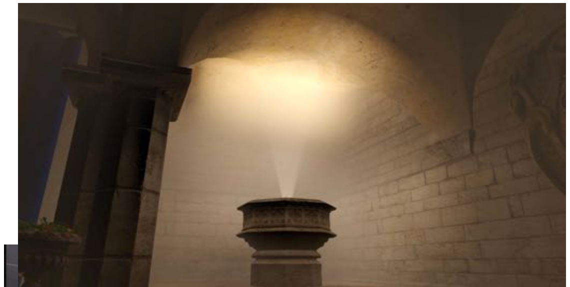
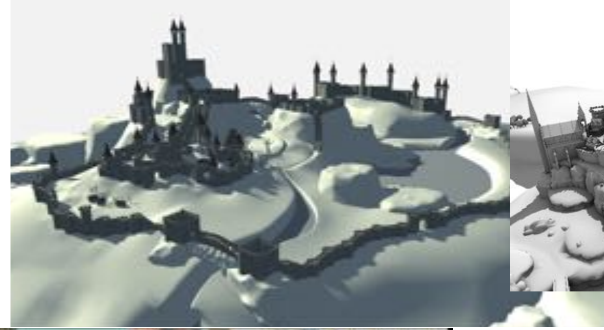
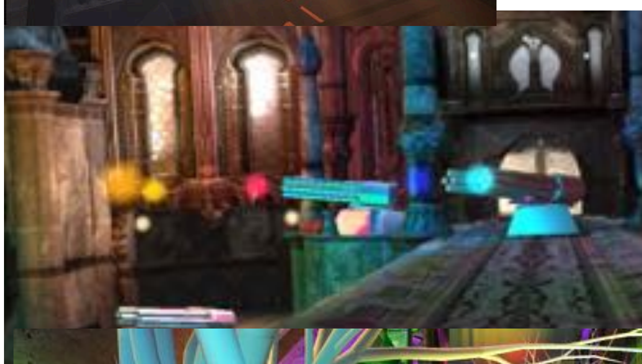






