MIT 1963)

1ST PAINT SYSTEMS

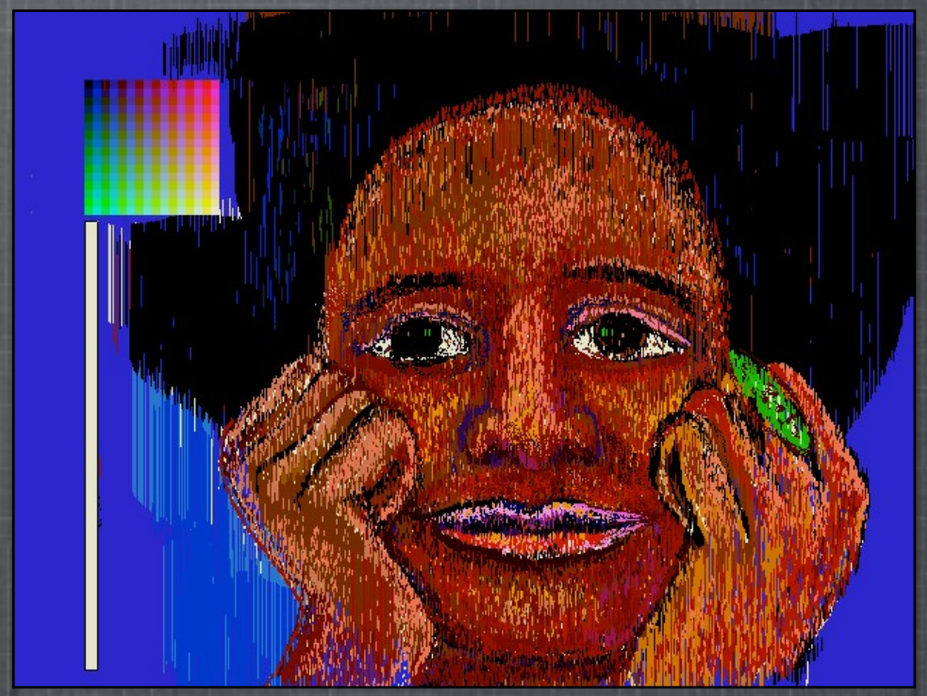


SuperPaint Framebuffer System (Shoup and Smith, 1973-1979)



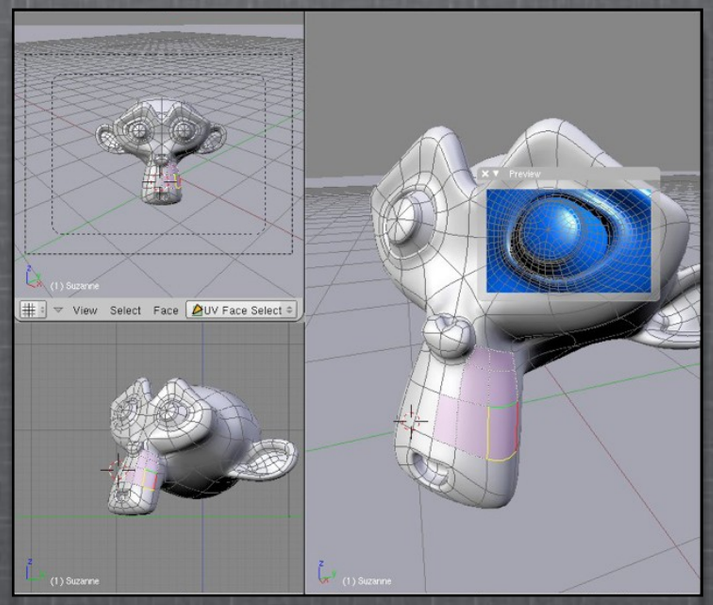
Janet, by Bob Flegal

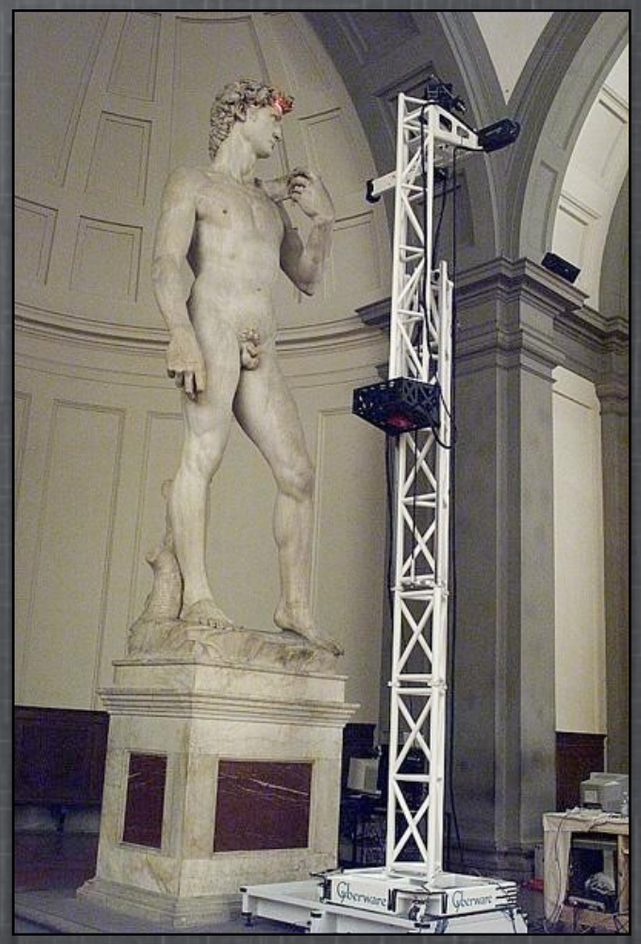
Black Girl, by Fritz Fisher

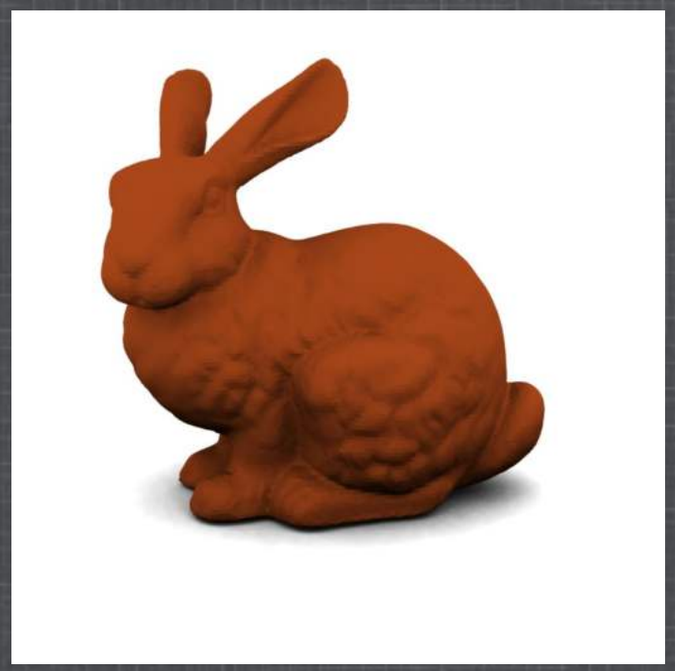


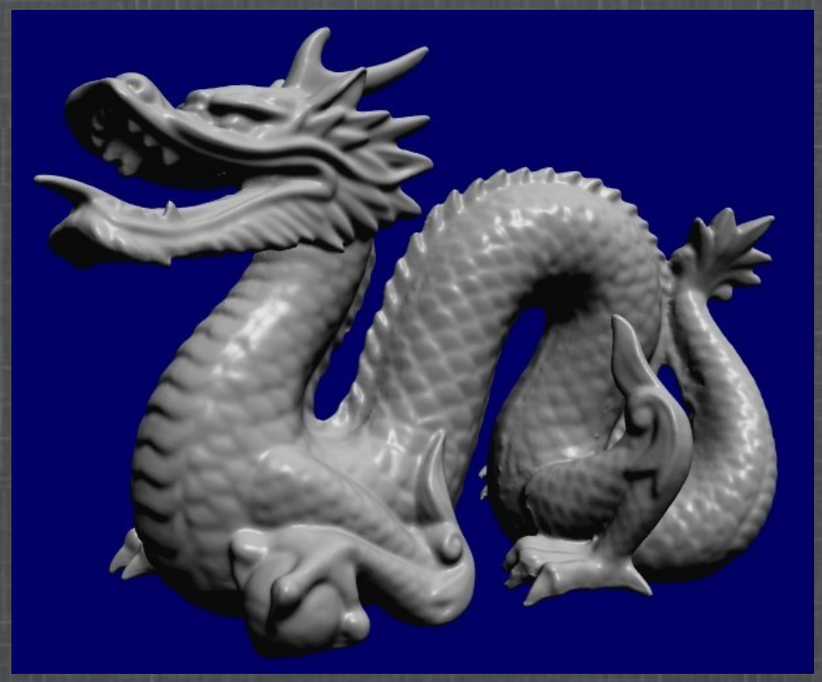
3D COMPUTER GRAPHICS

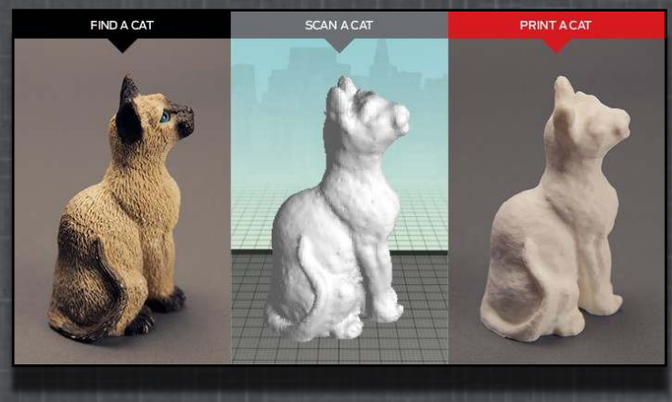
MODELING


(1) Suzanne
View
Select
Face
UV Face Select
Preview
(1) Suzanne
(1) Suzanne

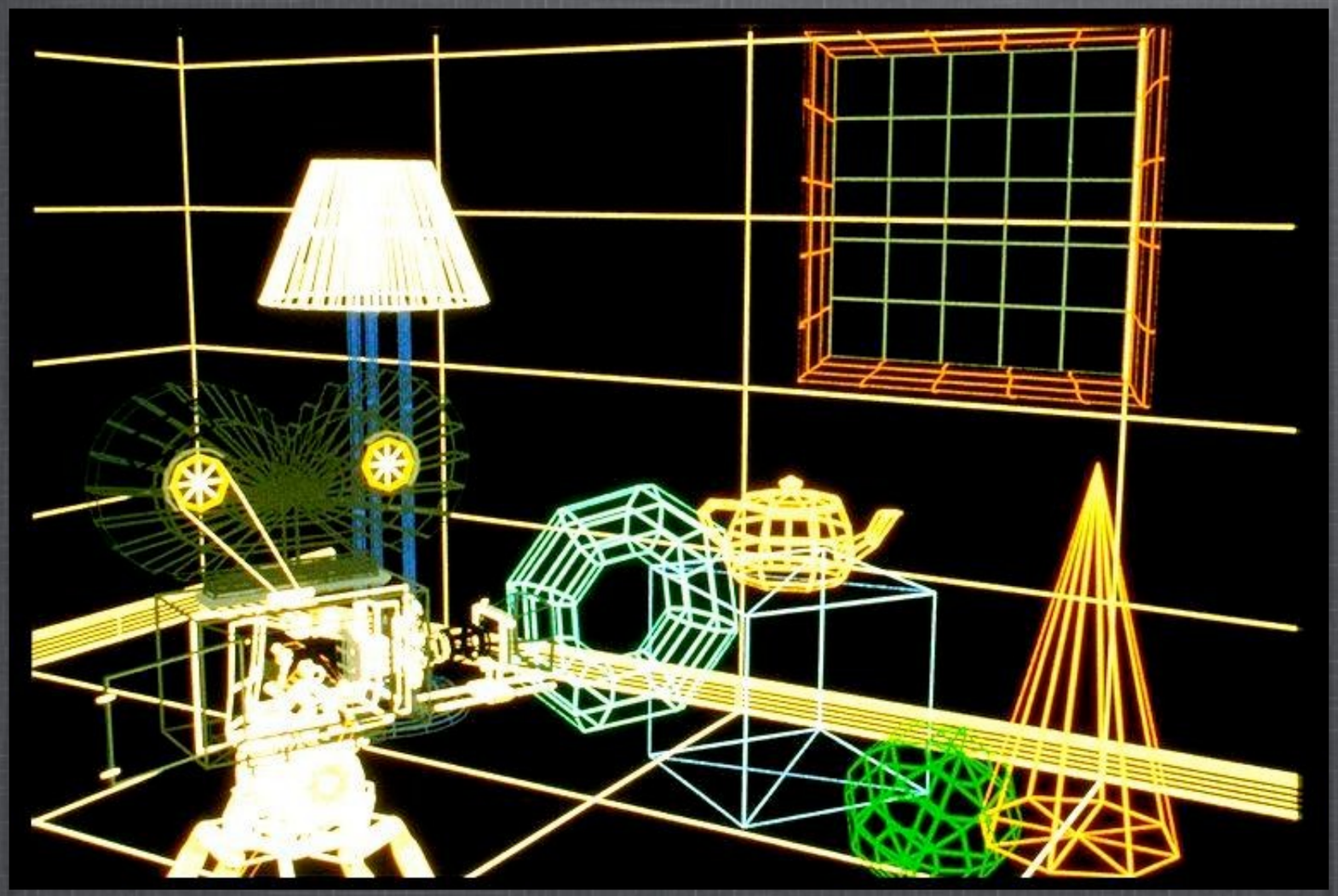






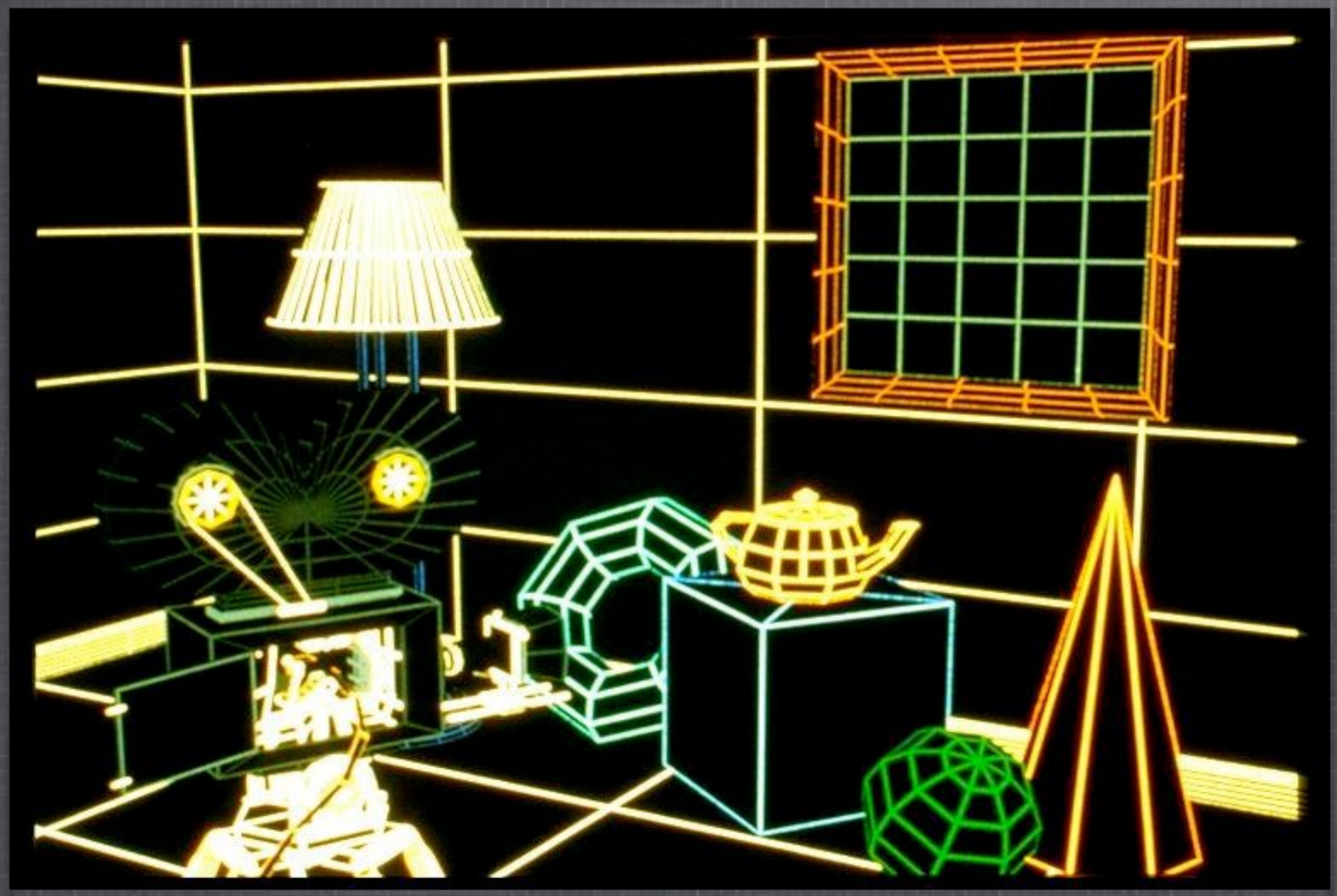

FIND A CAT
SCAN A CAT
PRINT A CAT

RENDERING



RENDERING

Visibility (1960's)