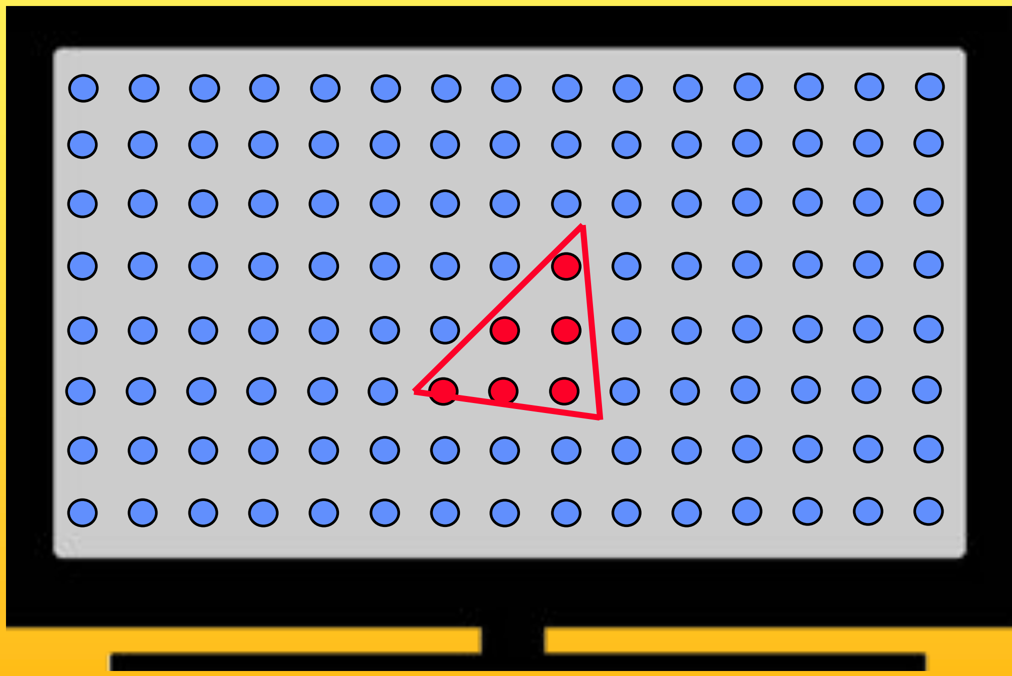



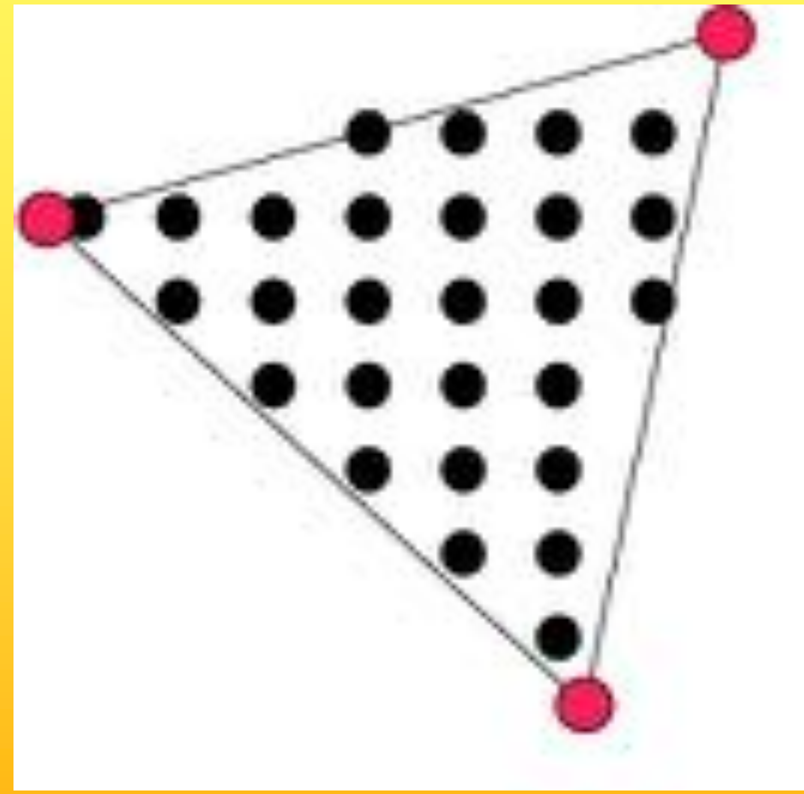

Realistic Real-time Rendering Today and in the Future

Ulf Assarsson

Chalmers University of Technology

The screen consists of pixels

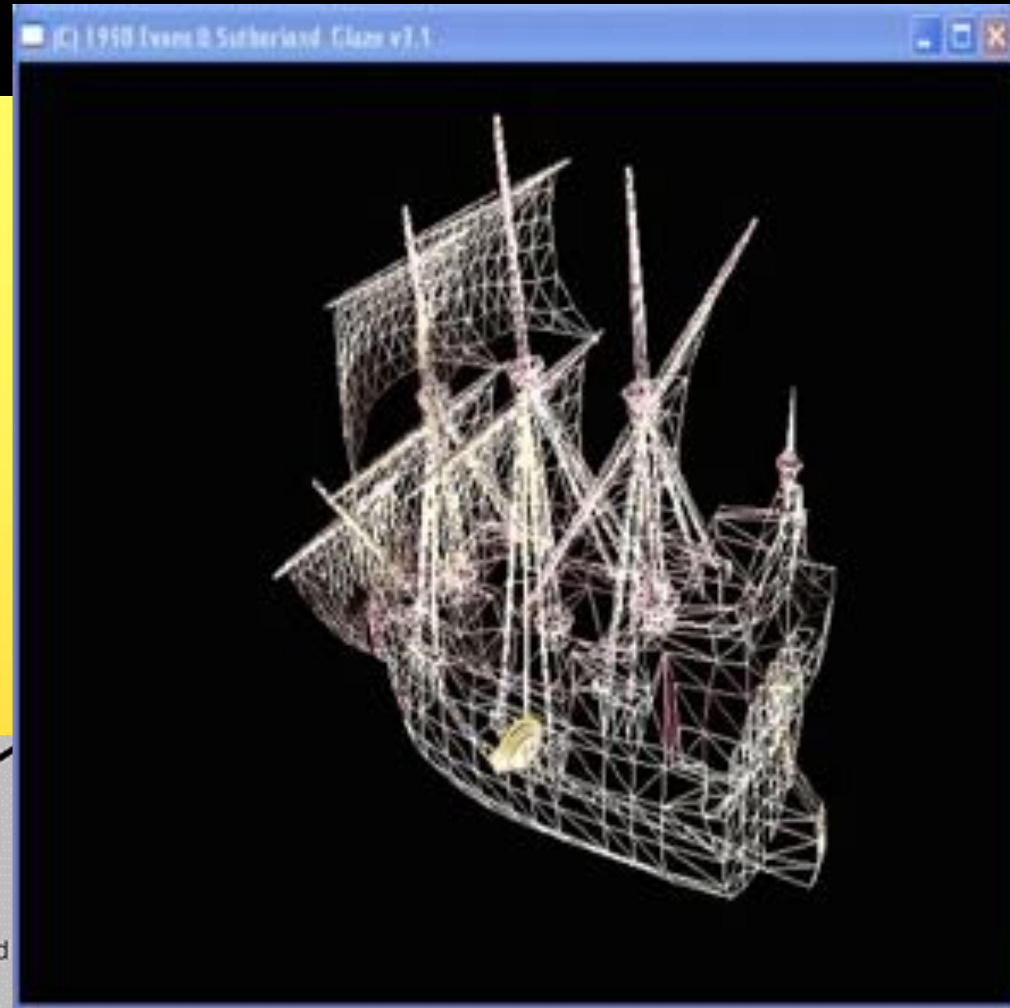


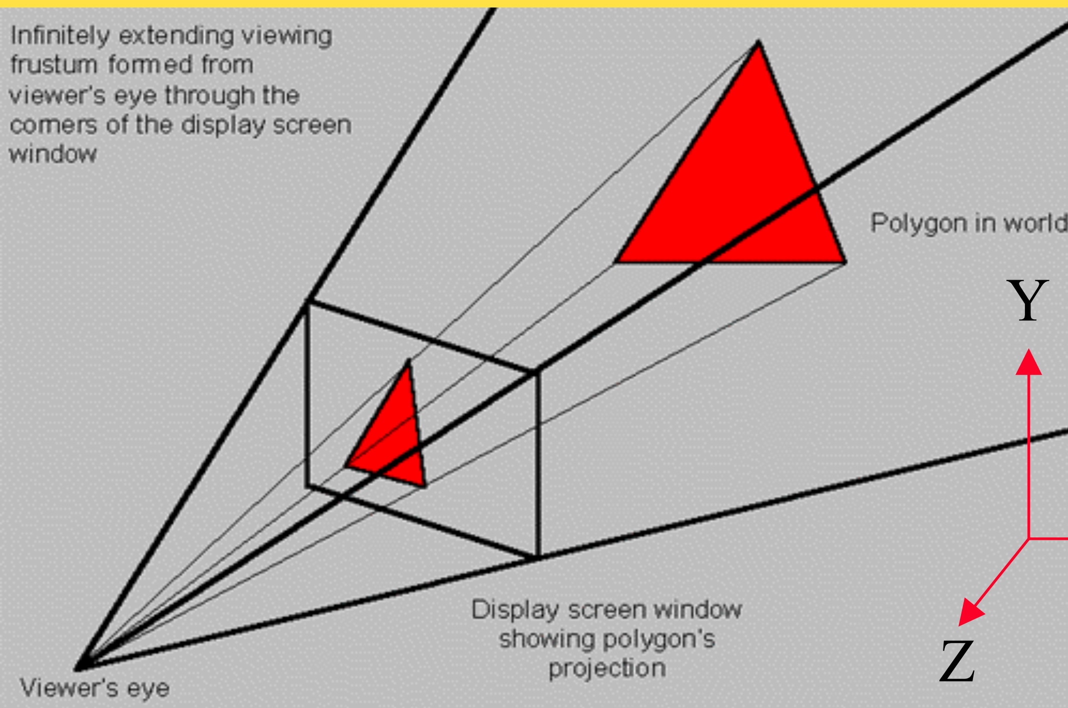


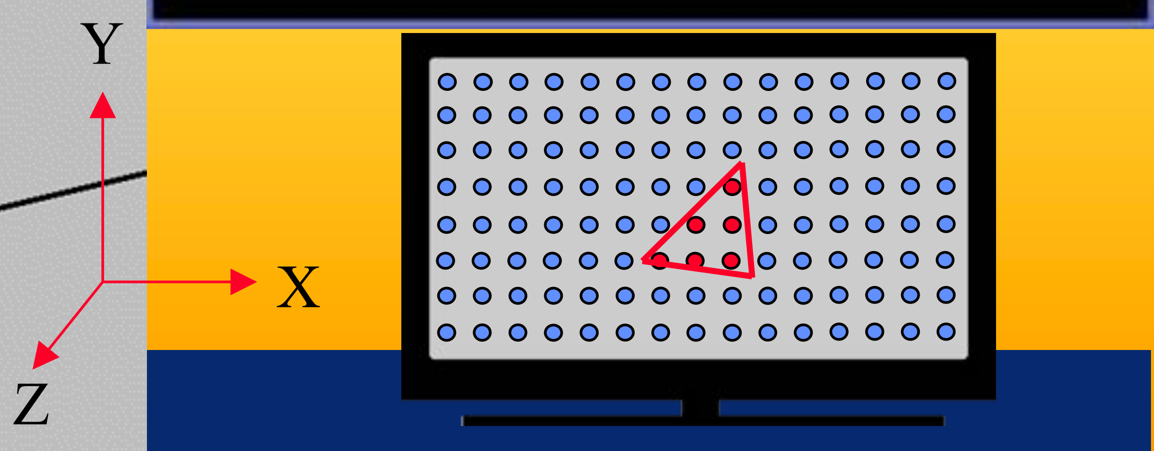
Grafikkort

3D-Rendering

- Objects are often made of triangles
- x,y,z- coordinate for each vertex







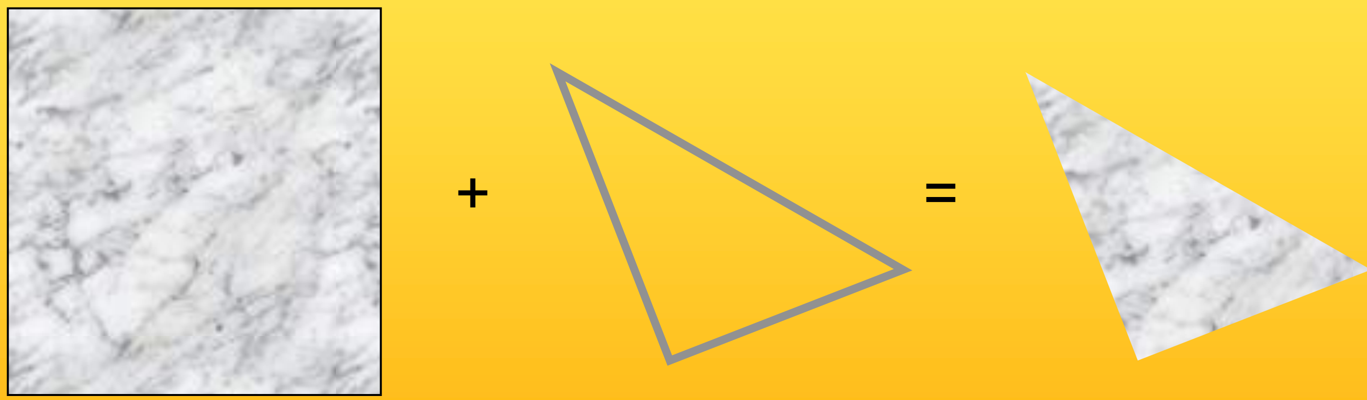
4D Matrix Multiplication

$$\begin{bmatrix} s_x & \bullet & \bullet & t_x \\ \bullet & s_y & \bullet & t_y \\ \bullet & \bullet & s_z & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$

Real-Time Rendering

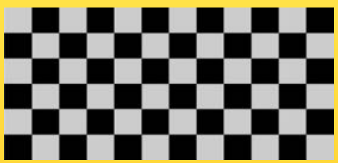
Textures

- One application of texturing is to ”glue” images onto geometrical object



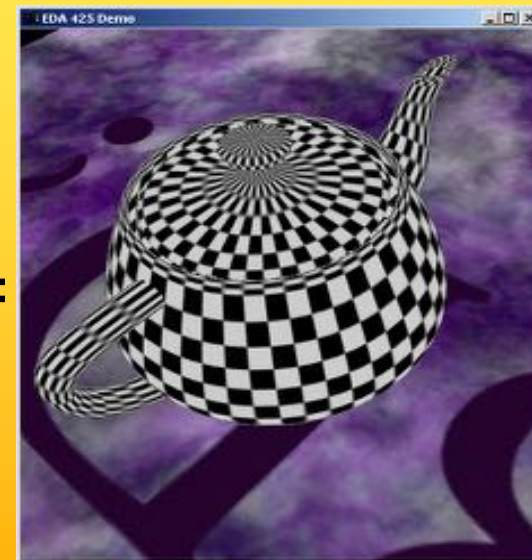
Texturing: Glue images onto geometrical objects

- Purpose: more realism, and this is a cheap way to do it

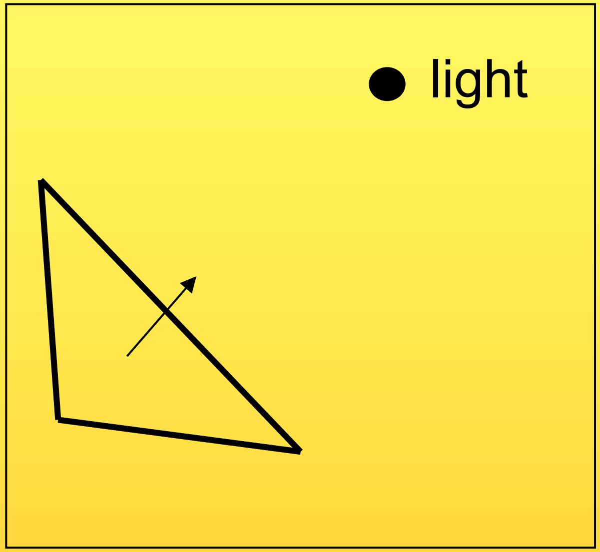


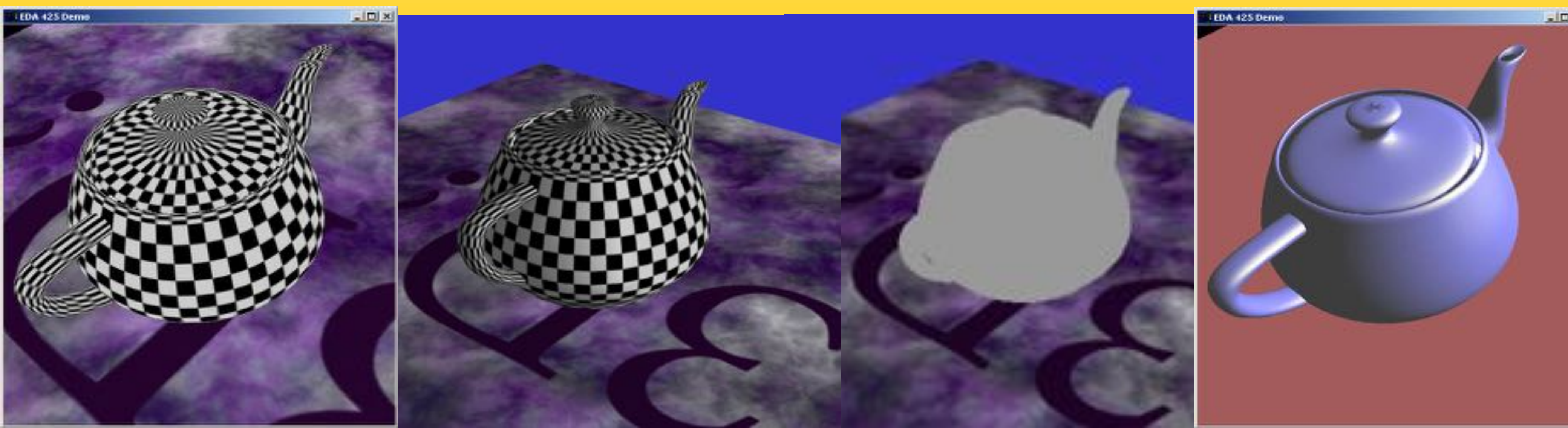
+

=

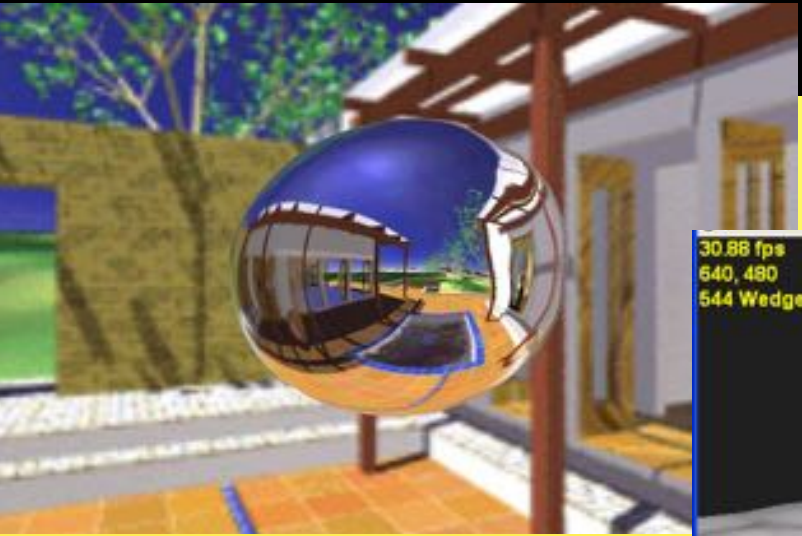


Light computation per triangle





More



Reflections

Shadows

Materials

However, this does not look very realistic! Why?

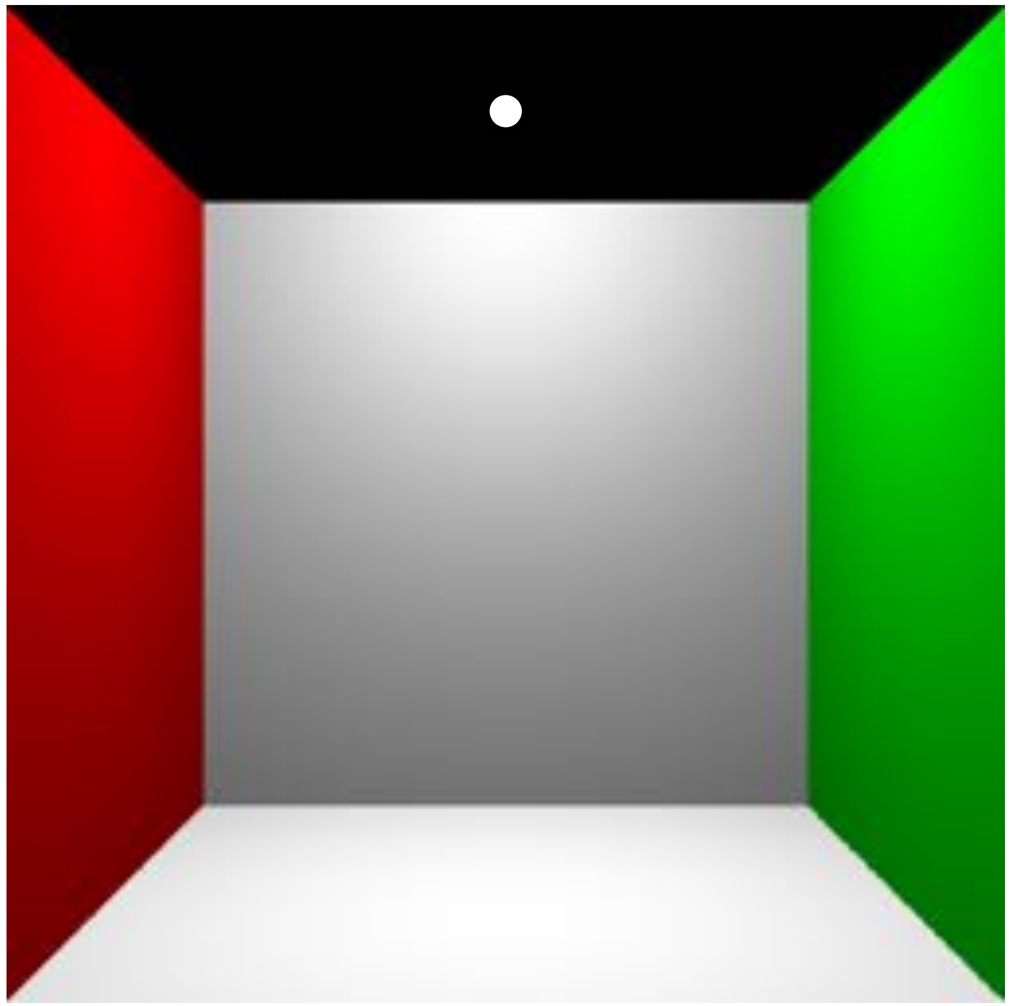
Water

Airlight

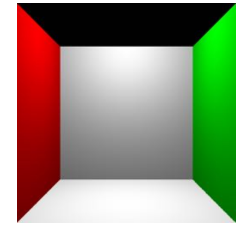
Fire

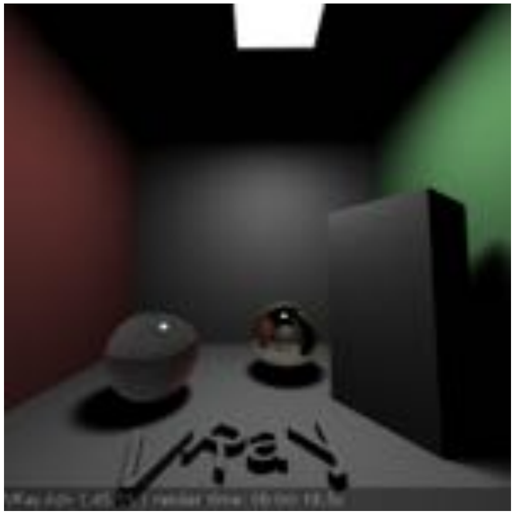
Light Bounces

Typical test box (Cornell box), often compared to actual photograph:

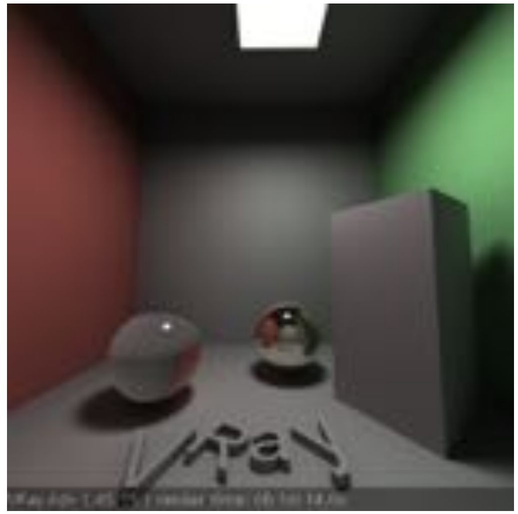


Light Bounces

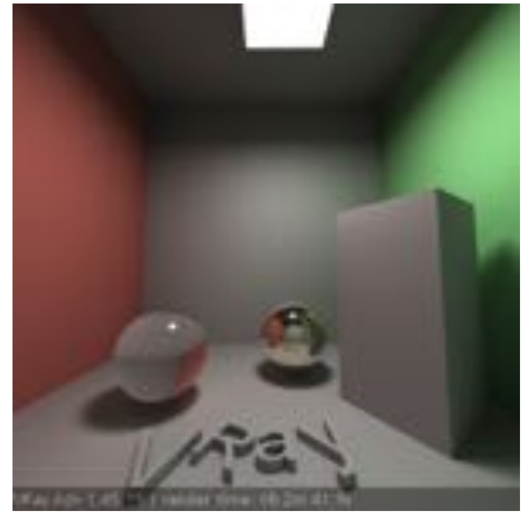




0 bounces

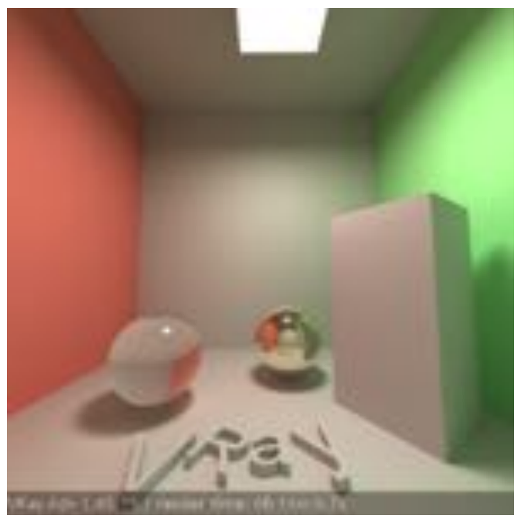


1 bounce



2 bounces

4 bounces



8 bounces

infinite bounces

The Problem of Computer Graphics

- Is not CG soon a solved problem?
- Will not computers soon be fast enough?

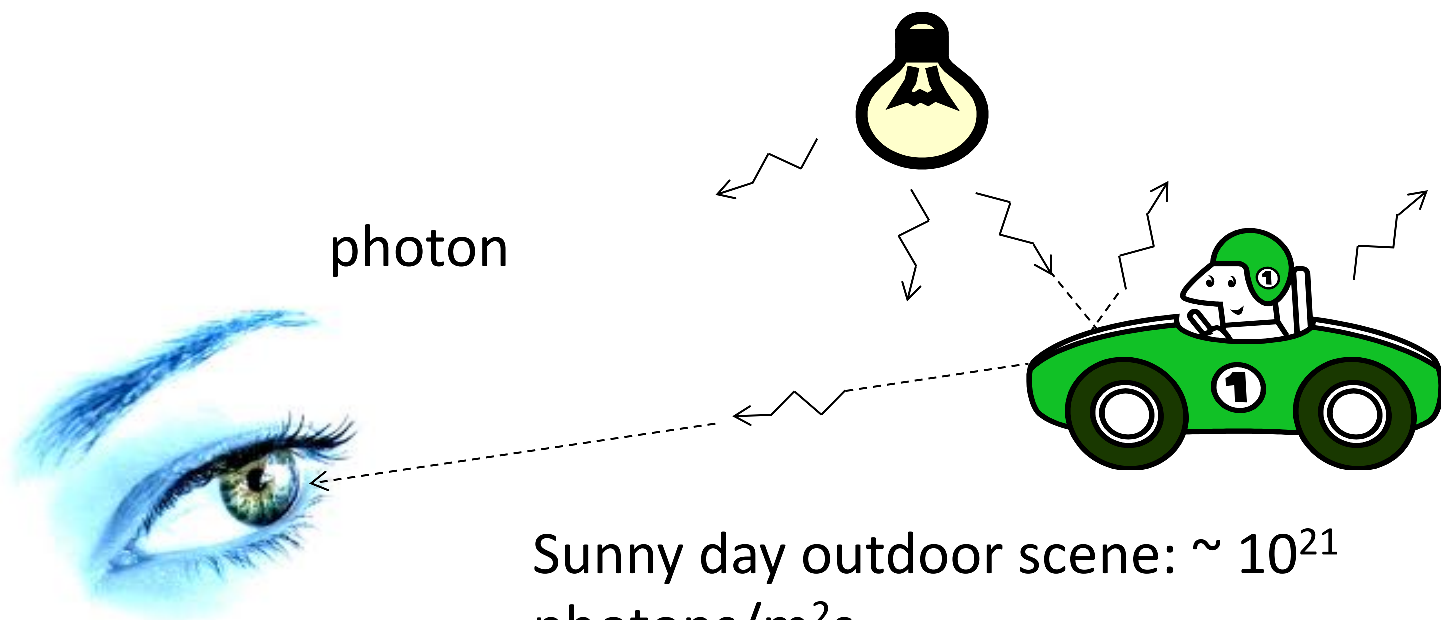
Probably not…

~20 to ~10^{15} photons/s

The eye has a resolution of 130M receptors:
120M gray scale (rods / stavar)
7M color (cones / tappar)

The problem of Computer Graphics

- Eye sensitivity: ~20 photons/s to ~ 10^{15} photons/s
- So, if we could trace only the photons that hit the eye, the problem would be limited.
- But, the only really guaranteed 100% correct way (still) is tracing photons from light to eye.



Sunny day outdoor scene: ~ 10^{21} photons/m^2s

Facts:

- Eye sensitivity: ~20 to ~10^{15} photons/s
- Sensitivity is logarithmic
  - i.e., difference between 100 or 200 photons is as noticable as for 10^{10} or $2*10^{10}$ photons
- ~ 10^{21} photons/m^2s
- 1 photon costs ~10.000 cycles

10 billion years per square meter for 1 computer

Solutions

For games: Smart specialized algorithms. And cheat, cheat, cheat…. as long as it is not too noticable

For movies: we typically trace ~100M – 10B photon bounces (and also cheat).

Typically: 100M – 1B photon bounces

So, what is the state-of-the-art for real-time graphics today?

Beäst + Unreal Engine

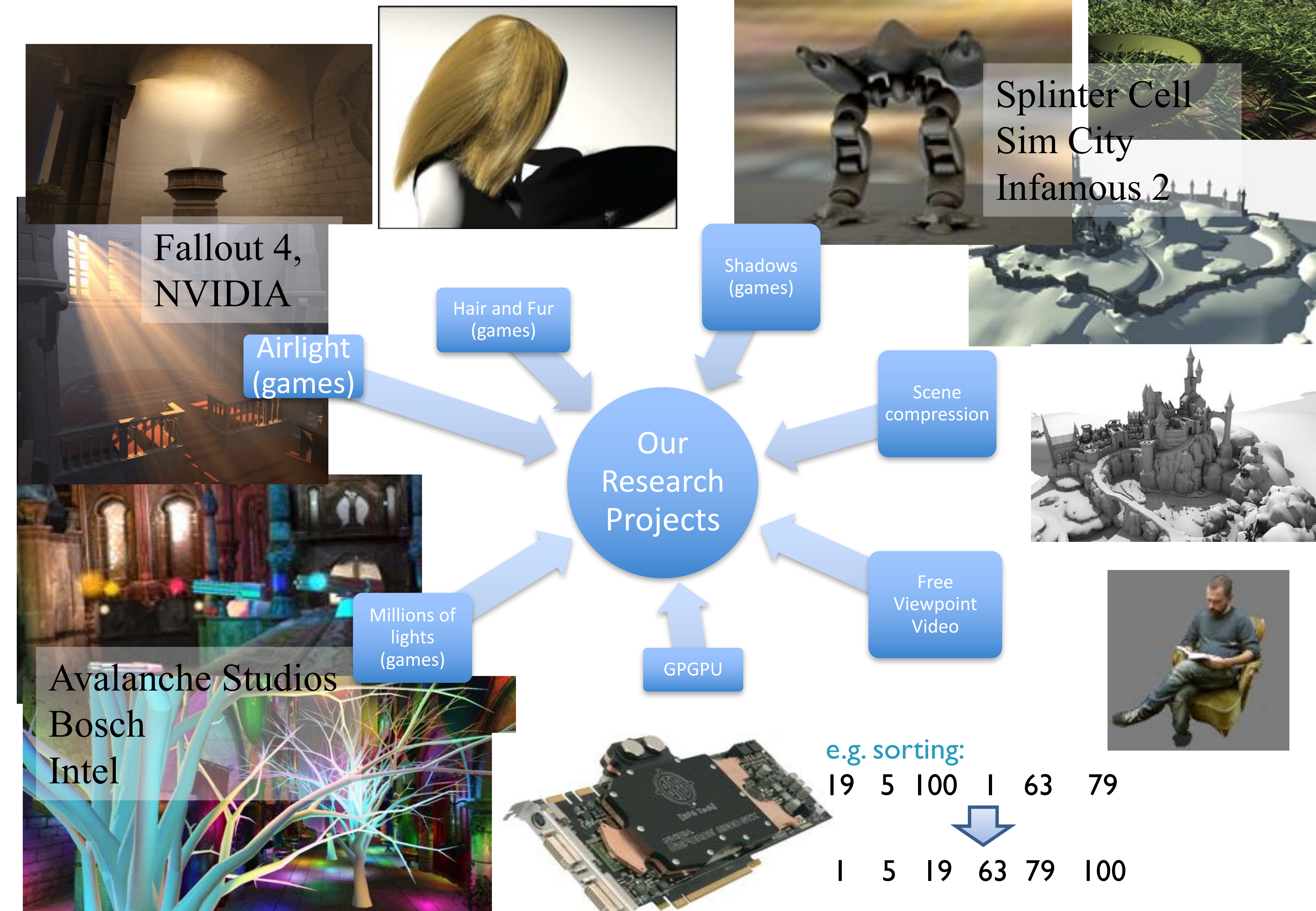
beäst™

Illuminate Labs

Illuminate Labs produkt “Beäst” för realistisk belystning i spel

Beast used in e.g.:

WET (A2M)
Mirror's Edge (EA Digital Illusions Creative Entertainment)
Mortal Kombat (Midway)
EVE Online (CCP Games)
CrimeCraft (Vogster Entertainment LLC)
Alpha Protocol (Obsidian Entertainment Inc)
Dragon Age: Origins (BioWare)
God of War III [8] (Sony Computer Entertainment)
Gran Turismo (Polyphony Digital)
and also the Unreal Engine