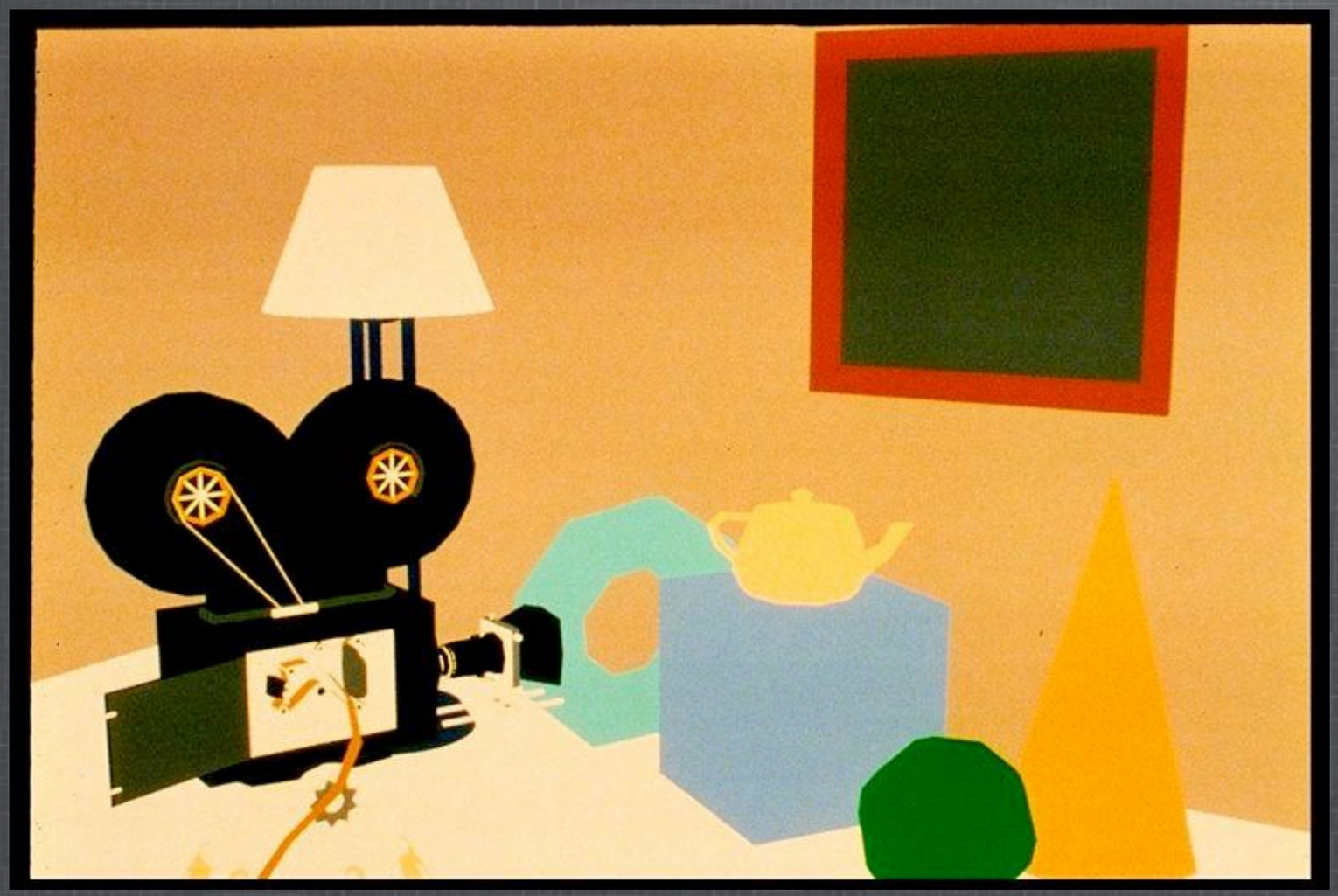
Hidden Line Algorithms:
Roberts (63), Appel (67)

Hidden Surface Algorithms:
Warnock (69), Watkins (70)

Visibility and Sorting:
Sutherland (74)

RENDERING

Visibility and Lighting (1970's)



Hidden Line Algorithms:
Roberts (63), Appel (67)

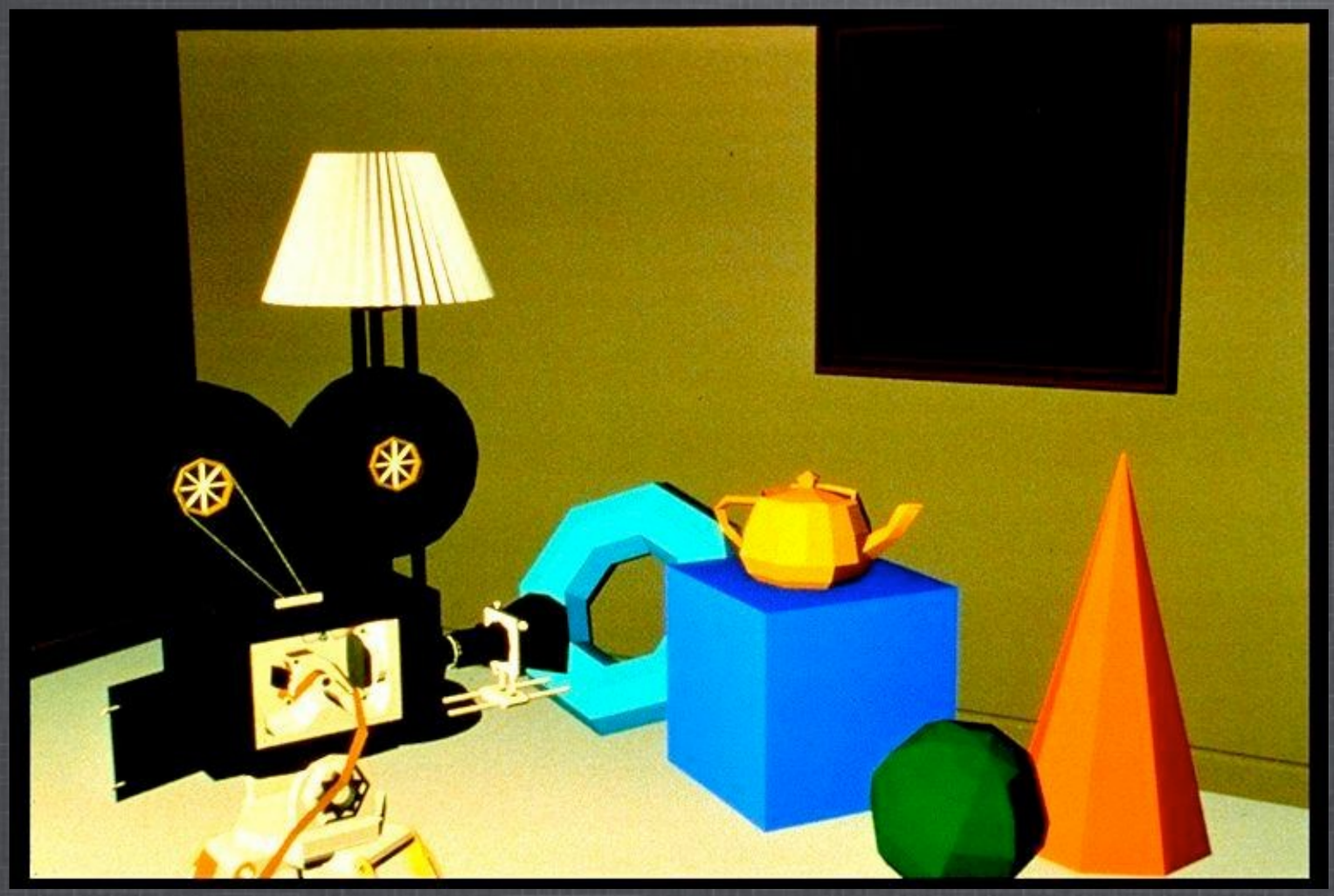
Hidden Surface Algorithms:
Warnock (69), Watkins (70)

Visibility and Sorting:
Sutherland (74)

Constant shading

RENDERING

Lighting (1970's)



Flat shading

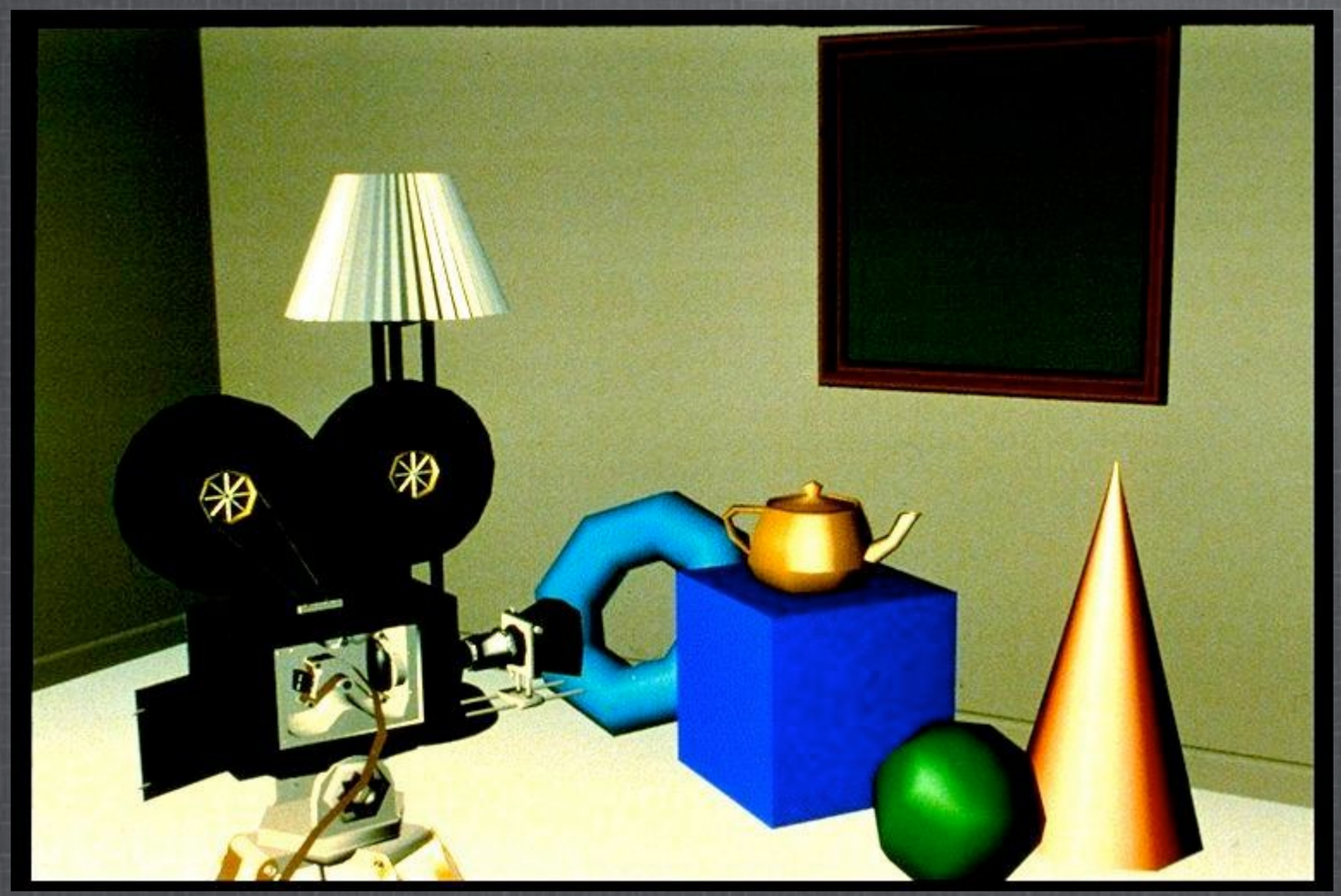
RENDERING

Lighting (1970's)

Diffuse lighting
(Gouraud 1971)

RENDERING

Lighting (1970's)



Diffuse lighting
(Gouraud 1971)

Specular lighting
(Phong 1974)

RENDERING

Lighting (1970's)

Diffuse lighting
(Gouraud 1971)

Specular lighting
(Phong 1974)

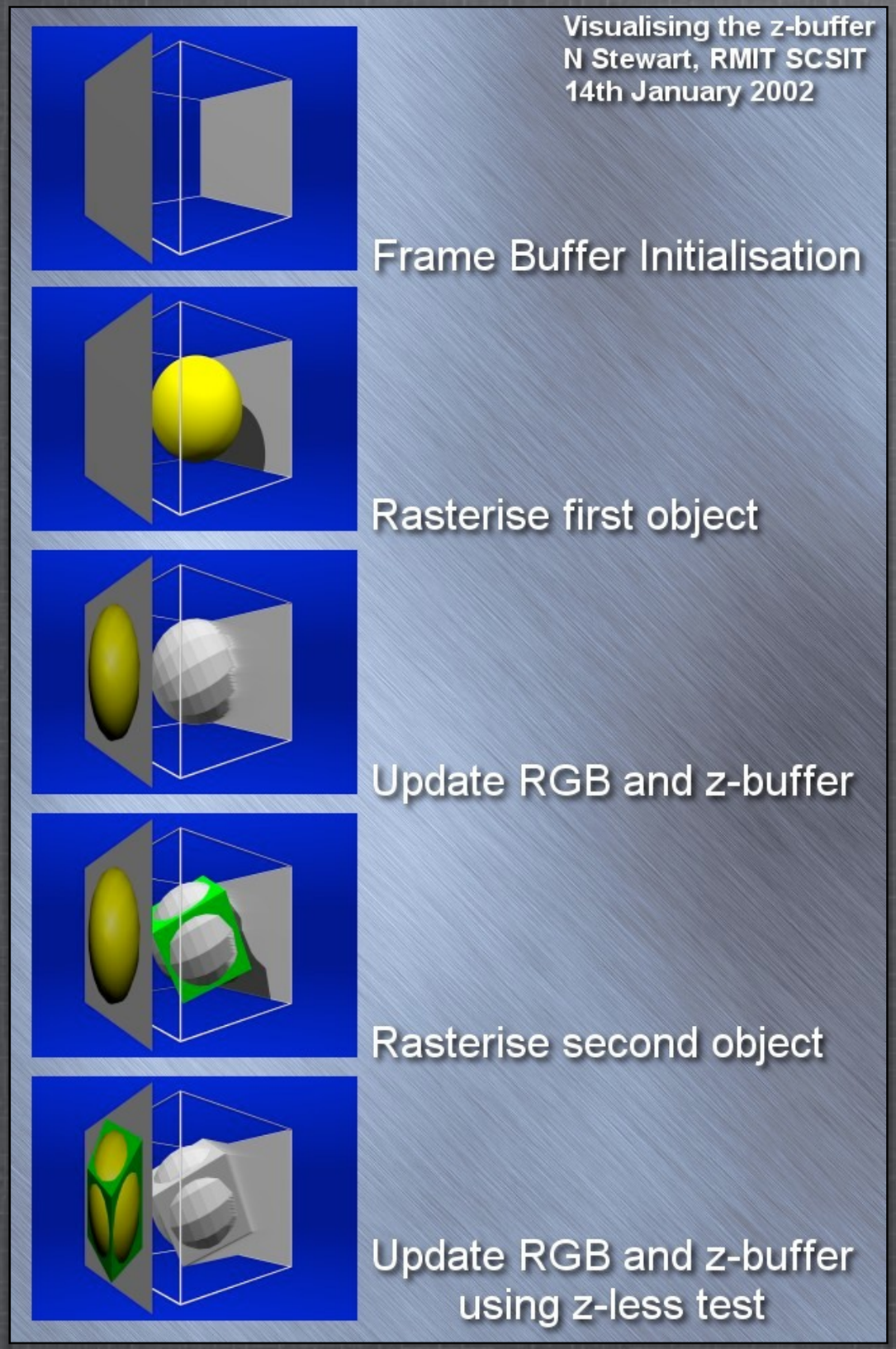
RENDERING

Lighting (1970's)

Diffuse lighting
(Gouraud 1971)

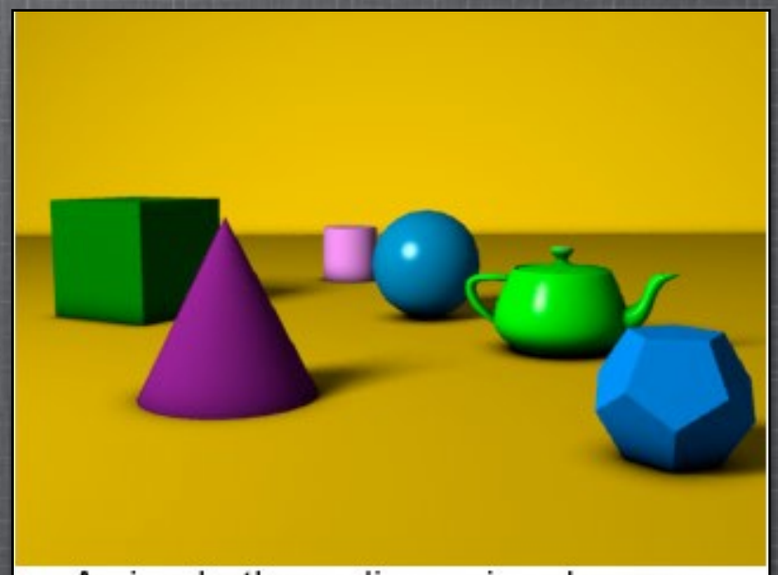
Specular lighting
(Phong 1974)

Curved Surfaces, Texture
(Blinn, 1974)



RENDERING

Z-buffer Hidden Surface
(Catmull, 1974)



A simple three dimensional scene

Z-buffer representation

RENDERING

Lighting (1970's)

Diffuse lighting
(Gouraud 1971)

Specular lighting
(Phong 1974)

Curved Surfaces, Texture
(Blinn, 1974)

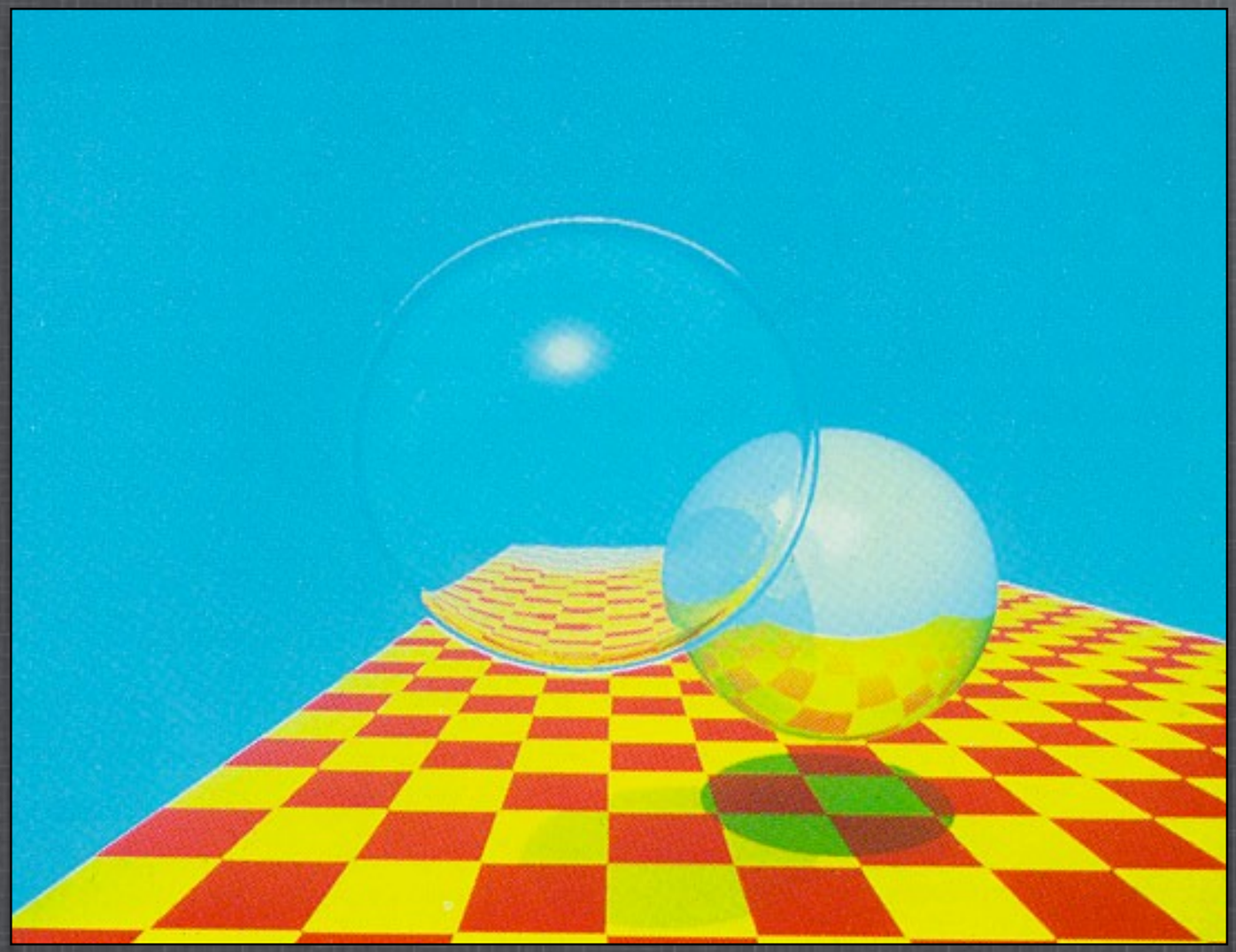
Antialias and Shadows
Crow (1977)

GLOBAL ILLUMINATION

(1980's and 1990's)

RENDERING

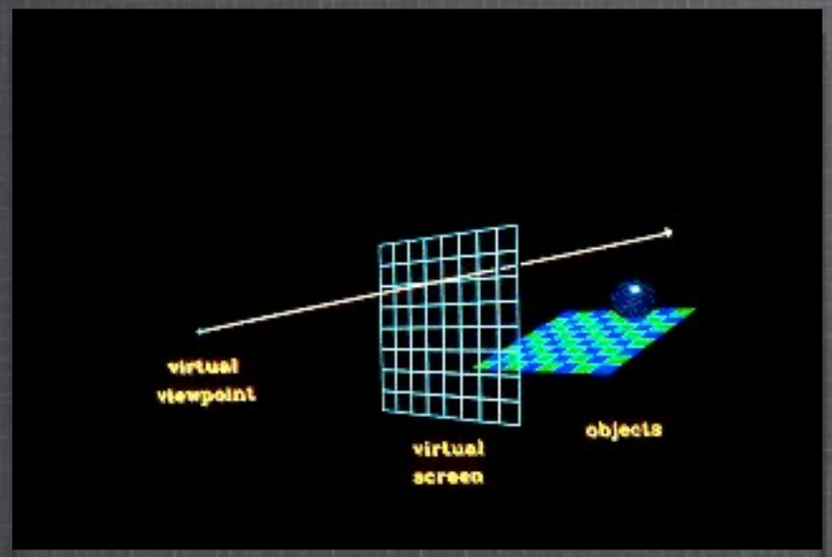
Global Illumination (1980's and 1990's)

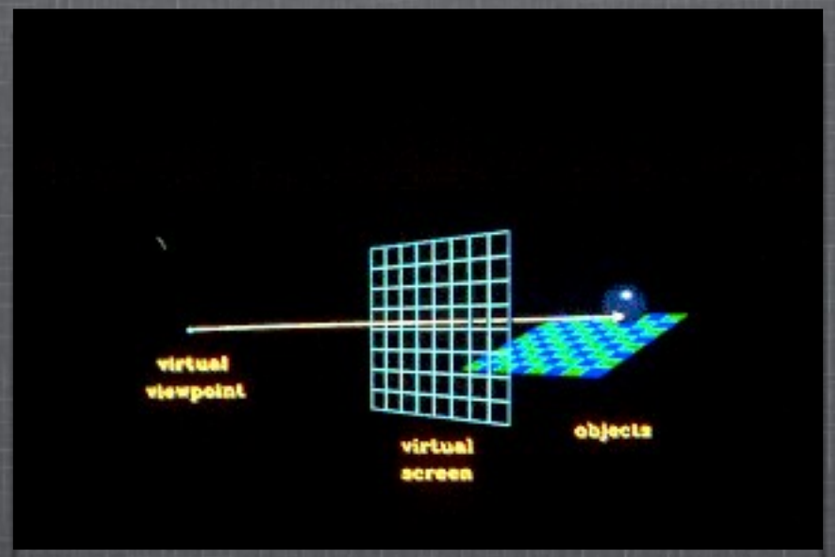


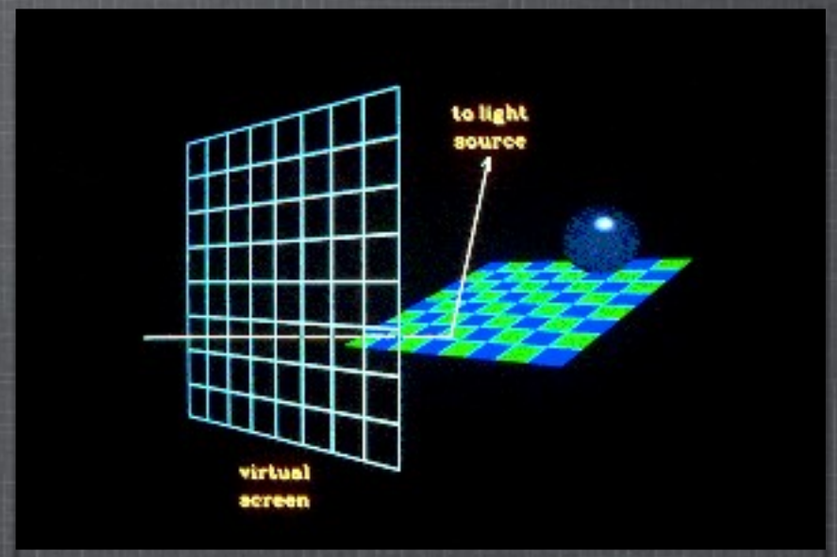
Raytracing (recursive)
(Whitted 1980)

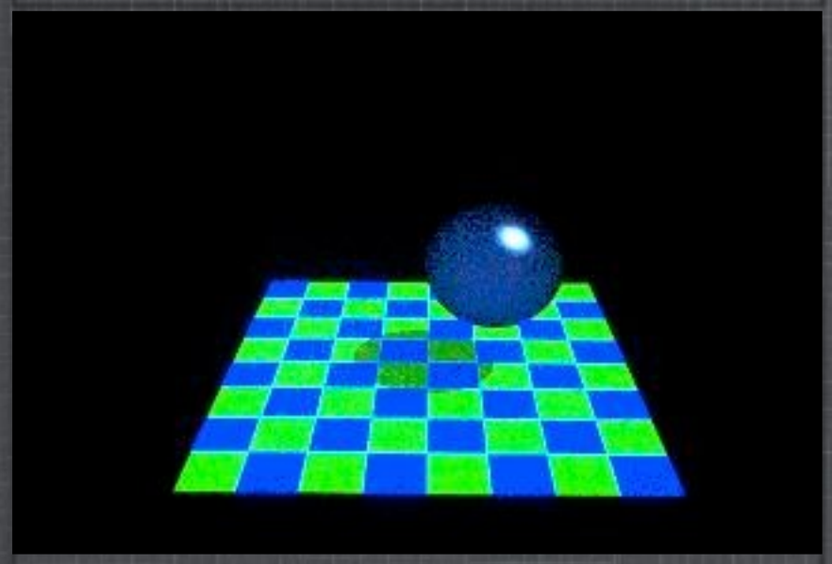
RENDERING

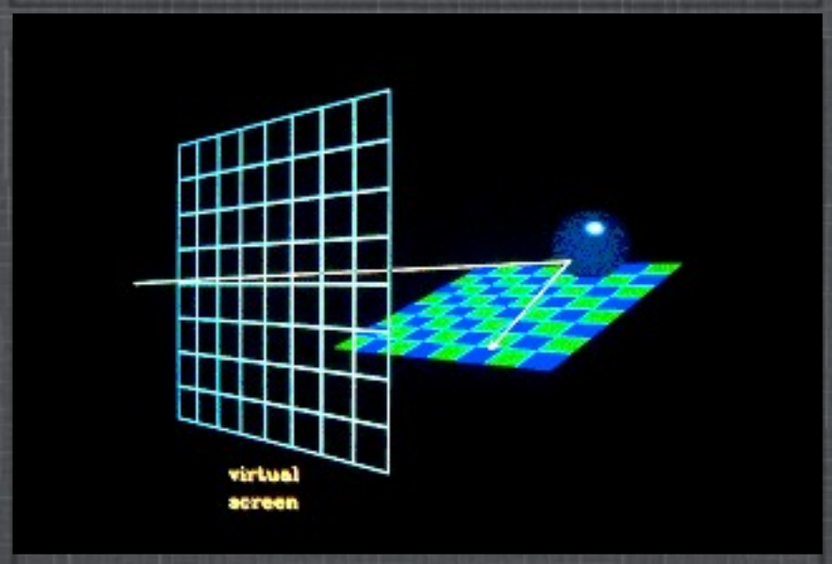
Raytracing (Whitted 1980)