Fallout 4,
NVIDIA
Airlight (games)
Hair and Fur (games)
Shadows (games)
Splinter Cell
Sim City
Infamous 2
Scene compression
Our Research Projects
Free Viewpoint Video
GPGPU
Millions of lights (games)
Avalanche Studios
Bosch
Intel
e.g. sorting:
19 5 100 1 63 79
1 5 19 63 79 100

Beast + Unreal Engine

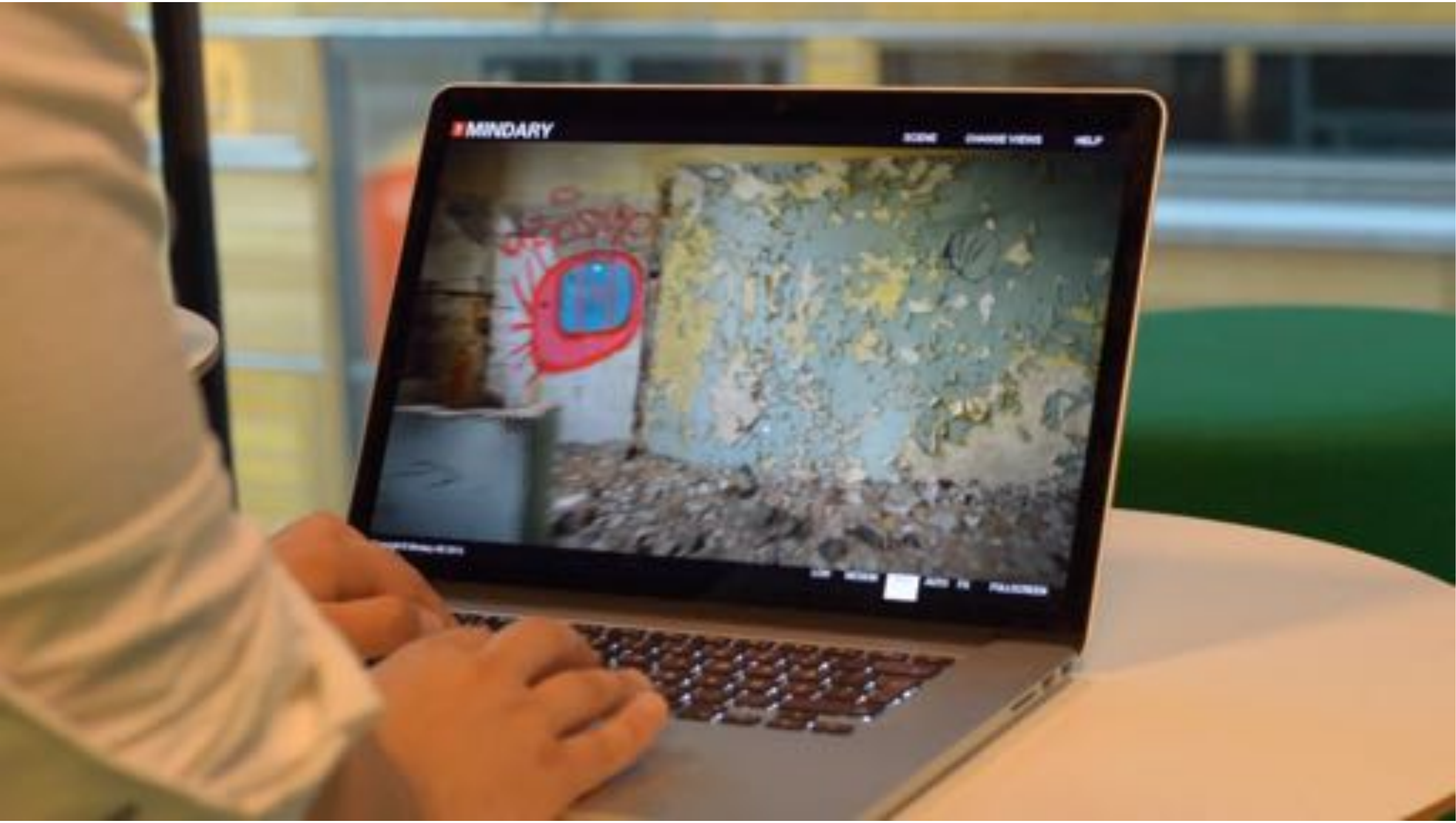
Mix of graphics and photographing

Textures from photographs

Mix of graphics and photographing

- Istället för att modellera 3D-grafik och beräkna realistiskt utseende:
  - Fotografera/filma verkligheten och konvertera den till 3D-grafik.

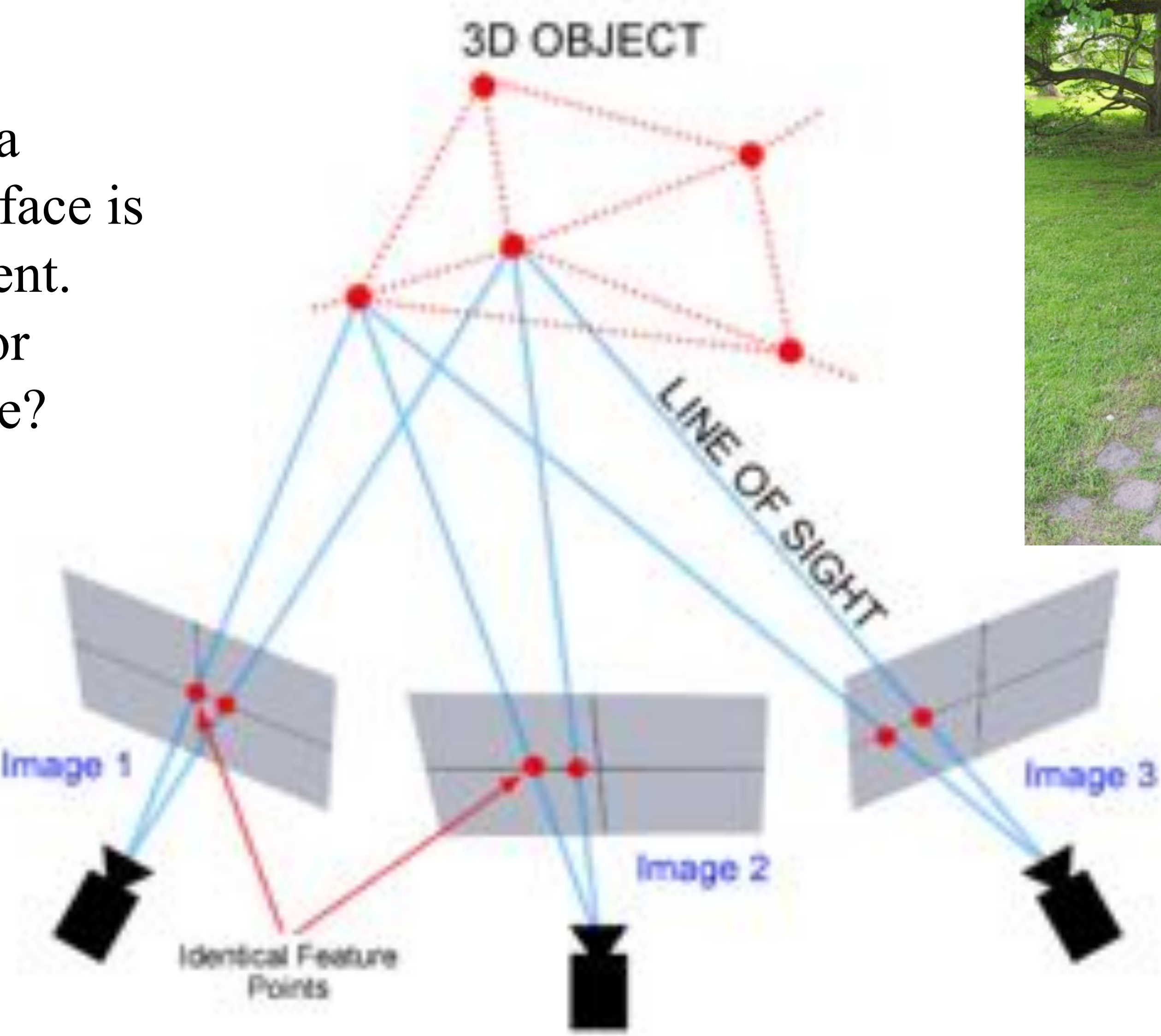
- Ofta vill man mixa modellerade och verkliga konverterade objekt.

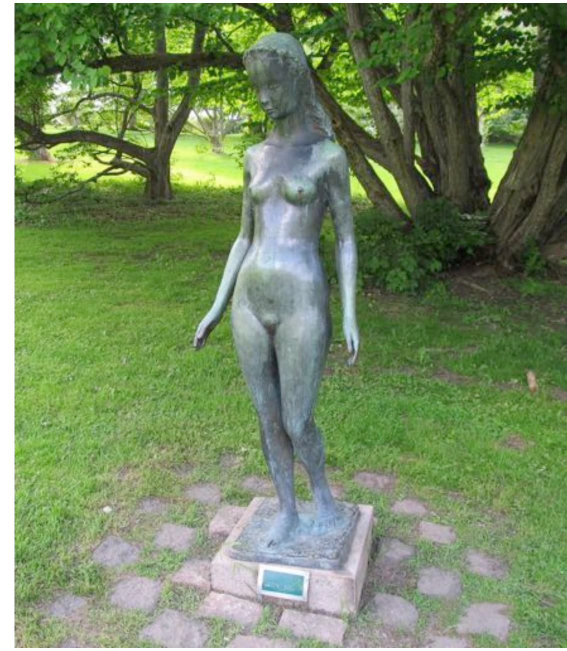


http://www.cse.chalmers.se/~uffe/mindary/demo/v2.html

Reflective surfaces (=view-dependent colors) are a problem

The color of a reflective surface is view dependent. So, what color should we use?





1. View-dependent colors. (here also different exposure times for the two cameras).

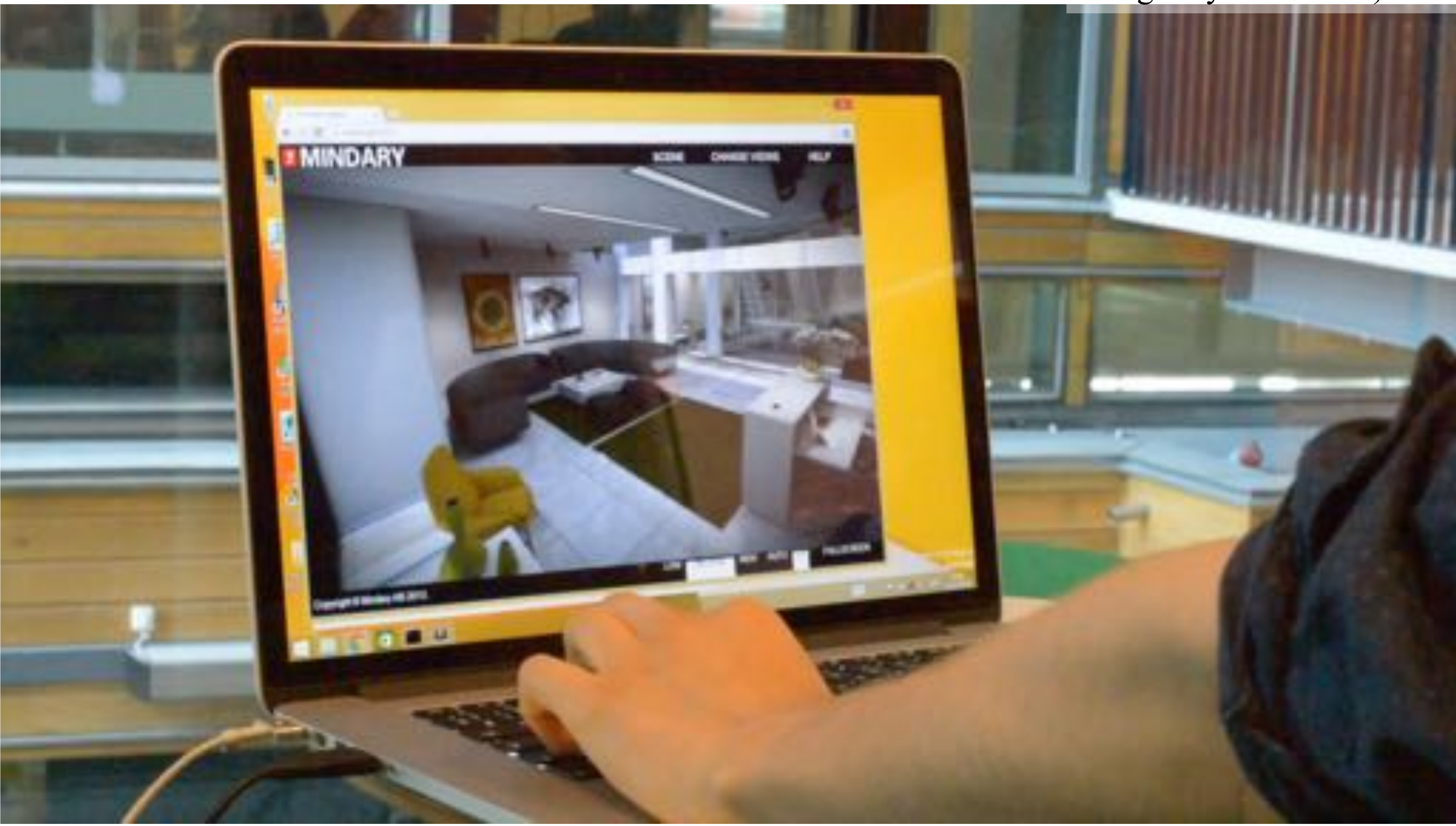
The more reflective surface, the larger the problem.