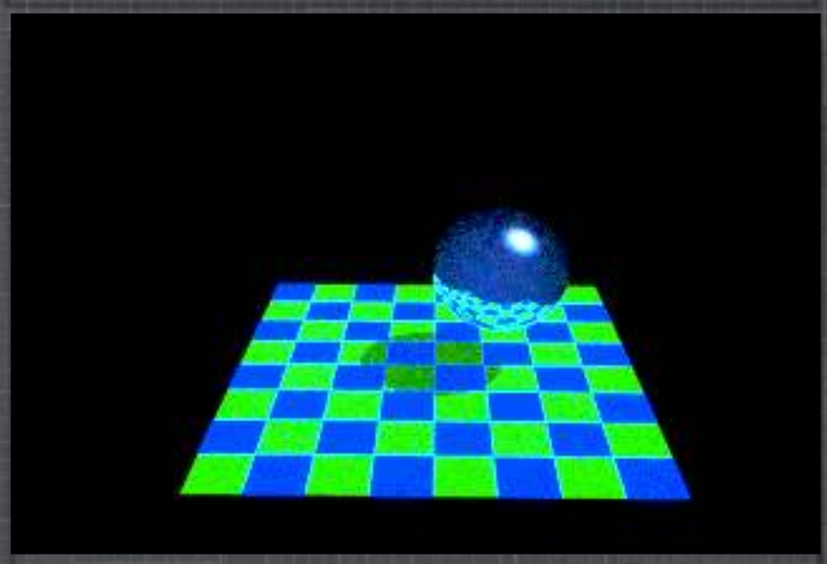






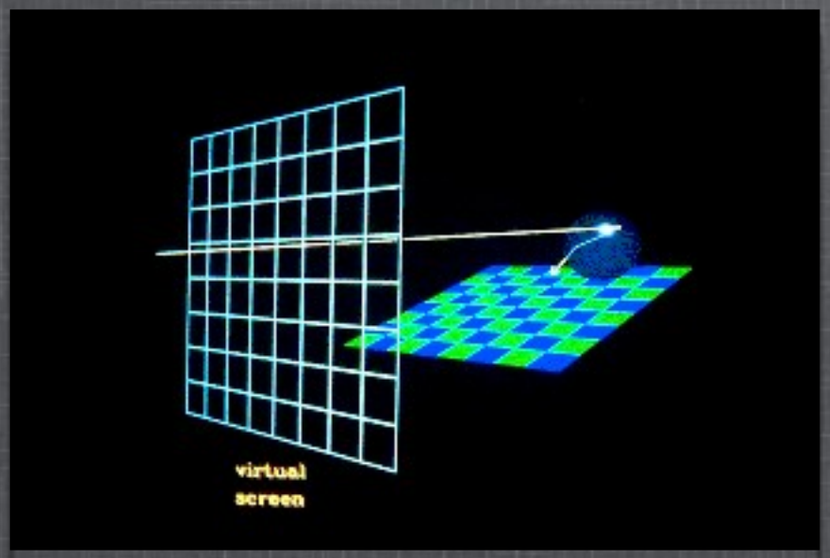


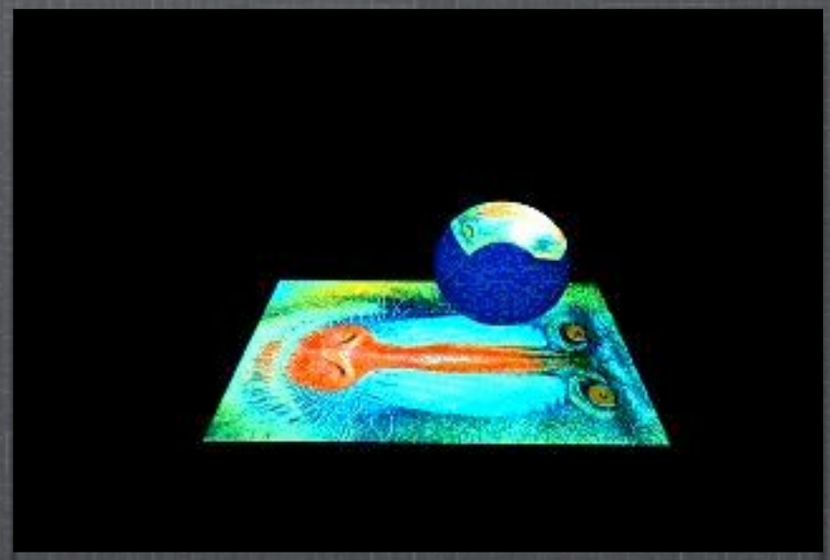




RENDERING

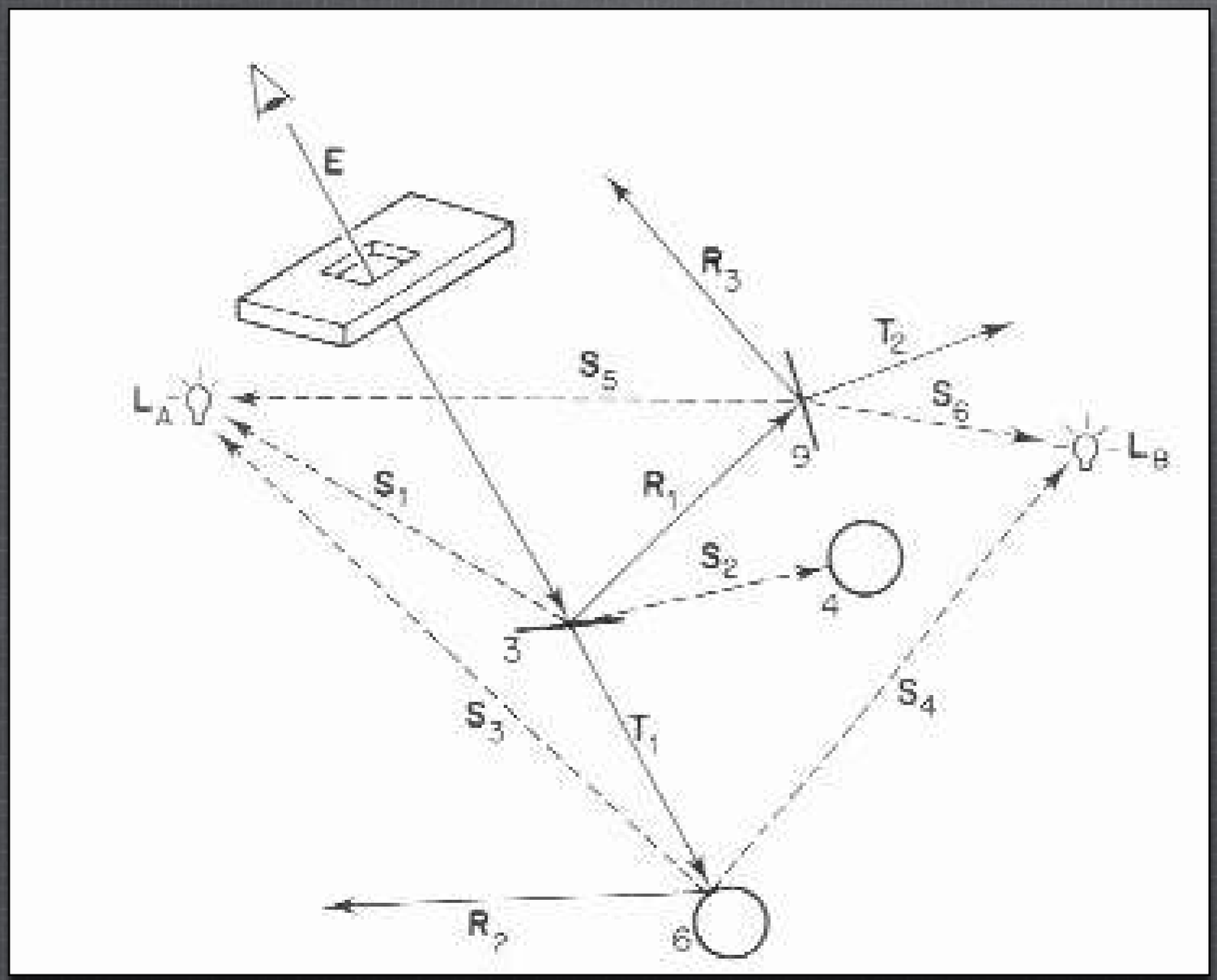
Raytracing (Whitted 1980)

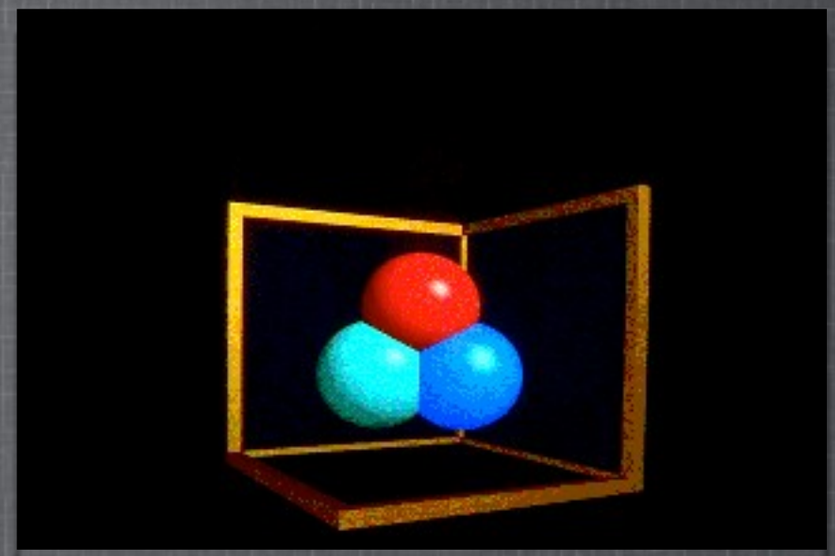


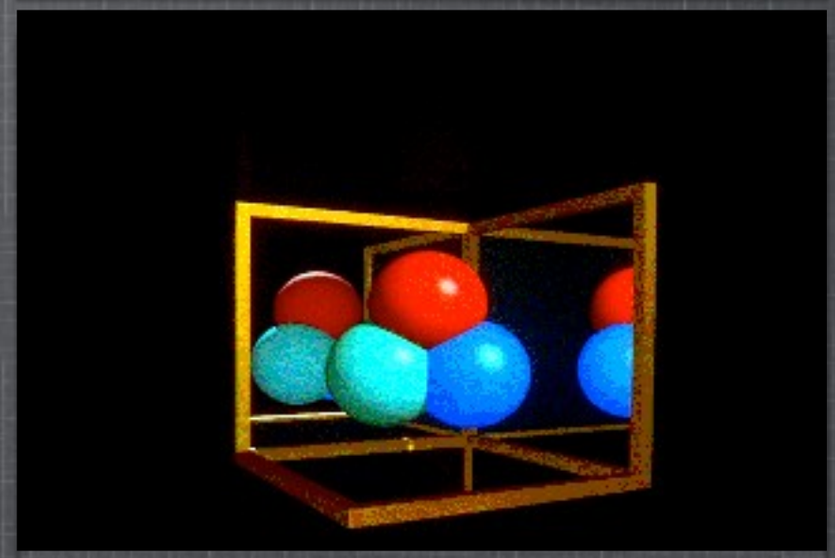


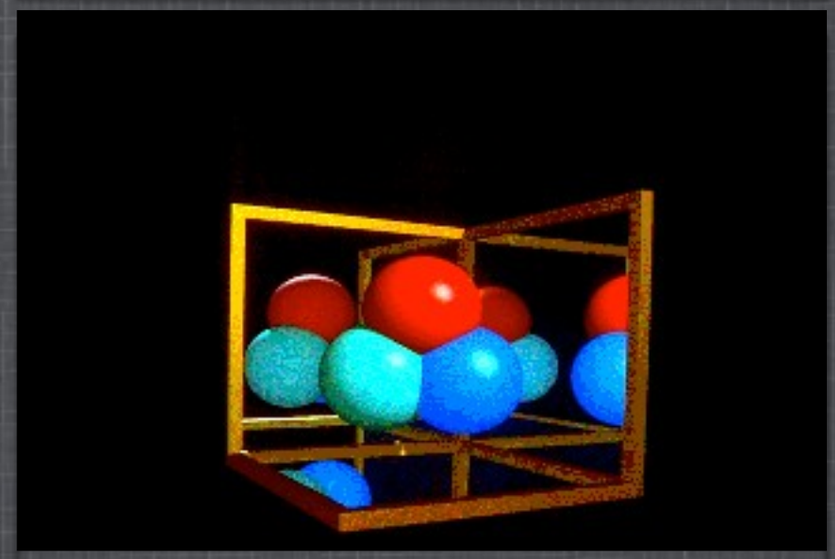
RENDERING

Raytracing (Whitted 1980)

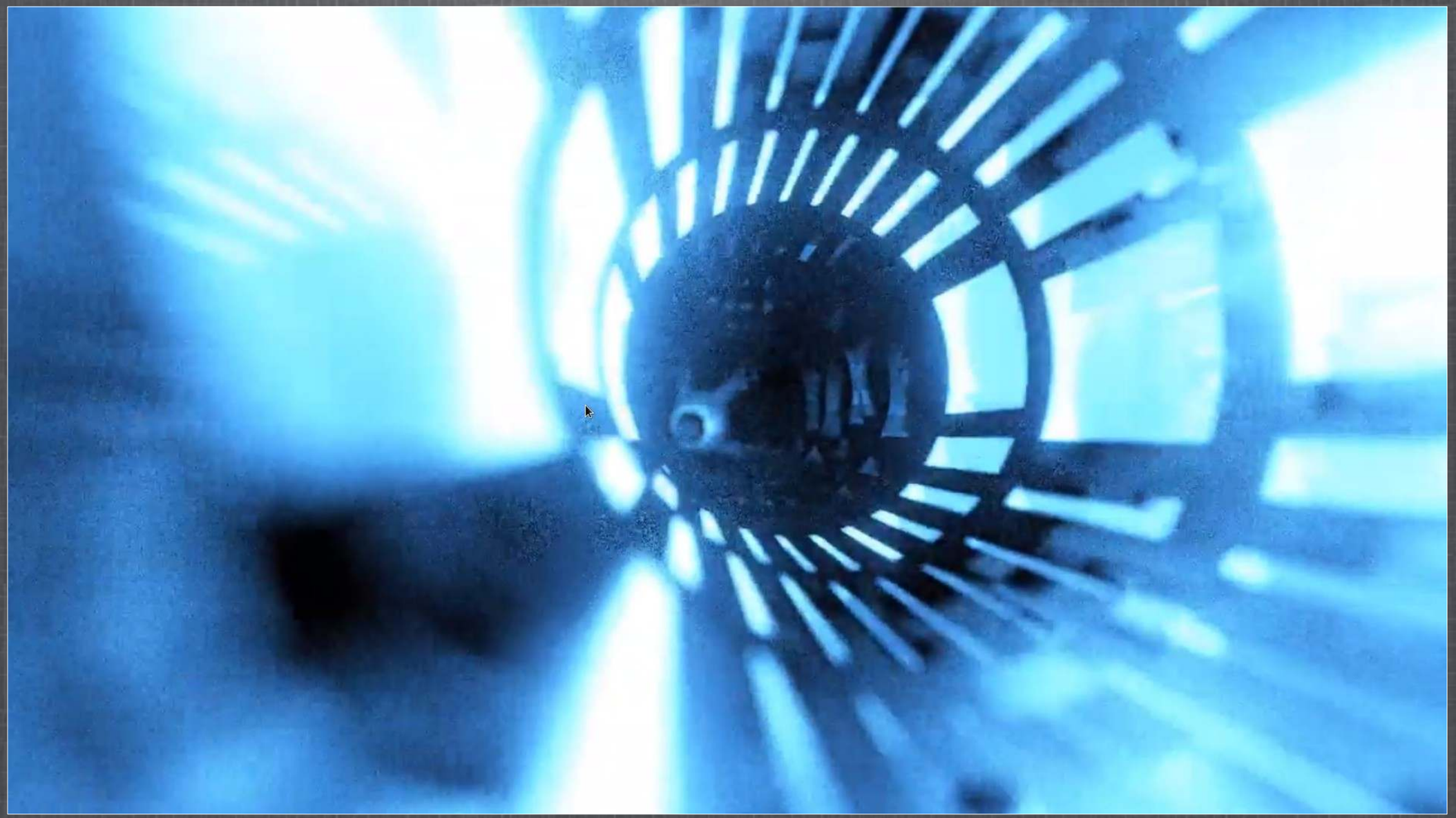








DEMO SCENE



Sync Cord by NuSan & Valden (PC 4k intro, 2020)

REFLECTIONS

Real-time Ray-Tracing in UE4 (2018)

RENDERING

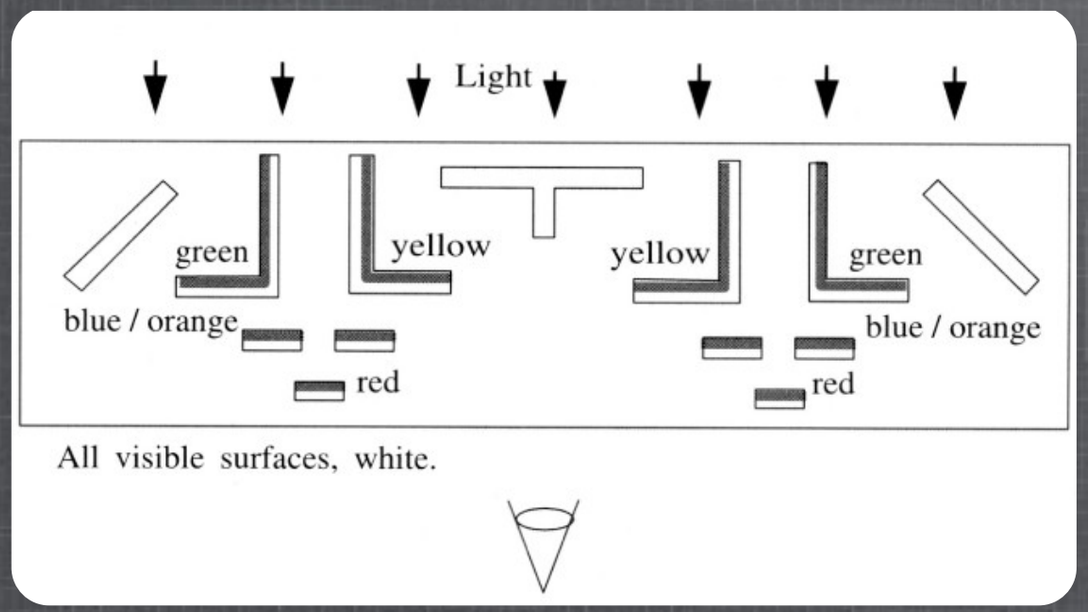
Global Illumination (1980's and 1990's)

Raytracing
(Whitted 1980)

Radiosity
(Goral, Torrance et al. 1984)

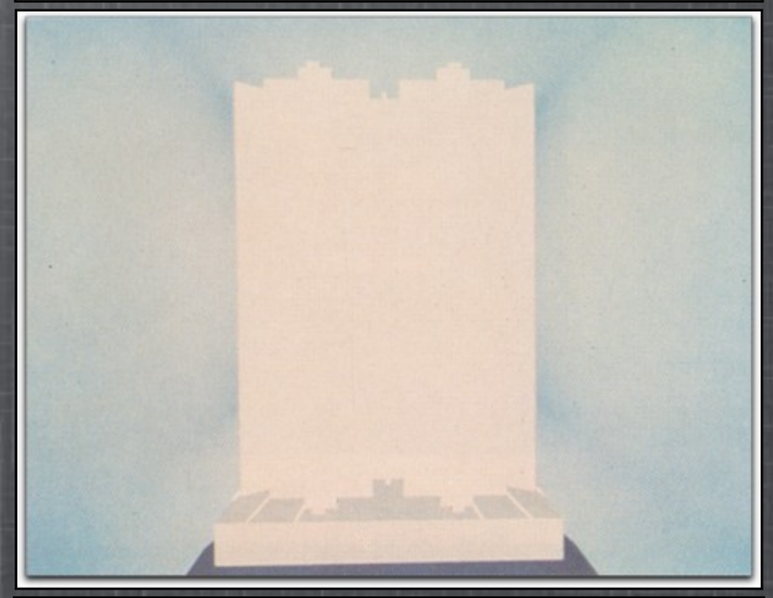
RENDERING

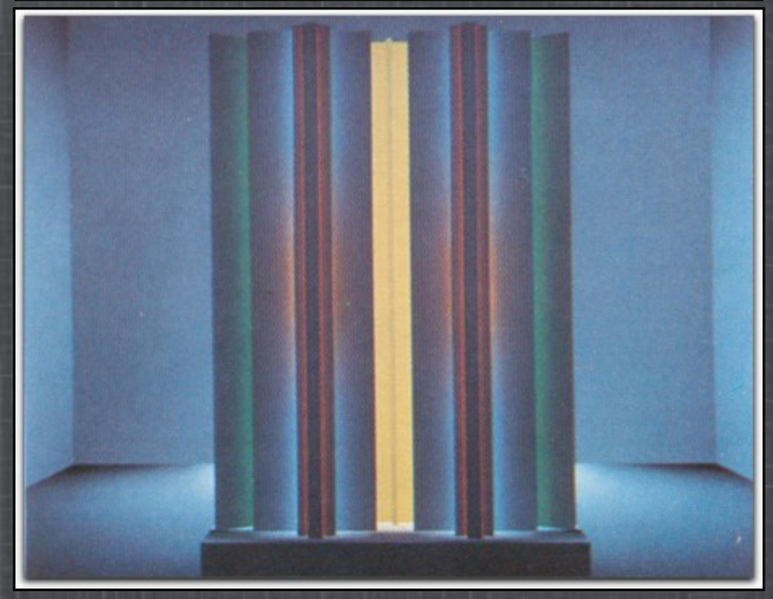
Radiosity (Goral, Torrance et al. 1984)



Sculpture by John Ferren

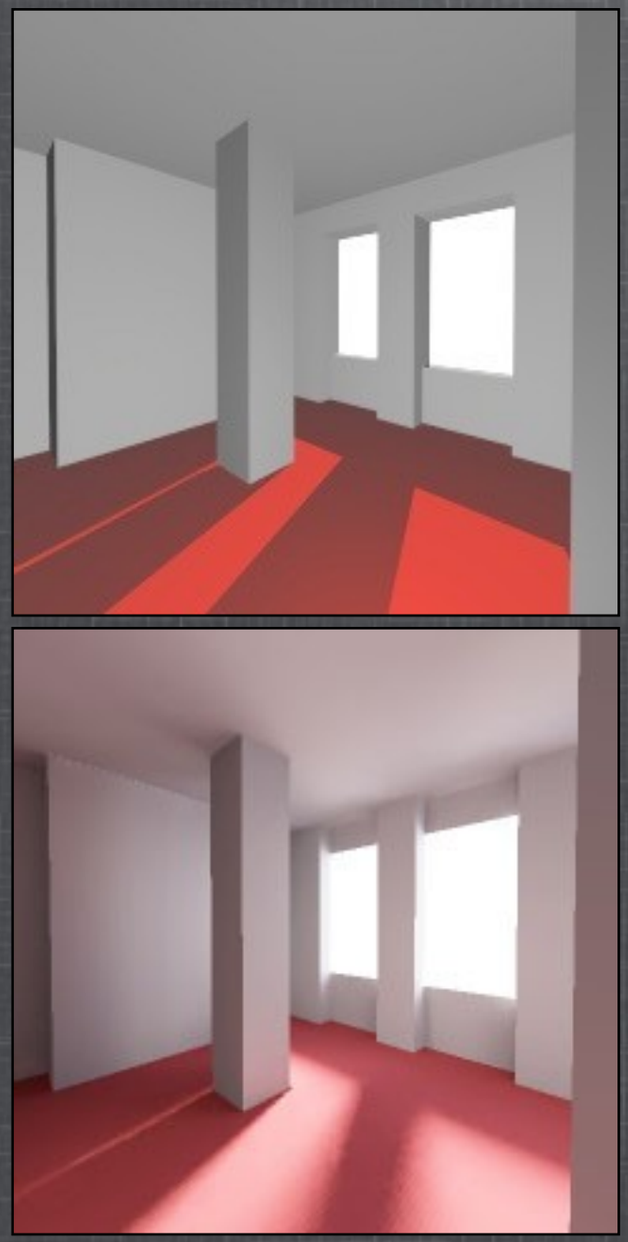
Goral et al. 1984

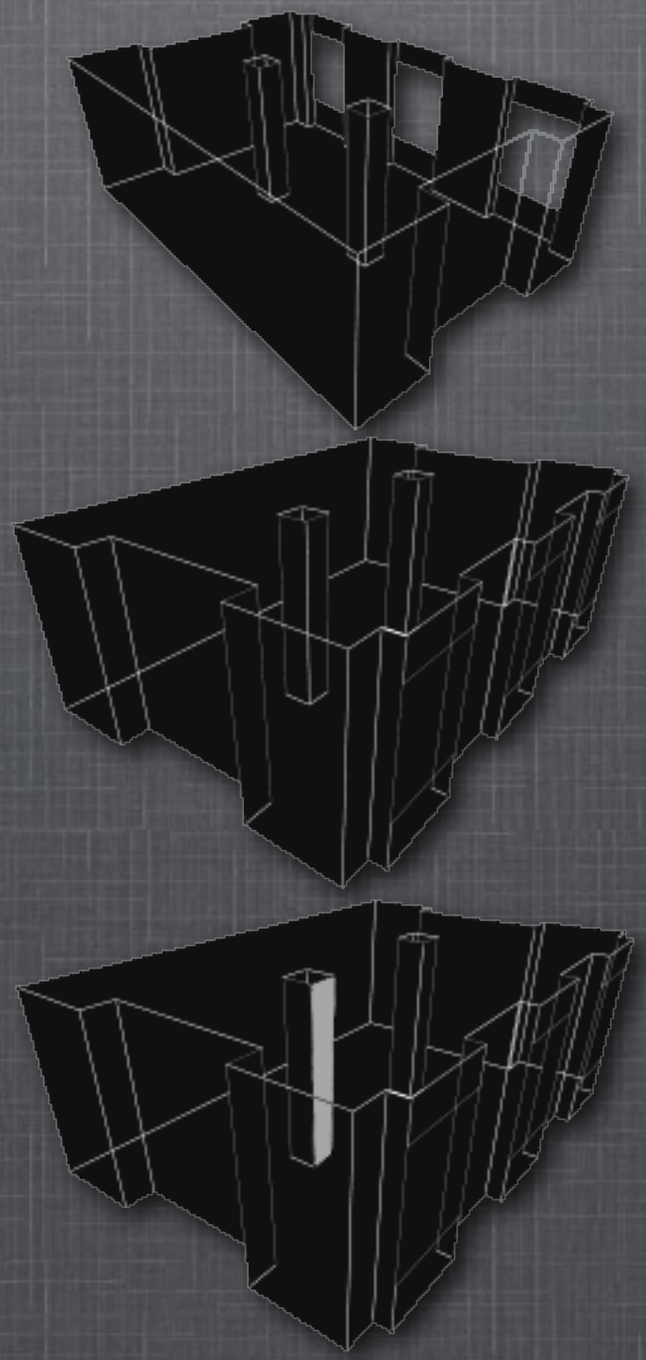


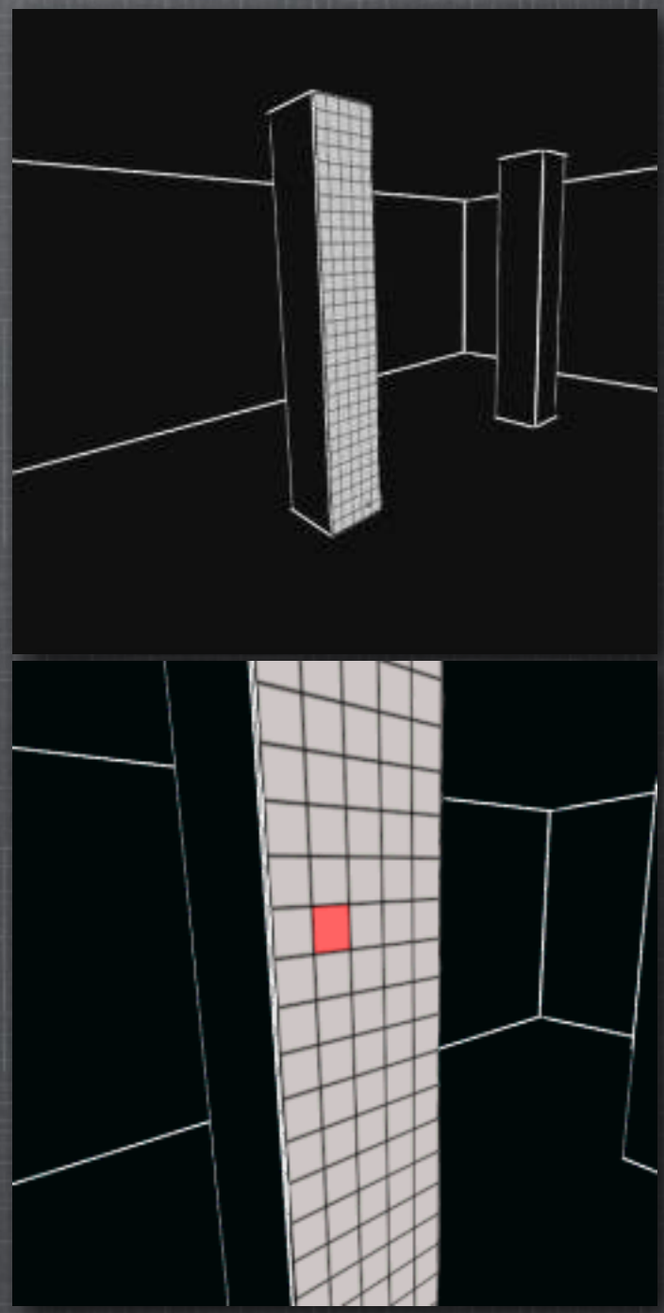


RENDERING

Radiosity (Goral, Torrance et al. 1984)

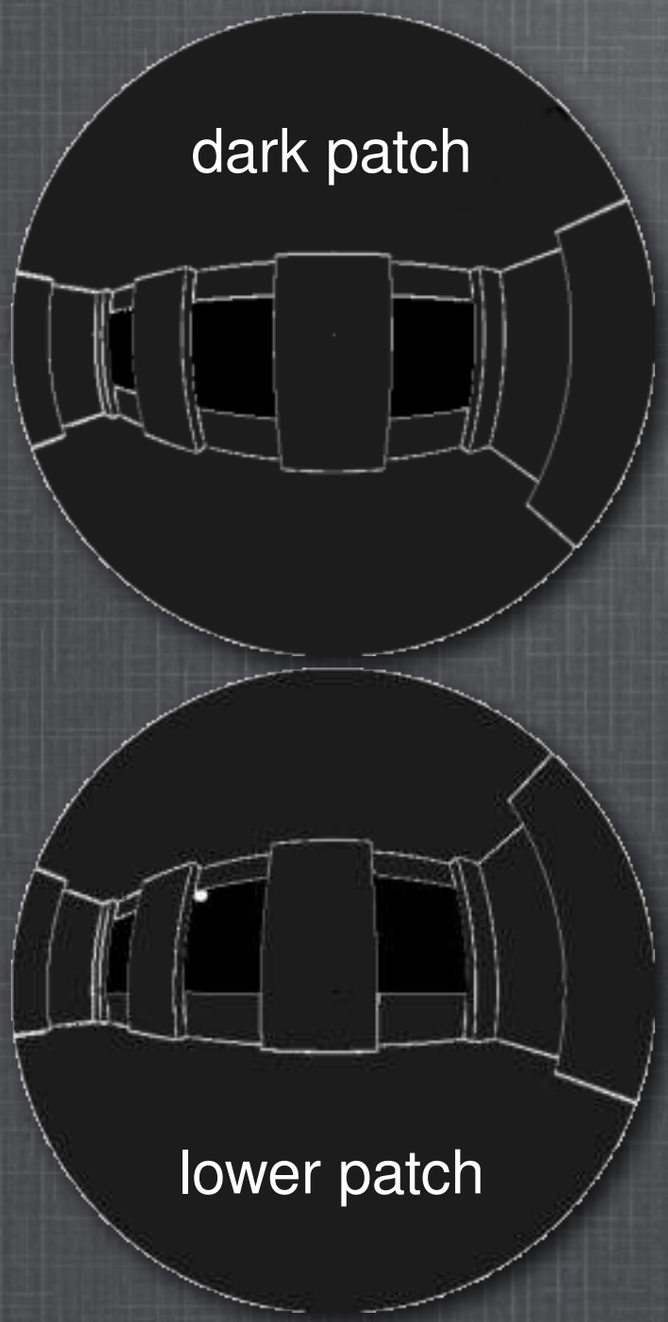


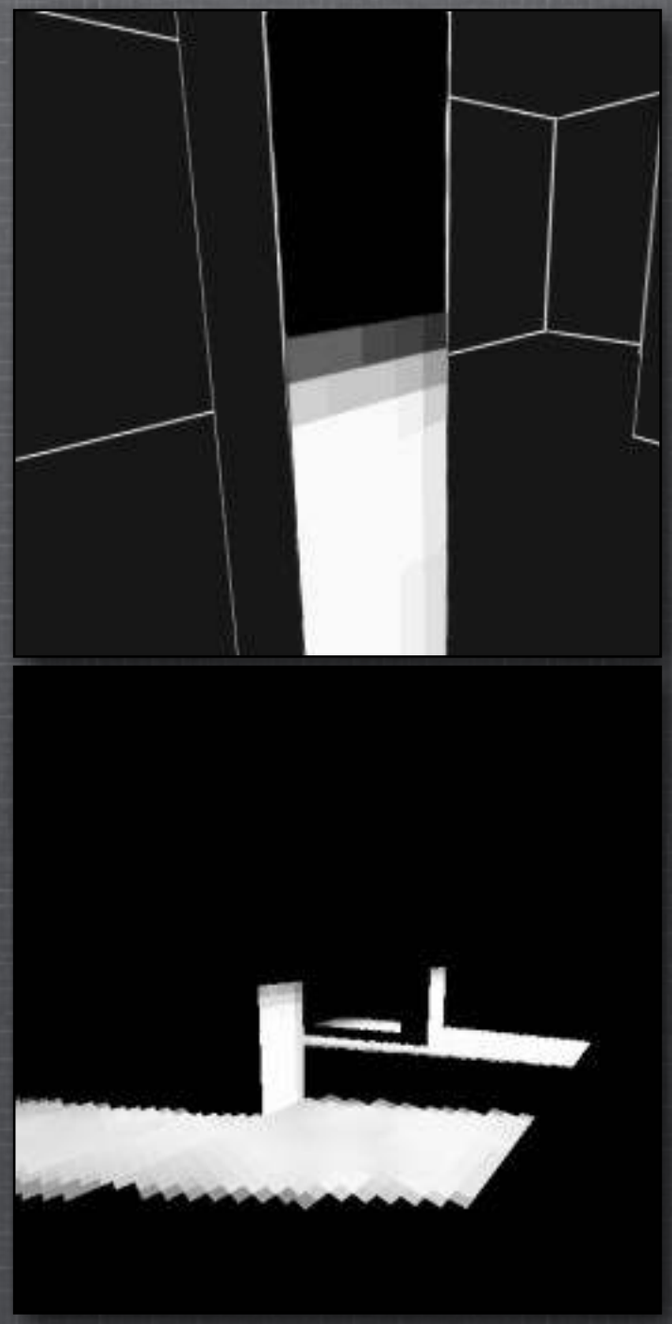




RENDERING

Radiosity (Goral, Torrance et al. 1984)