“Solution”:
- Supress reflections when photographing.
- Computer-generate the reflections when visualizing the 3D scene.

Very reflective surfaces (e.g. mirror) does not work.

MINDARY
TECHNOLOGY AND VISUALISATION

The most important 1st light bounces (i.e., sharp and glossy reflections)



Combinig photo textures and computer-generated view-dependent reflections

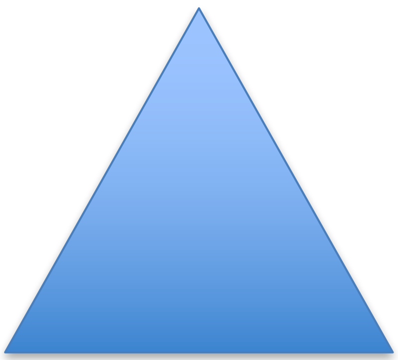
Unreal Engine 4

The most important 1st light bounces (i.e., sharp and glossy reflections)

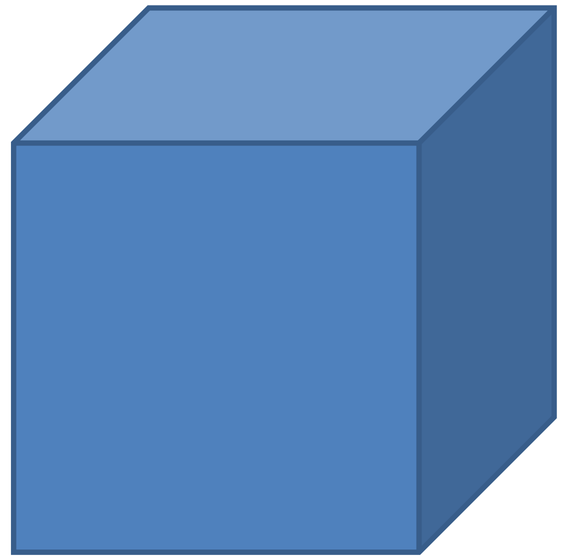
Increasing the amount of geometric details

Triangles

Voxels



Triangle
36 bytes



Voxel
Volume – element
1 bit

Voxels

- Desirable to be able to use very high resolutions

Voxels

Why use voxels?

Autodesk fluid simulation

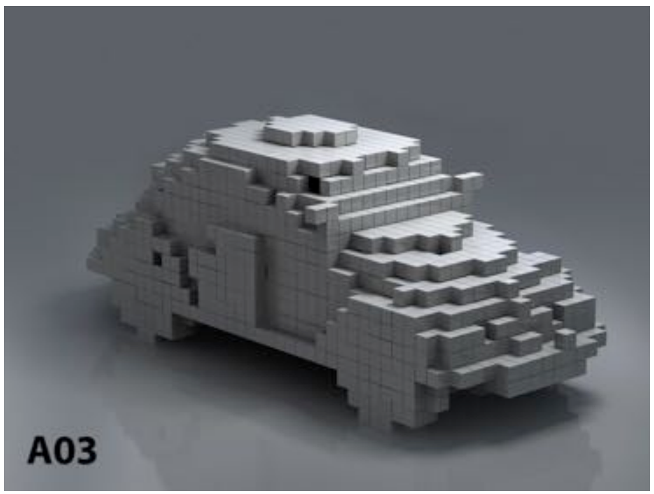
Voxels

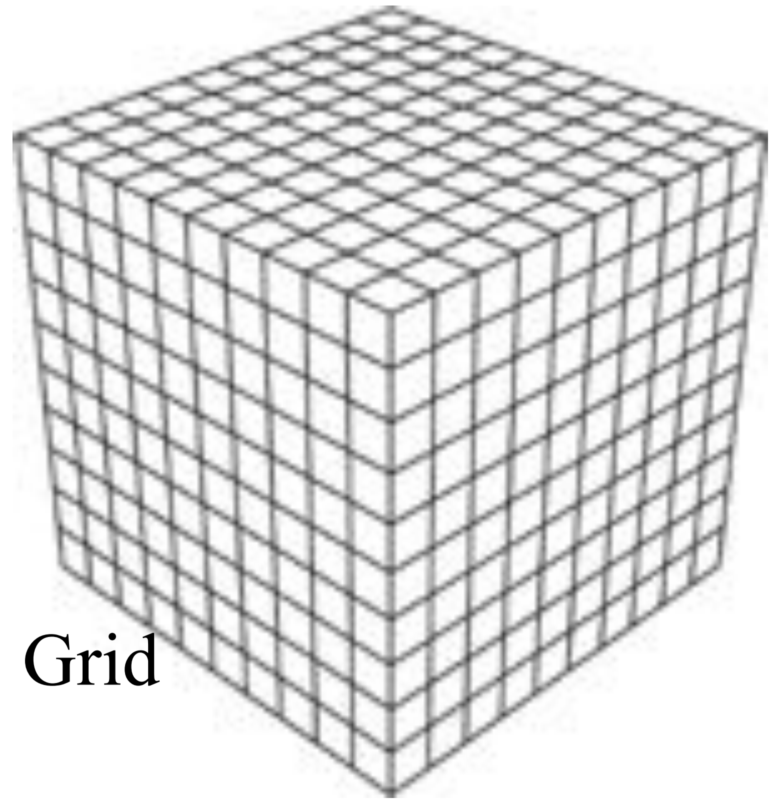
RealFlow by nextLimit

Voxels

One possible data structure:

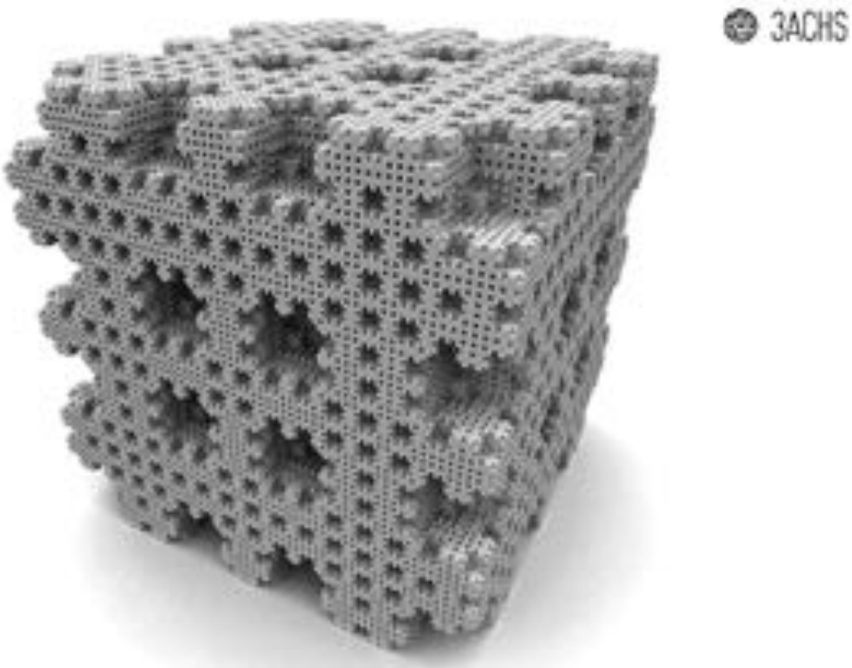
- Grids – 3D array of 0:s and 1:s

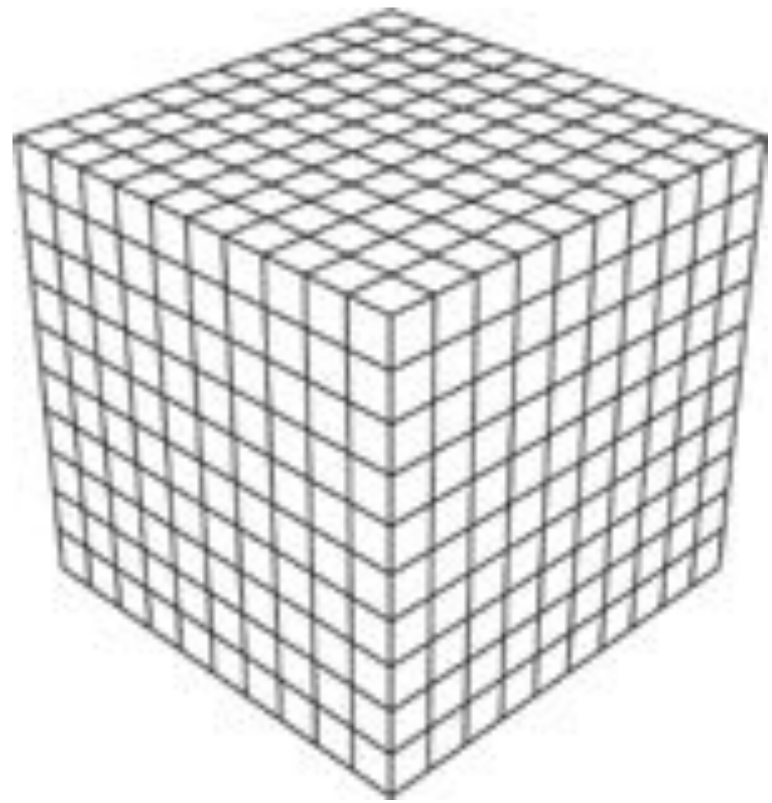




Voxels

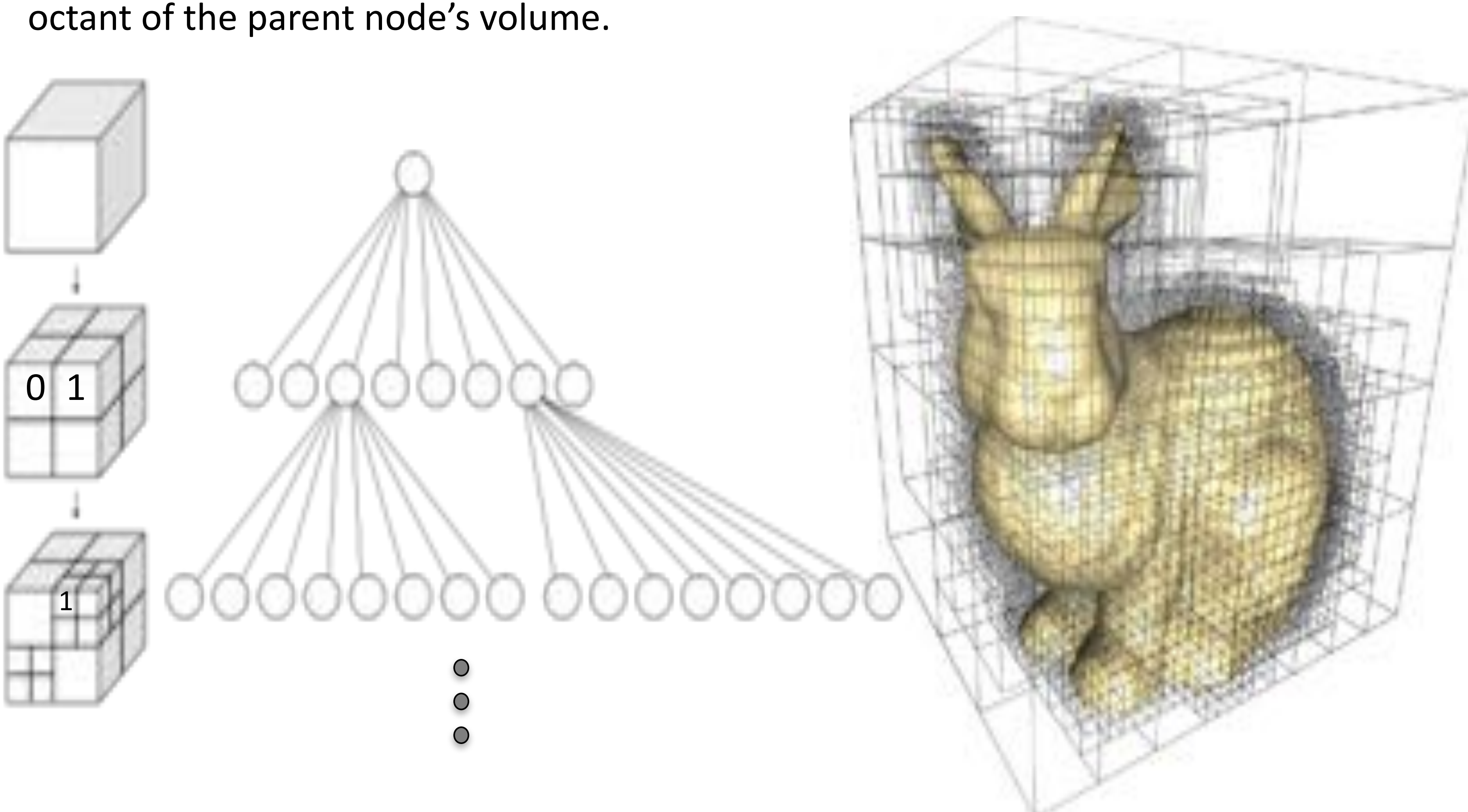
One possible data structure:





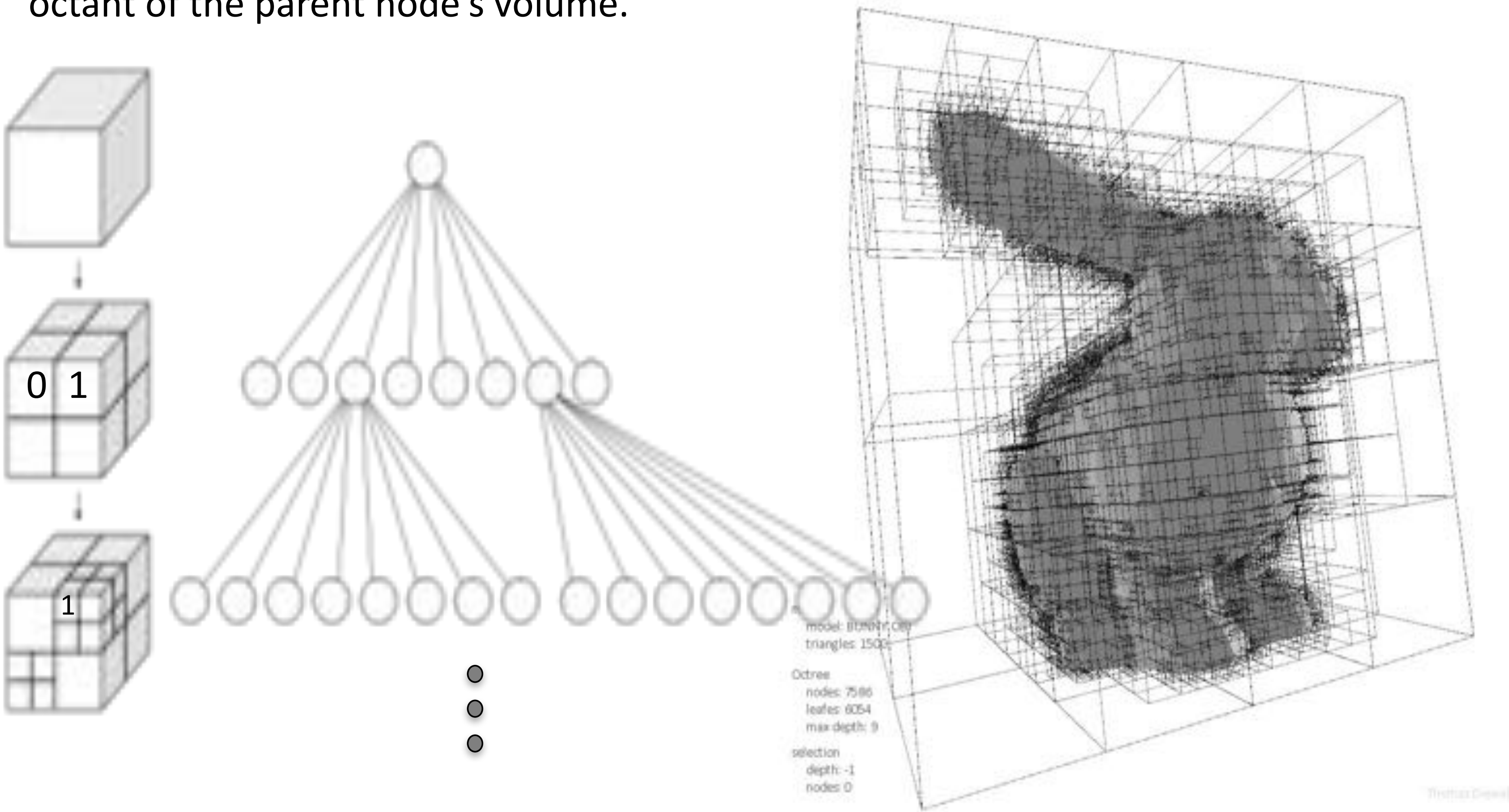
Sparse Voxel Octree

Each node has eight children, representing an octant of the parent node's volume.



Sparse Voxel Octree

Each node has eight children, representing an octant of the parent node's volume.

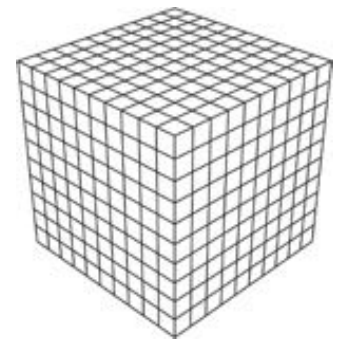


Sparse Voxel Octree

- SVO: Id Software, rage 6
- 1.15 bits/ non-empty voxel
- We: 0.08 bit/non-empty voxel

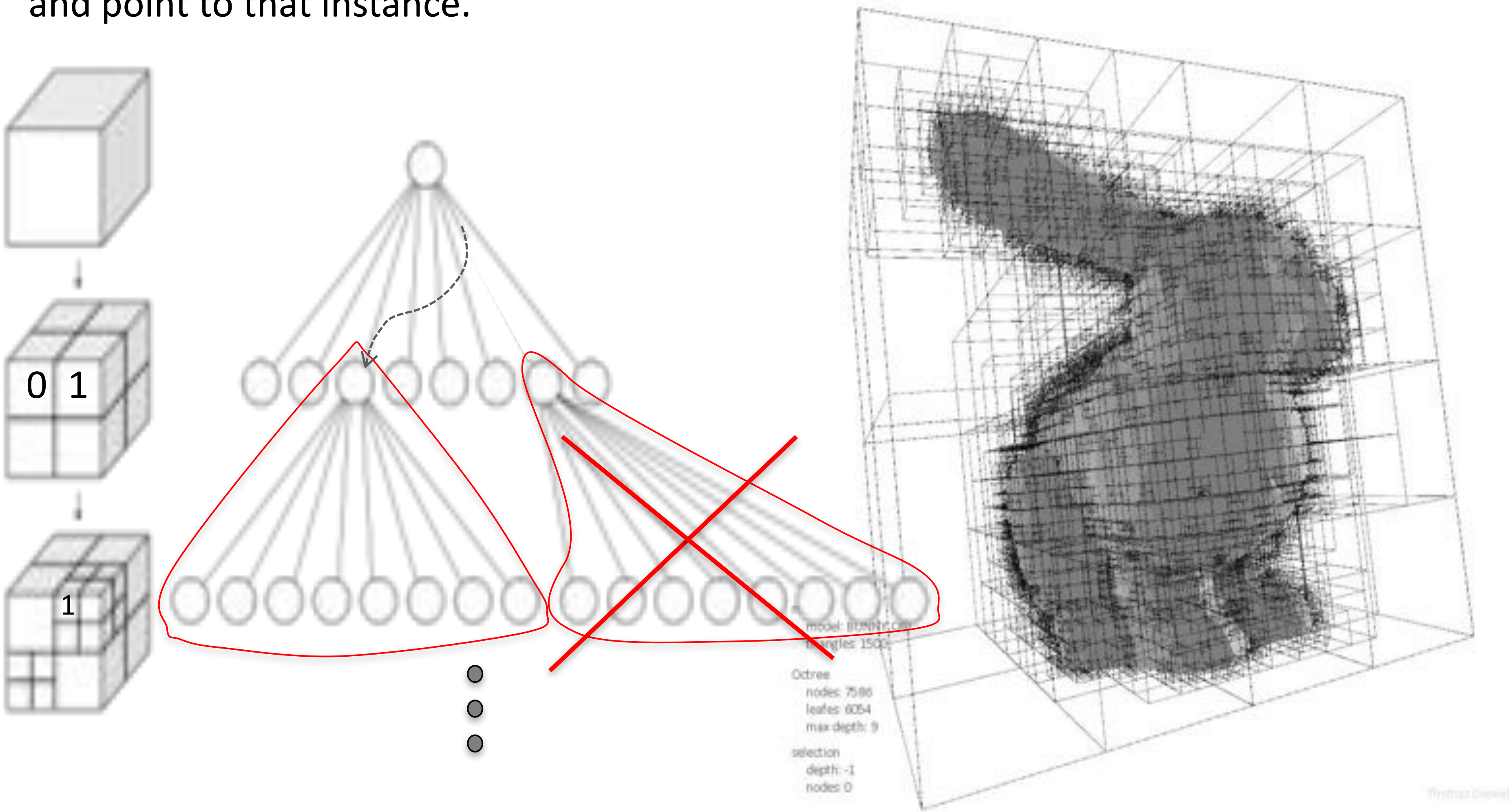
Voxels

- Voxel = 1 bit.
- We currently handle scene of res = 128.000^3
  - Naively: 262 TB
  - We => < 1GB