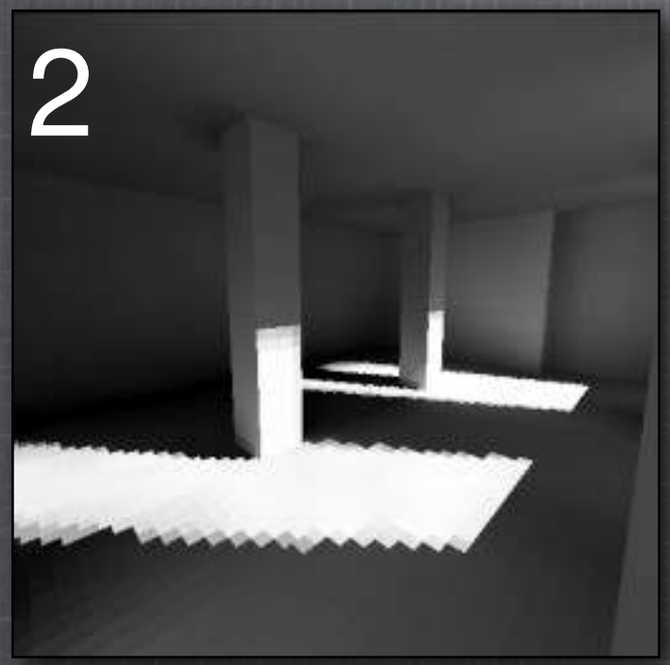


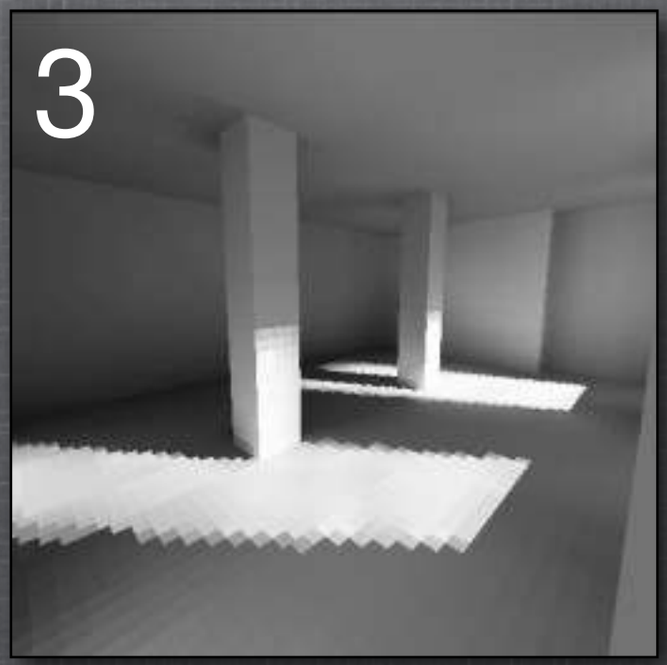


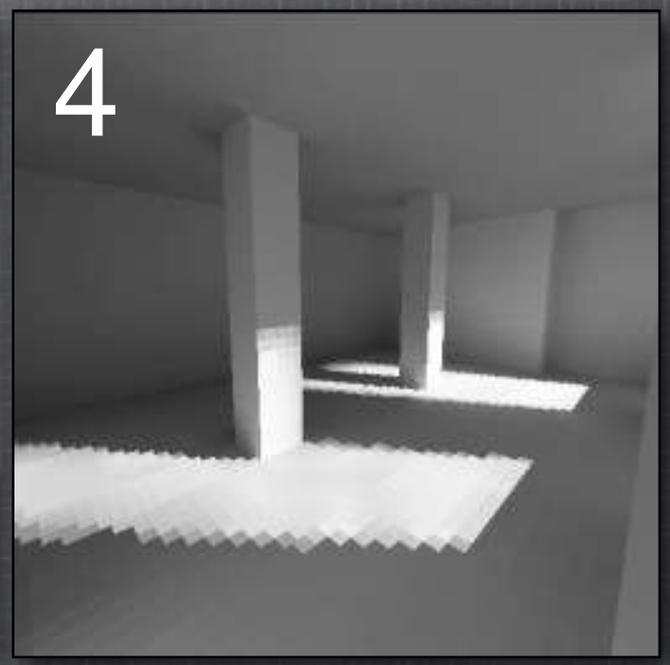
dark patch
not dark
anymore

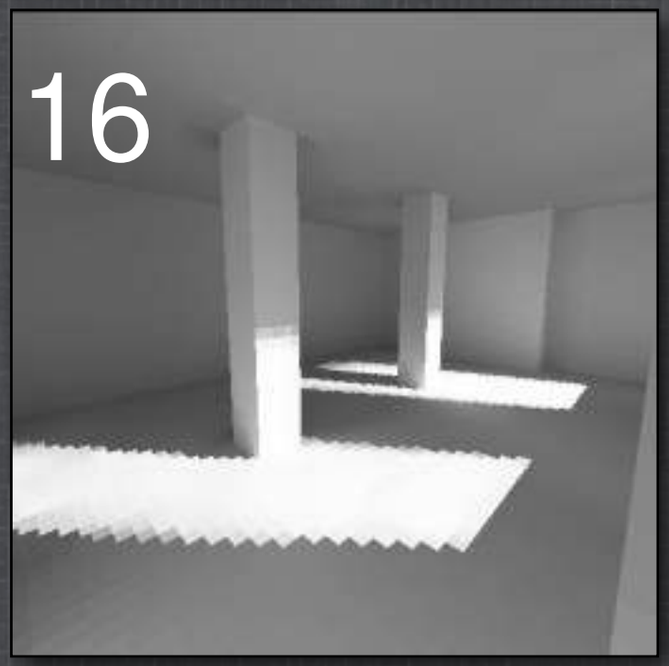
RENDERING

Radiosity (Goral, Torrance et al. 1984)



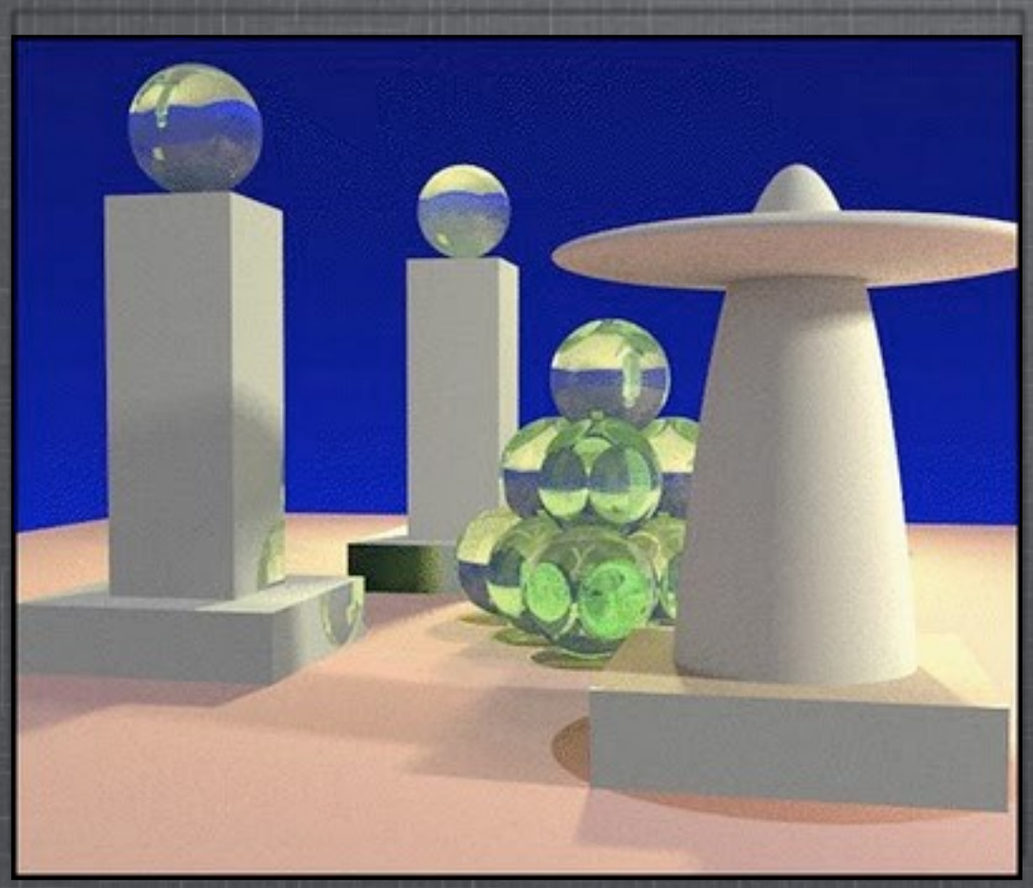






RENDERING

Global Illumination (1980's and 1990's)



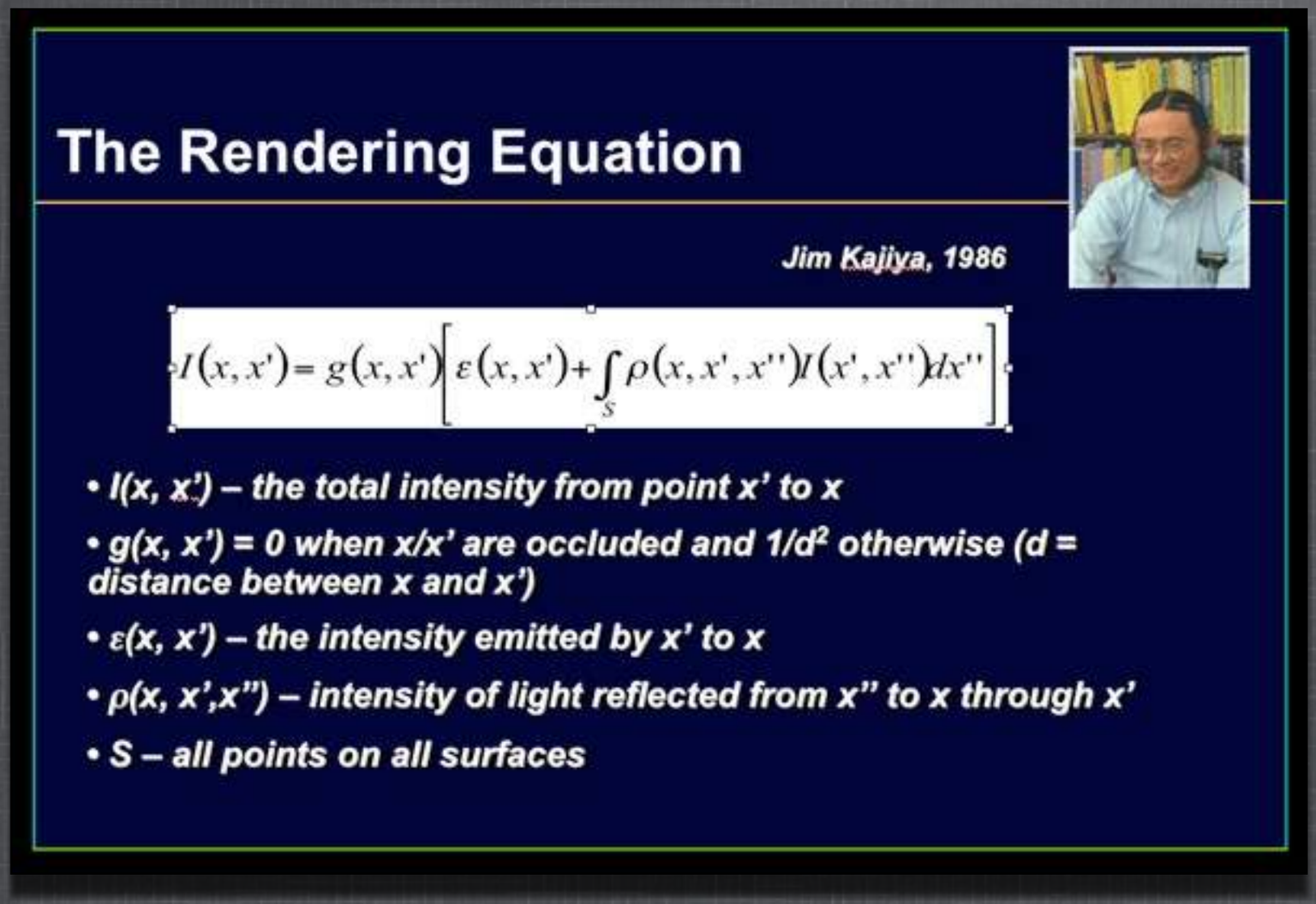
Raytracing
(Whitted 1980)

Radiosity
(Goral, Torrance et al. 1984)

Rendering equation
(Kajiya, 1986)

RENDERING

Global Illumination (1980's and 1990's)



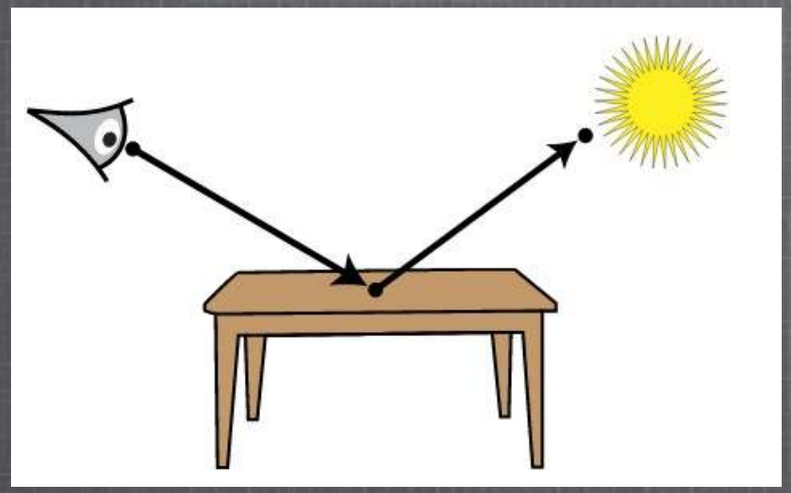
Raytracing
(Whitted 1980)

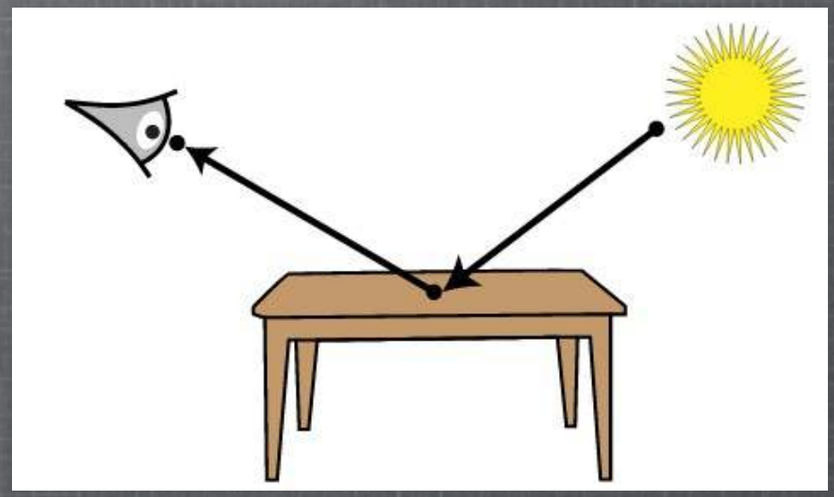
Radiosity
(Goral, Torrance et al. 1984)

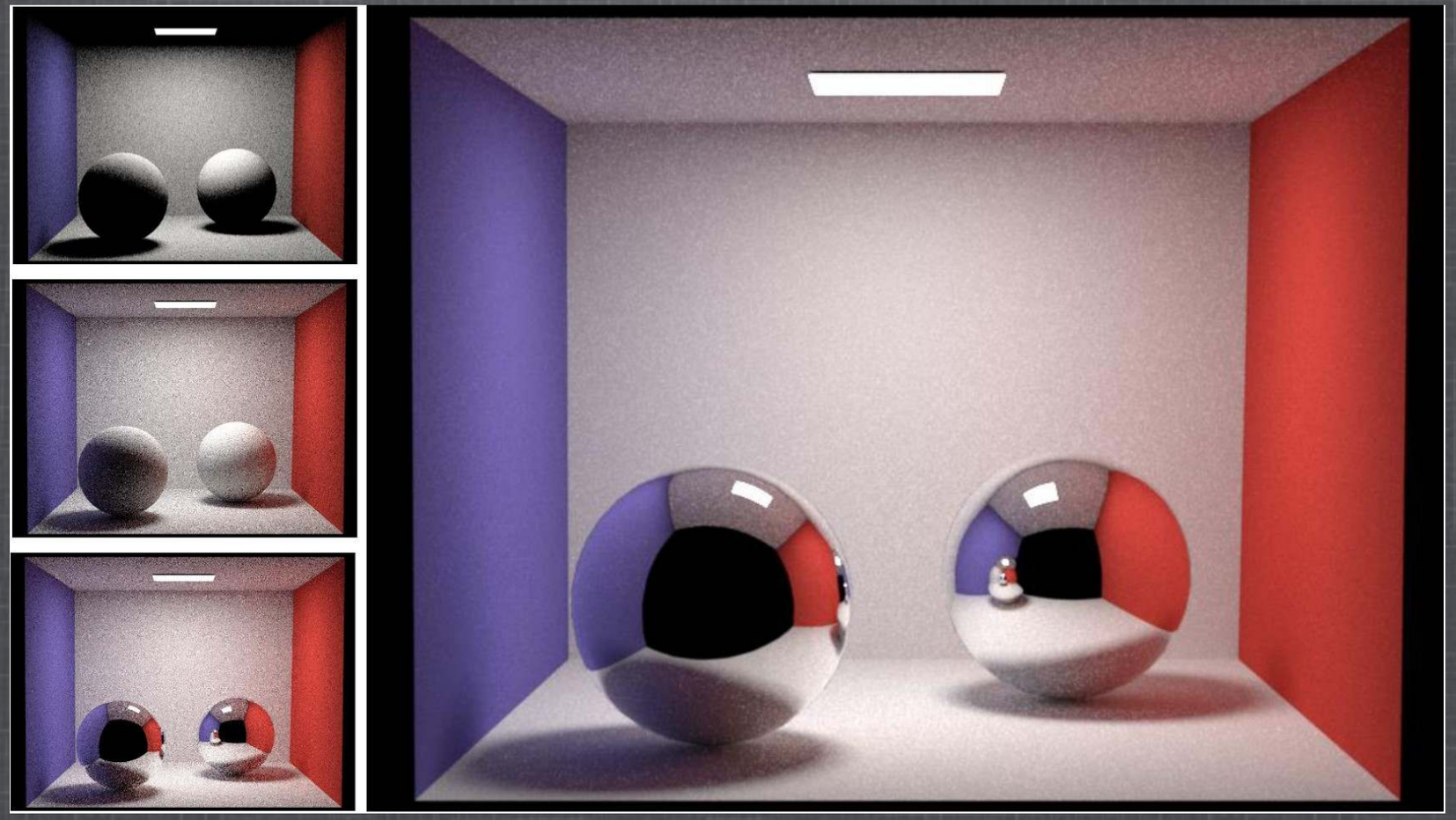
Rendering equation
Path Tracing
(Kajiya, 1986)

RENDERING

Path Tracing

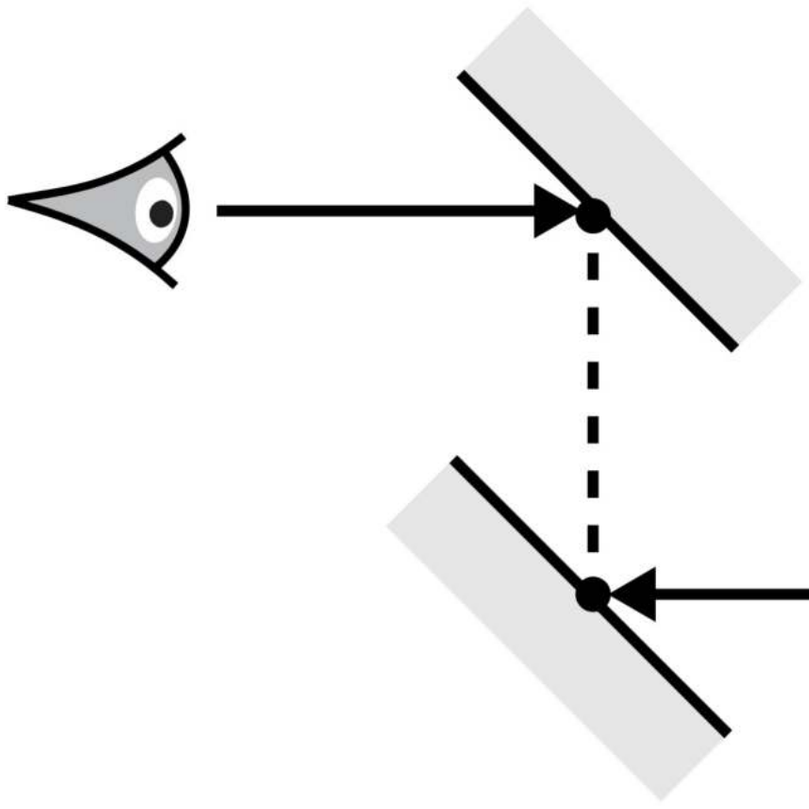


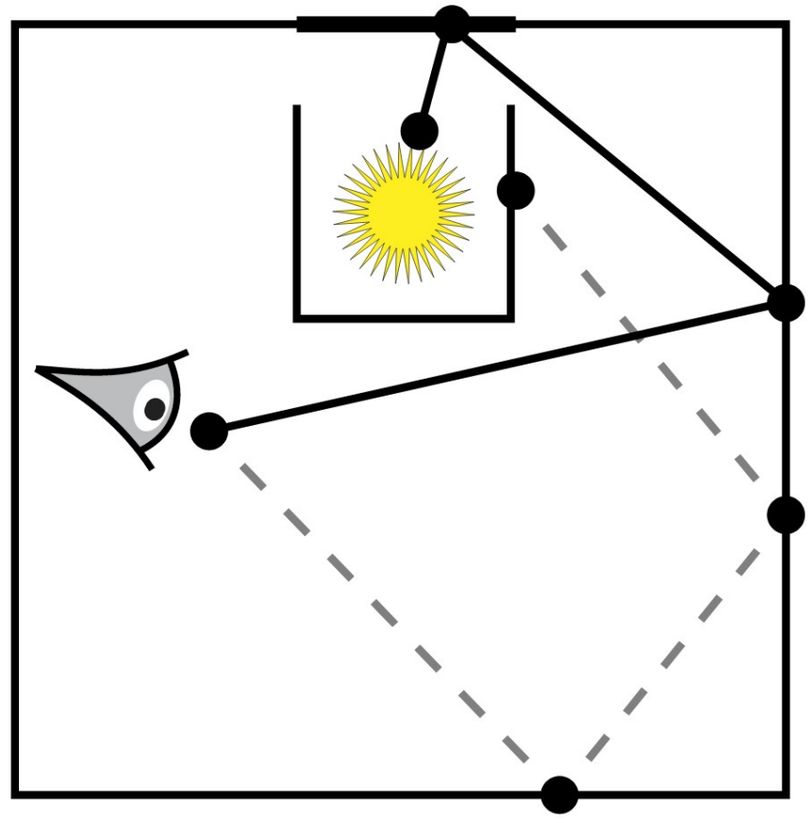




MODERN
PATH TRACING

Bidirectional Path Tracing



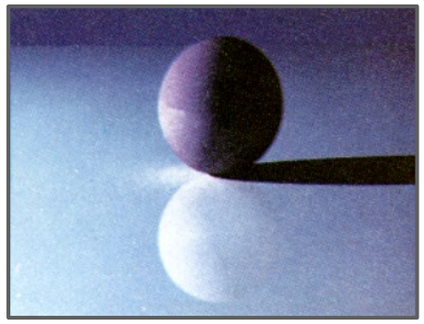


RENDERING

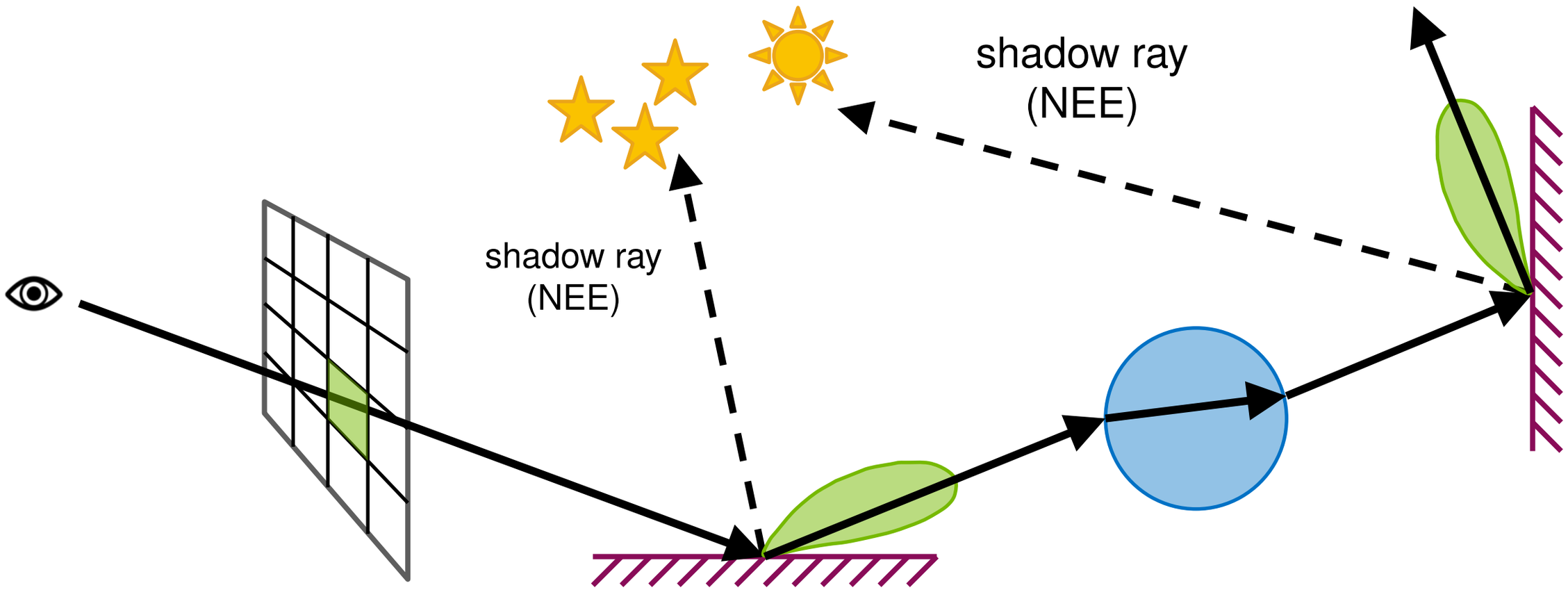
(early) Realtime Path Tracing

RENDERING EQUATION

Kajiya @ SIGGRAPH 1986



MODERN PATH TRACING



RENDERING WITH DENOISING

Realtime Path Tracing NVIDIA 2022 (30FPS, 1080p, 30 bounces, 30B triangles)

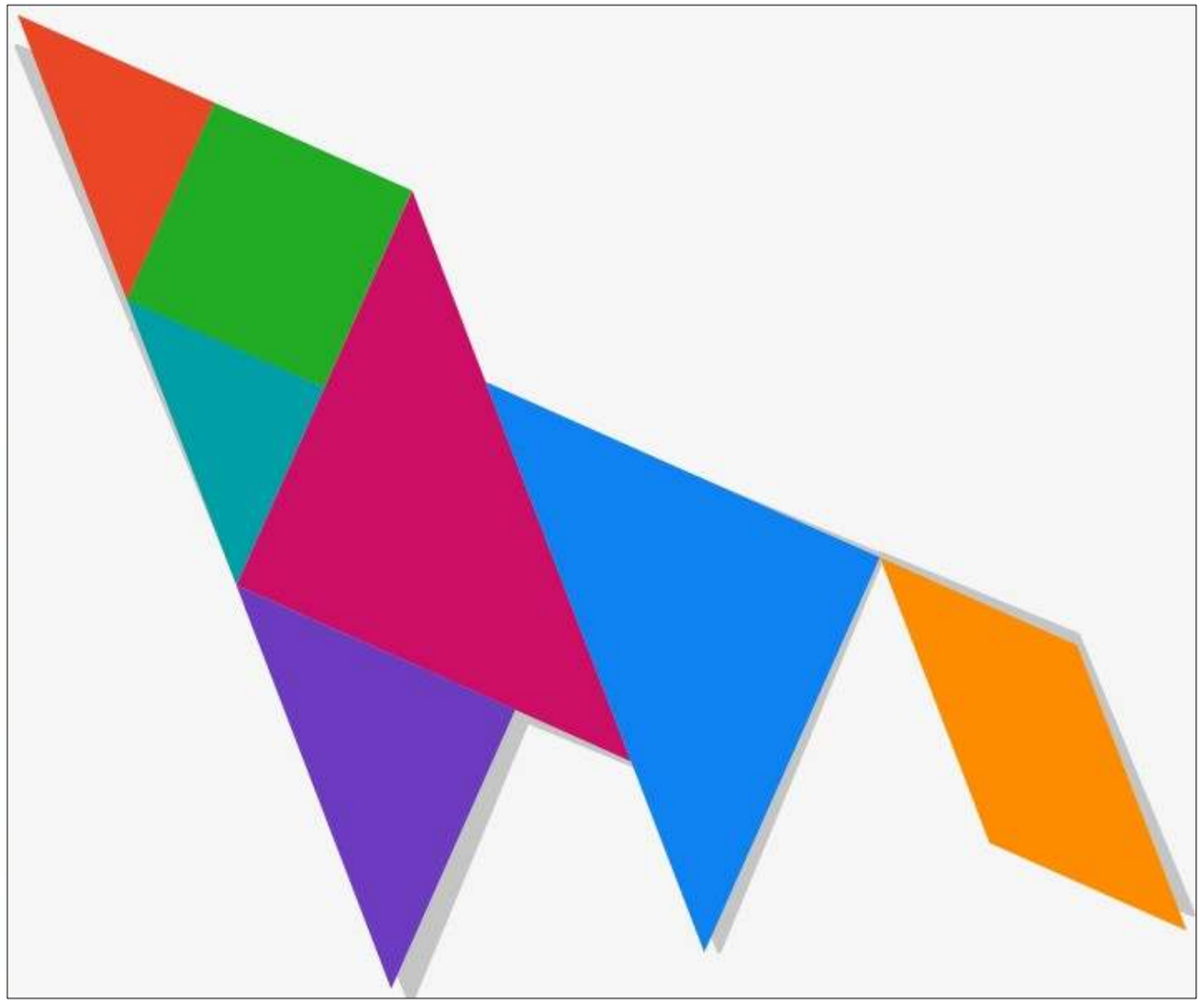
Realtime Path Tracing NVIDIA 2022 (30FPS, 1080p, 30 bounces, 30B triangles)