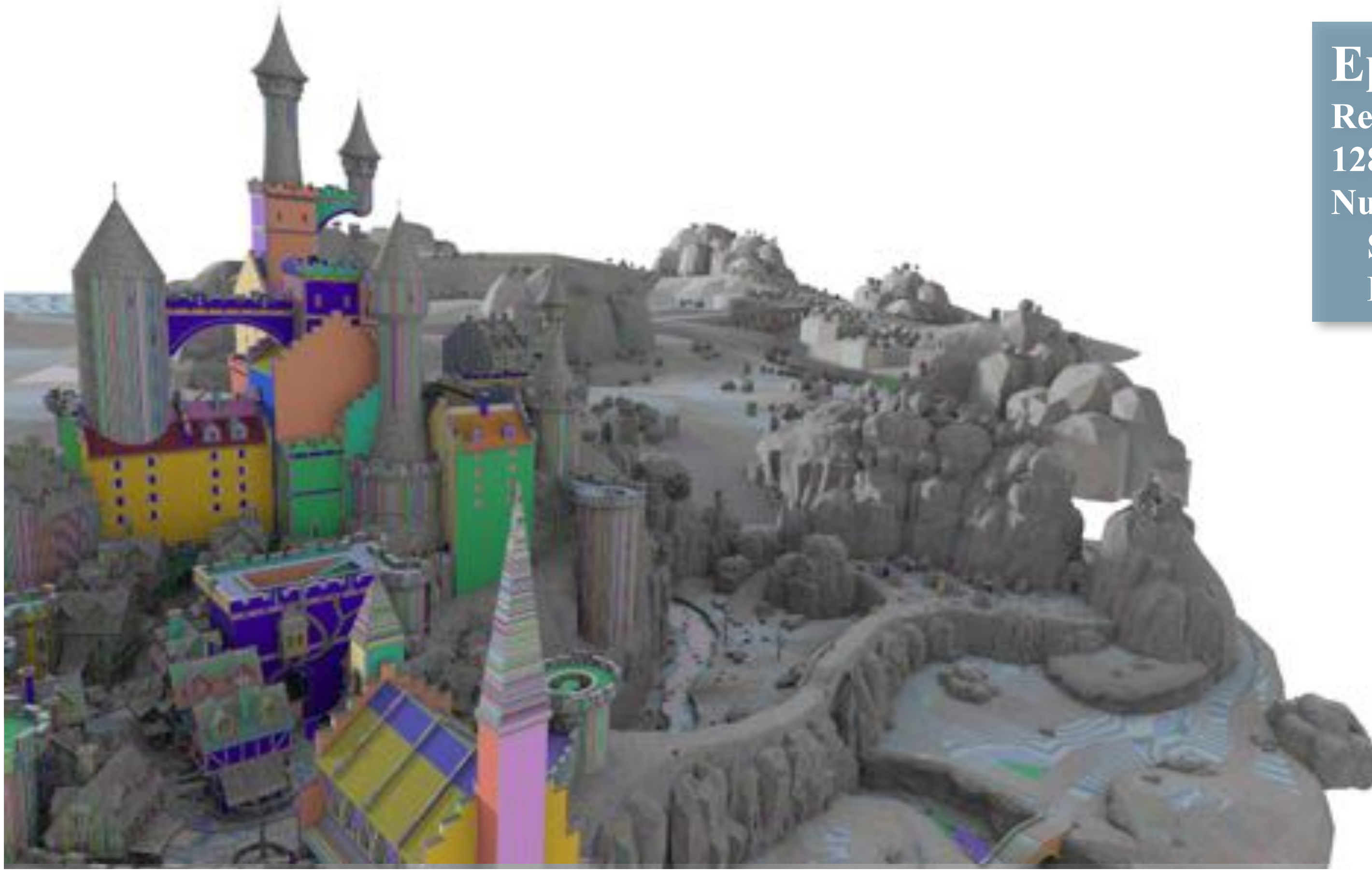


Our Voxel DAGs

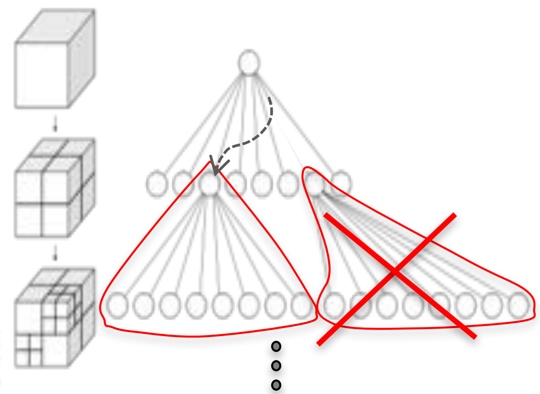
For identical subgraphs, only store one instance, and point to that instance.



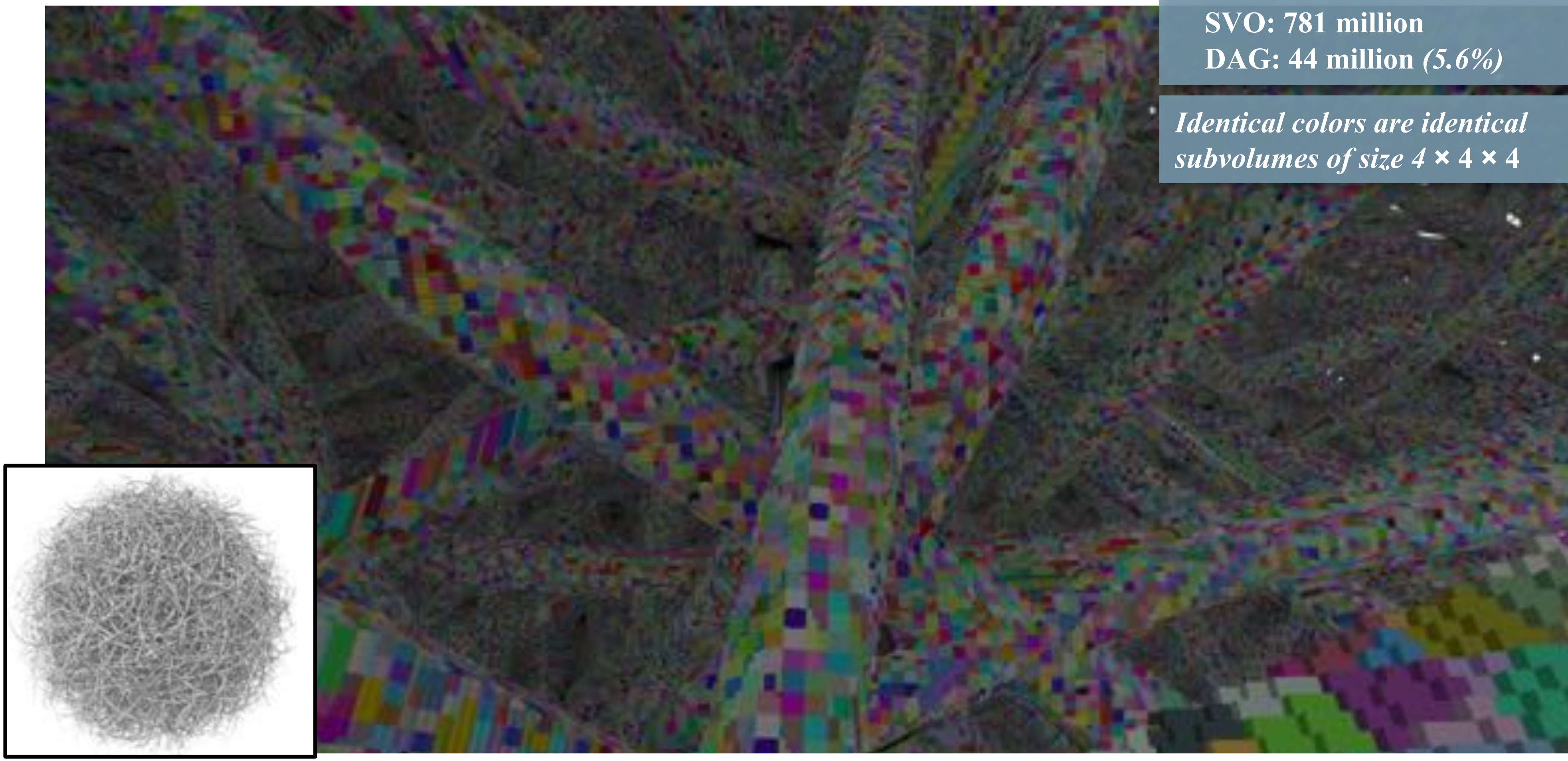
Visualizing Identical Subtrees



Epic Citadel
Resolution: 128K × 128K × 128K
Number of nodes
- **SVO: 5.5 billion**
- **DAG: 45 million *(0.8%)***