QUARTER PLANNING

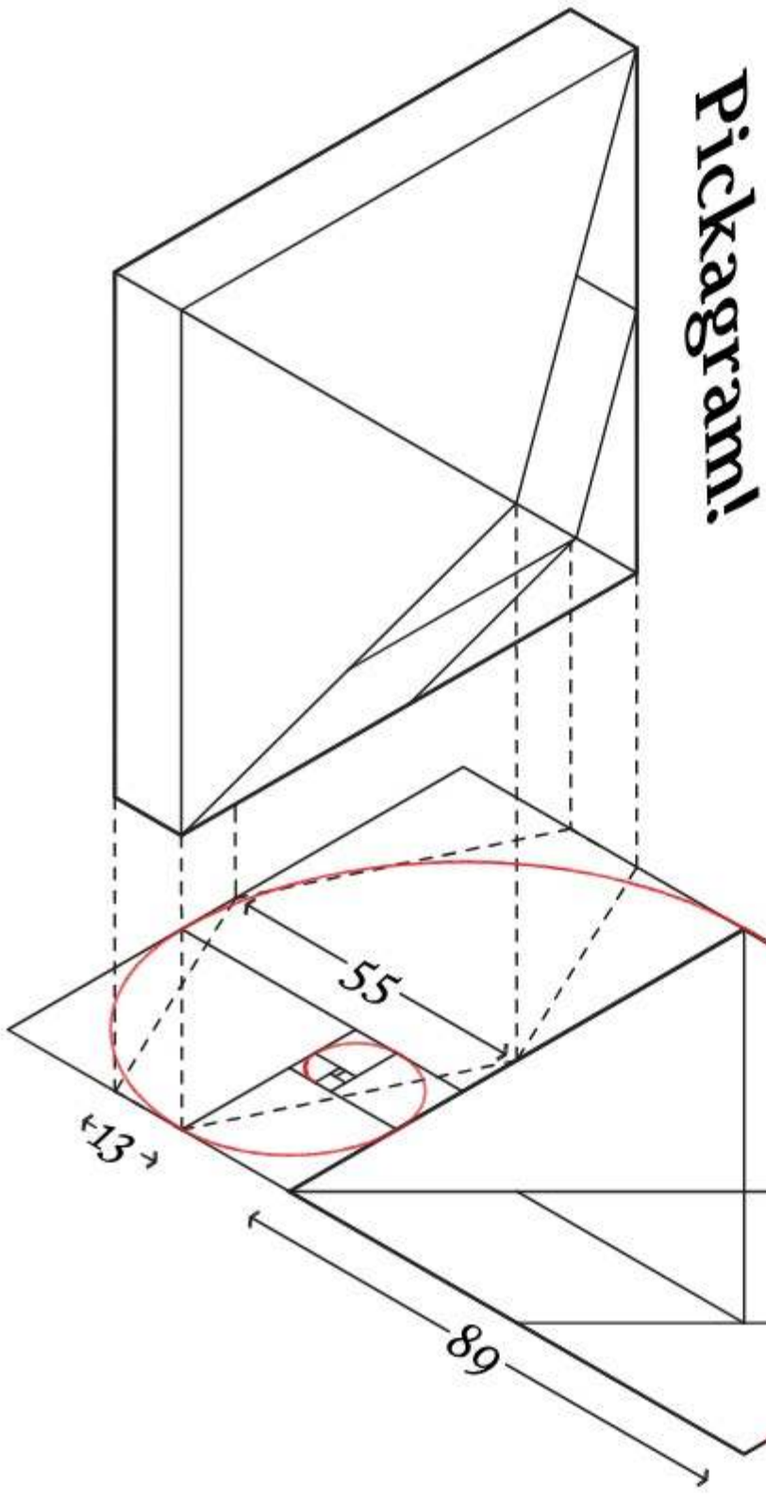
November 2025

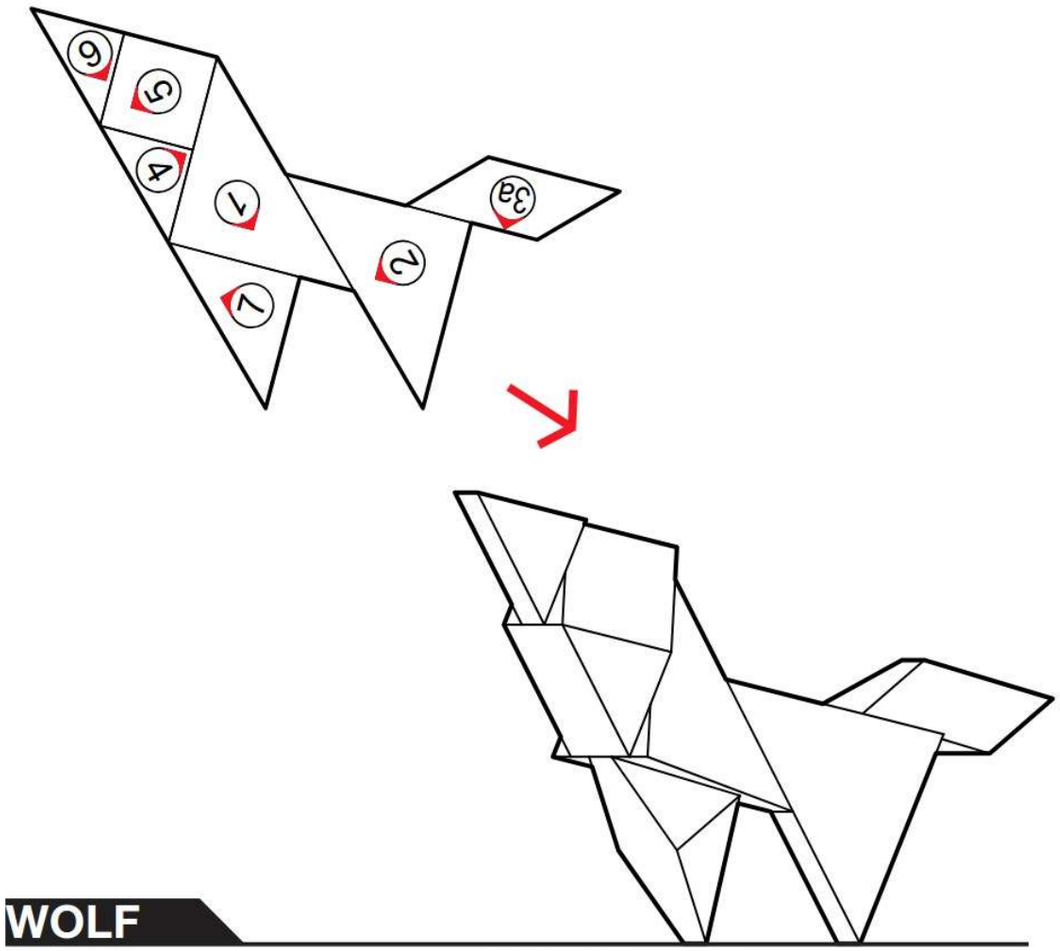
W	Date	Lectures	Labs	Assignments due	Grade
1	Nov 11 (Tue)	Introduction, Review	Team forming C++ IDE		
1	Nov 14 (Fri)	Mathematics for CG (part 1): dot, cross, matrices	GLM / GLM setup		
2	Nov 18 (Tue)	Mathematics for CG (part 2): matrices, quaternions	GLM library	**A1 : GLM (50% in class)** Nov 20 (Thu)	**1.0**
2	Nov 21 (Fri)	Programmable Graphics Pipeline, Drawing in OpenGL	OpenGL / GLFW / GLEW Setup	Tan/Picka-gram Shape (in class)	
3	Nov 25 (Tue)	Viewing Pipeline Viewing in OpenGL	Drawing + Transformations	**A2 : 2D Tangram** Nov 27 (Thu)	**3.0**
3	Nov 28 (Fri)	Geometry Management Meshes and Scene Graphs	Viewing + Quaternions		

Tangram

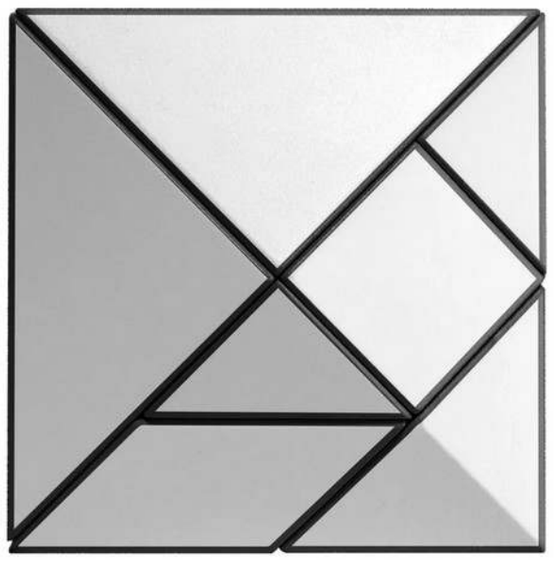


Pickagram



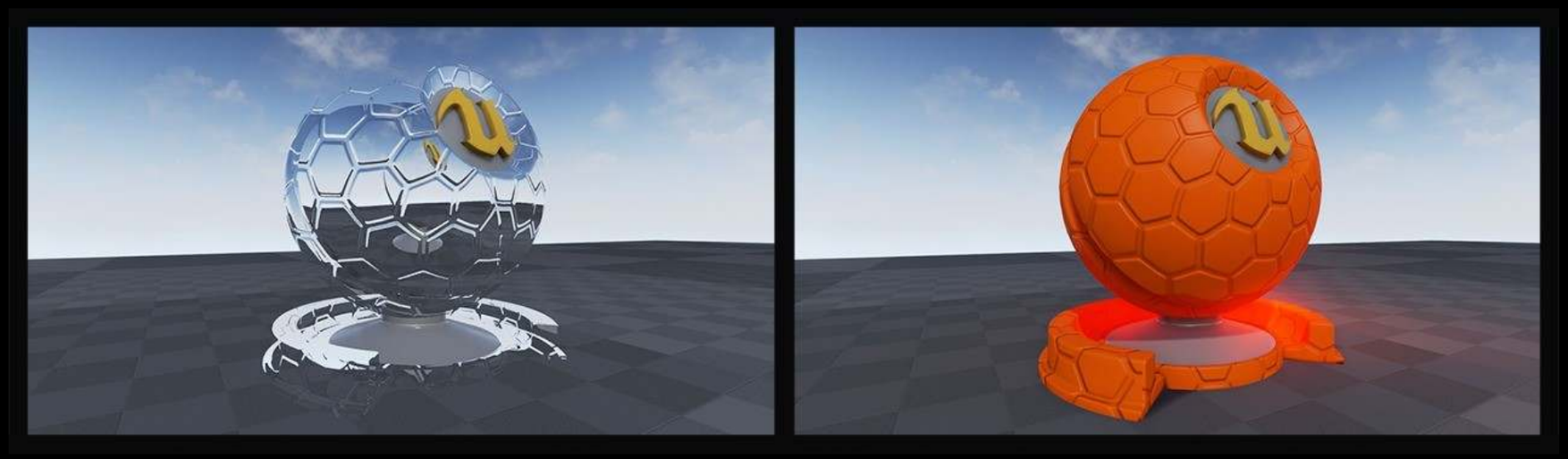


WOLF



December 2025

W	Date	Lectures	Labs	Assignment due	Grade
4	Dec 2 (Tue)	GLSL Shaders Lighting (1): Cel Shading	Meshes + ASSIMP Library		
4	Dec 5 (Fri)	Lighting (2): Gouraud and Phong Shading	Project Support	**A3 : 3D Pickagram** Dec 8 (Mon)	**4.0**
5	Dec 9 (Tue)	Mapping (1): Texturing, Sampling and Filtering	Shader Project Individual Project Definition	**A4 : Project Proposal** Dec 11 (Thu)	**1.0**
5	Dec 12 (Fri)	Mapping (2): Procedural Textures and Noise	Goal Definition and Execution Planning		
6	Dec 16 (Tue)	Mapping (3): Environment Mapping and Fresnel Reflectance	Project Support		
6	Dec 19 (Fri)	Mapping (4): Render Target Textures, Tangent Space, Normal and Displacement Mapping	Project Support		

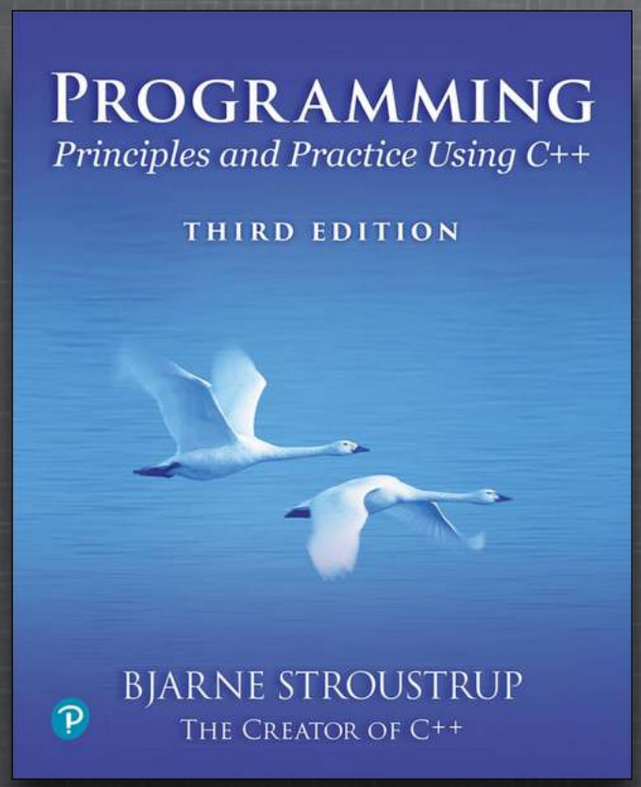


January 2026

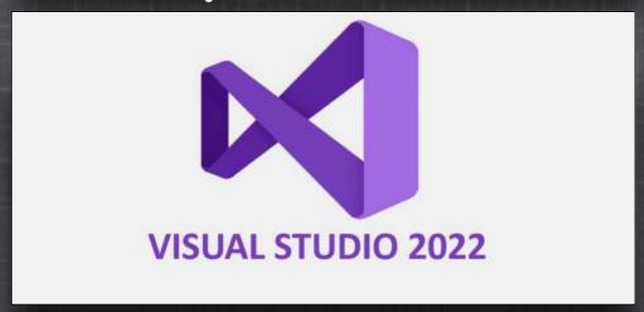
W	Date	Lectures	Labs	Assignments Due	Grade
7	Jan 6 (Tue)	Mapping (5): Ambient Occlusion and Physically Based Rendering	Project Support		
7	Jan 9 (Fri)	(project report Q&A)	Project Support	**A5 : Shader Project** Jan 9 (Fri)	**8.0**
8				**A5: Project Report** Jan 12 (Mon) **Project Discussions** Jan 15-16 (Thu-Fri)	**3.0**

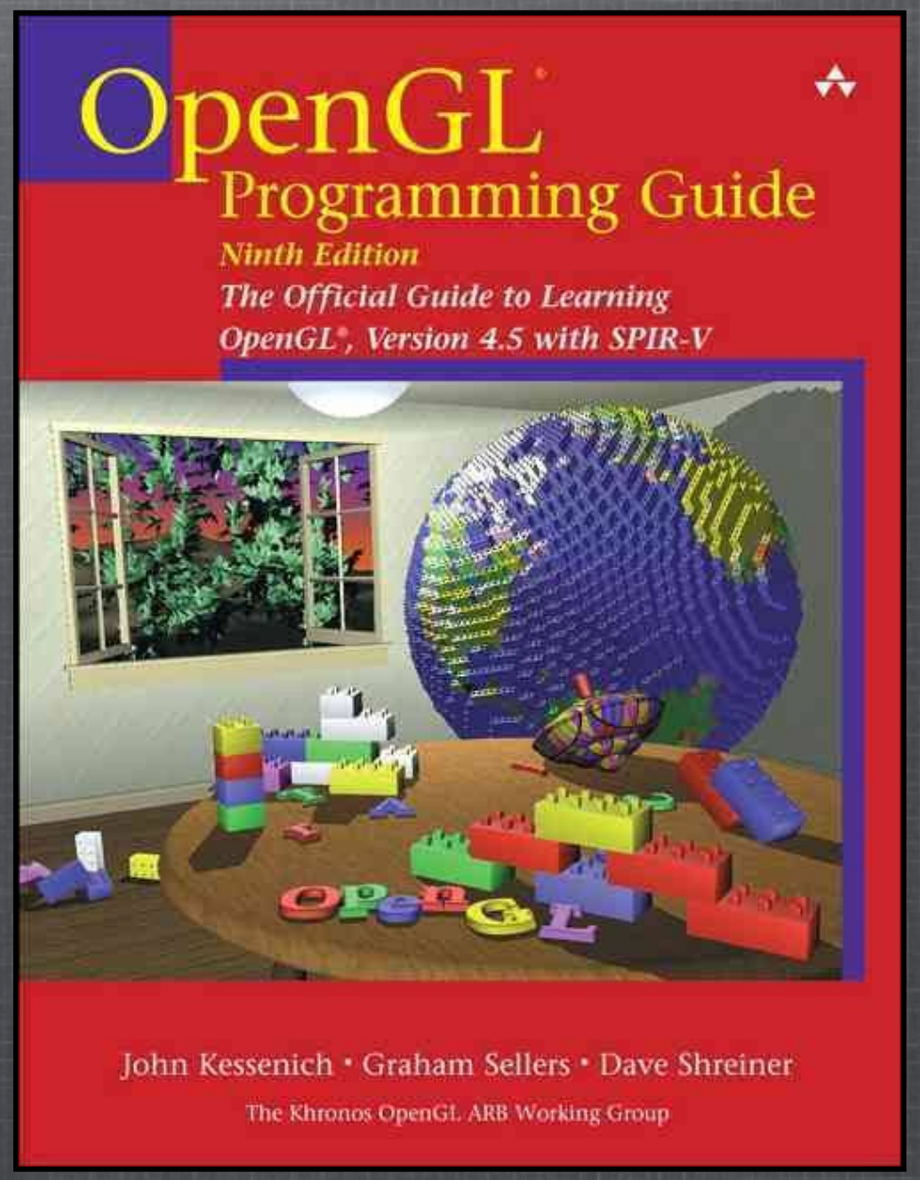
IN BRIEF

- 40% - Base Project (teams of 2, 3 checkpoints)
- 60% - Shader Project (individual, 2 checkpoints)



Choose your C++/GLSL IDE





CONTACT

CARLOS.MARTINHO@TECNICO.PT

CGJ Office hours on Fénix.