Visualizing Identical Subtrees



Hairball

Resolution: 8K × 8K × 8K

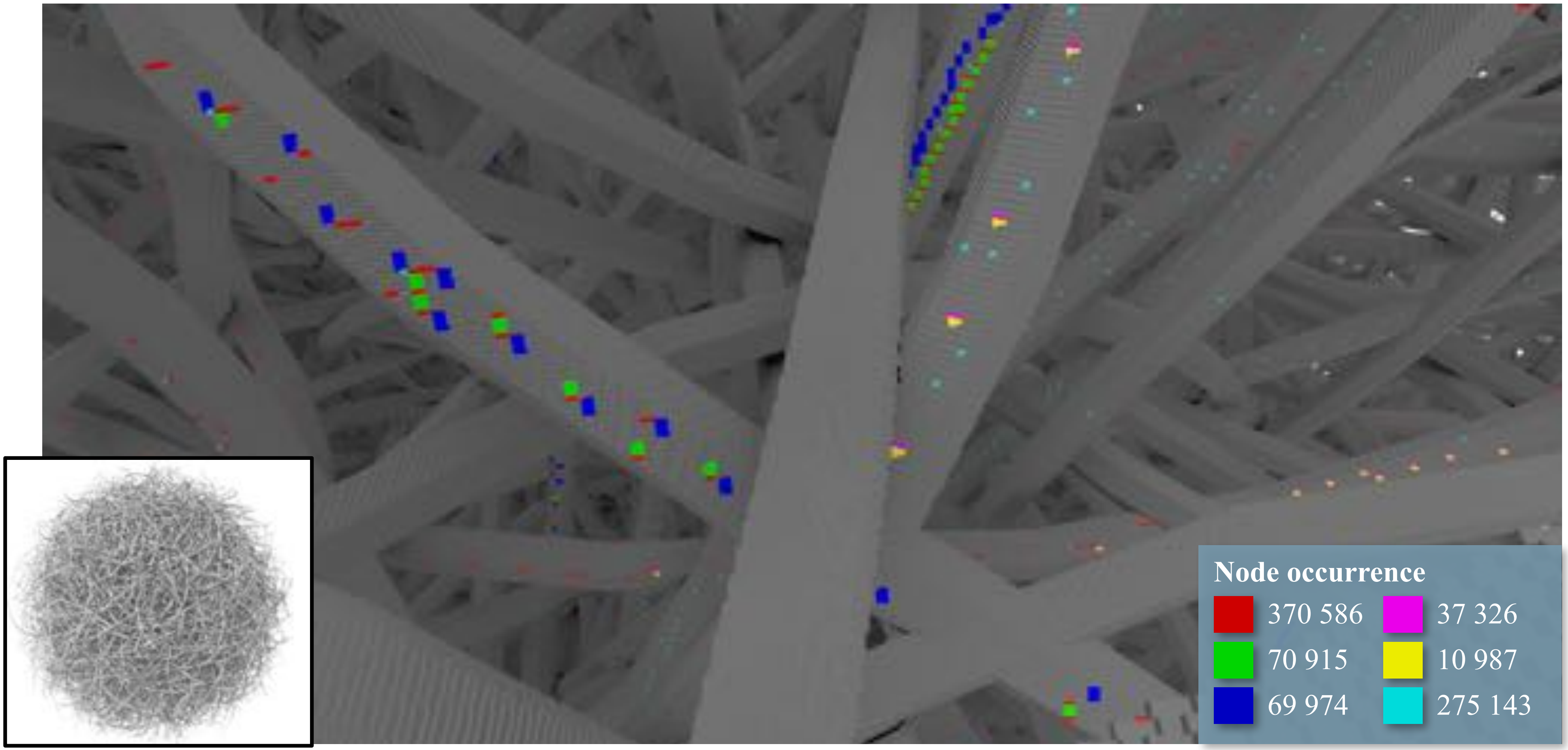
Number of nodes

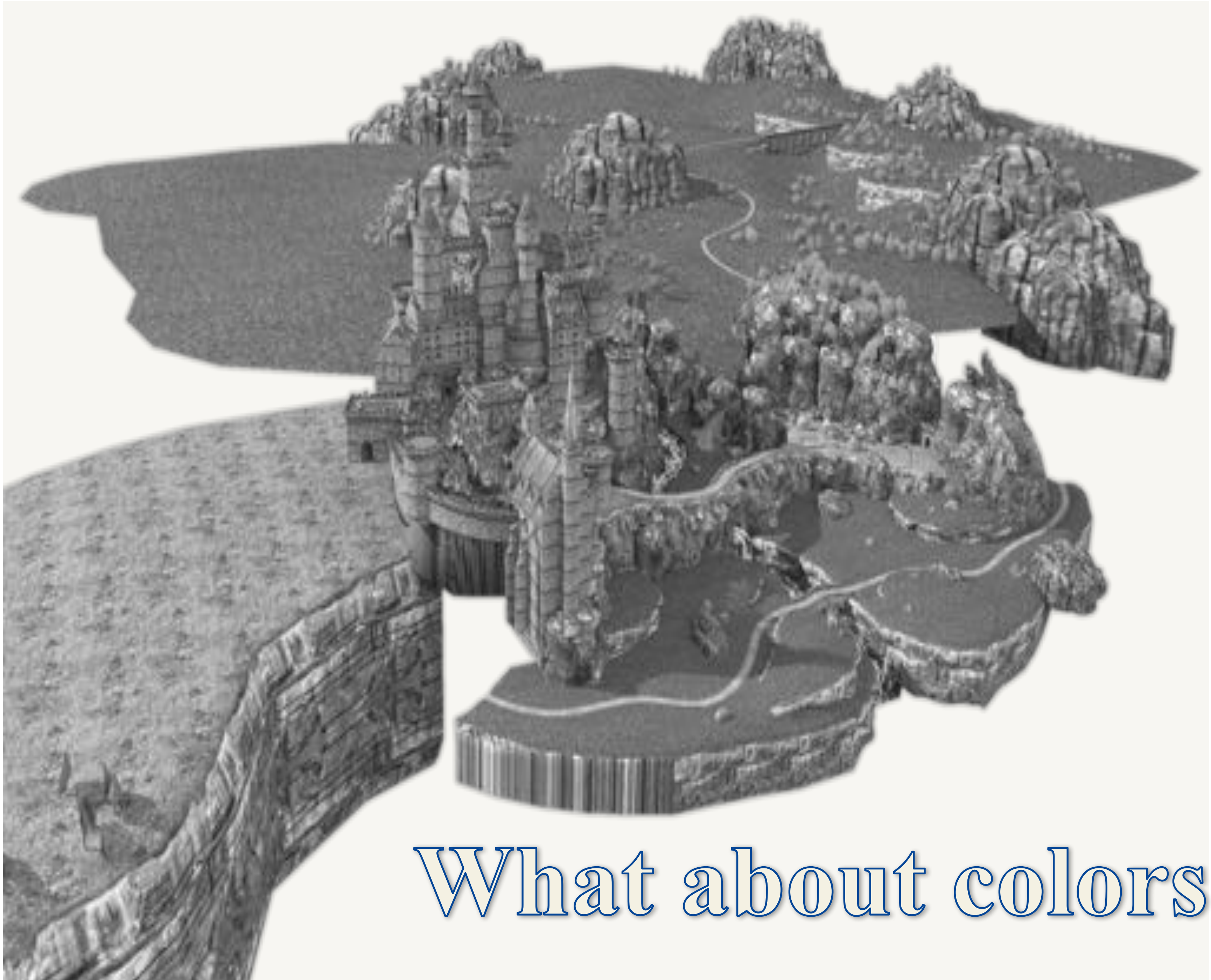
SVO: 781 million

DAG: 44 million *(5.6%)*

***Identical colors are identical subvolumes of size 4* × 4 × 4**

Visualizing Identical Subtrees



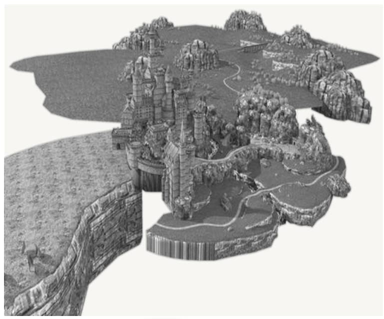


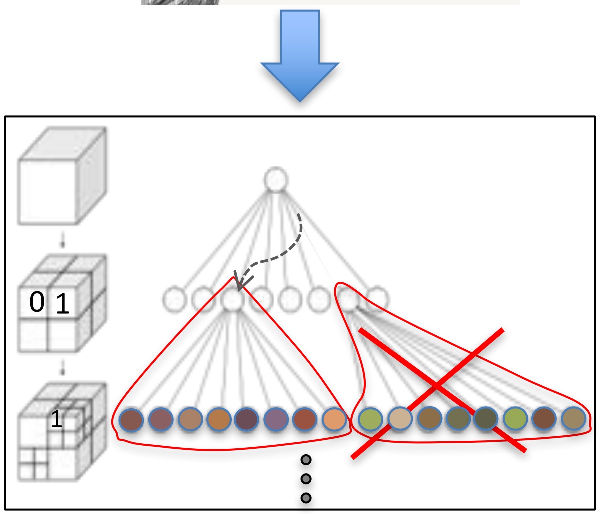
What about colors?

What about colors?

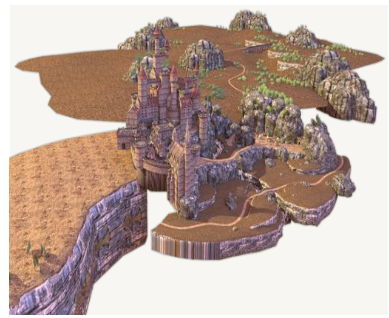
Compress geometry and colors separately with different specialized methods.

Geometry:





Voxel colors:



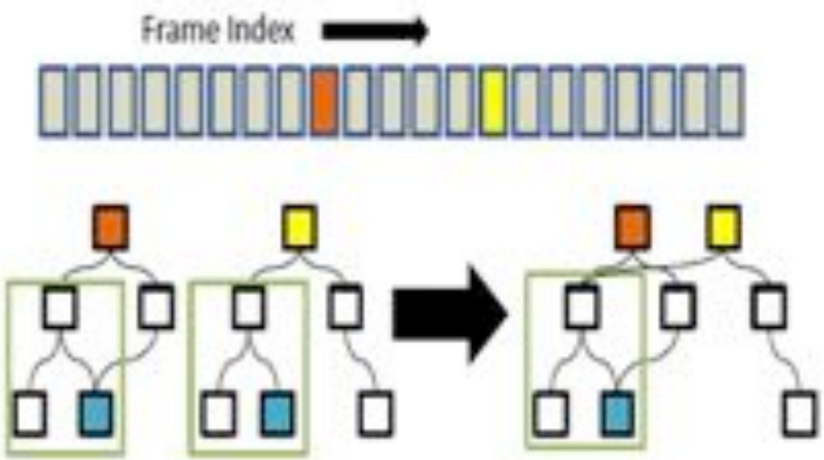
Three problems:

1. How can we compress the colors efficiently?
2. Connection between voxels and their colors
3. Fast color lookups

From static scenes
to dynamic (moving) scenes
to Free Viewpoint Video

Voxel DAGs – dynamic scenes
3D geometry + time dimension

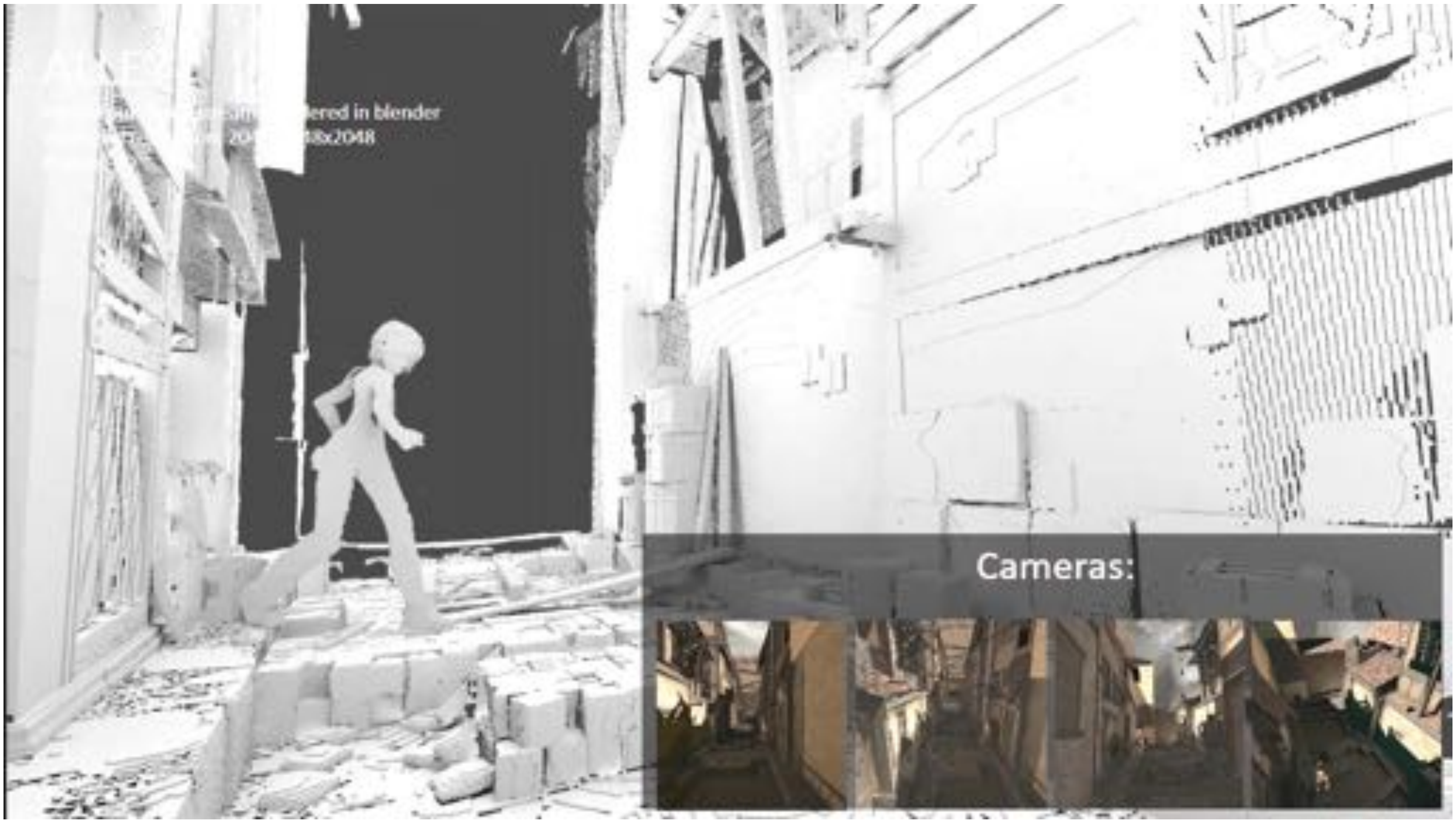
For every time step (=frame) in a dynamic scene, convert the whole voxelized 3D scene to a DAG:



70 frames
2048^3 grid
Length: 2.9 sec
@24Hz

1.9 MiB
(=2.3 GiB/hour)
5.2 Mbits/s

Voxel DAGs – dynamic scenes
3D geometry + time dimension



70 frames
2048^3 grid
Length: 2.9 sec
@24Hz

1.9 MiB
(=2.3 GiB/hour)
5.2 Mbits/s

Free Viewpoint Video

Want:

1. convert a real scene to 3D graphics, 24 times/second.
2. Render scene from any viewpoint.

Med 2 eller fler kameror kan man beräkna djup – precis som våra ögon.

Med 2 eller fler kameror kan man beräkna djup – precis som våra ögon.

Med 2 eller fler kameror kan man beräkna djup – precis som våra ögon.

Med 2 eller fler kameror kan man beräkna djup – precis som våra ögon.

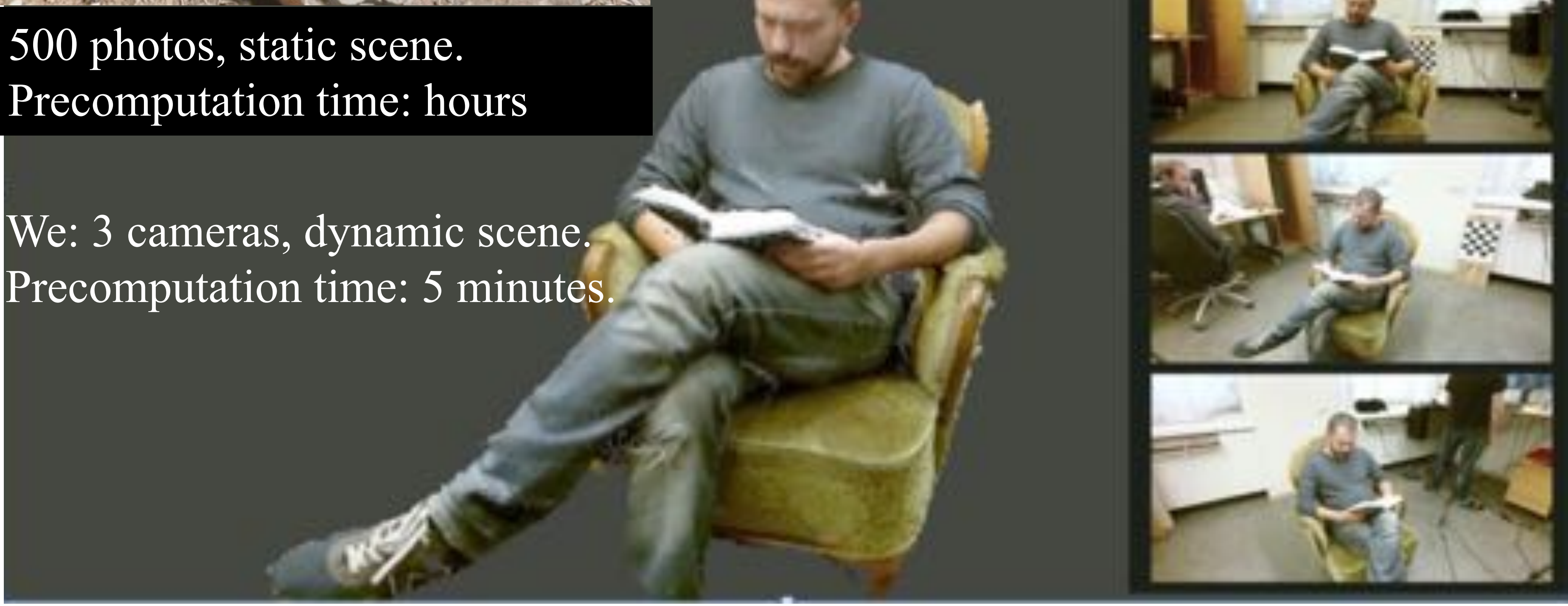
Varje kub ~1cm^3. Önskar ~1mm^3

Free Viewpoint Video

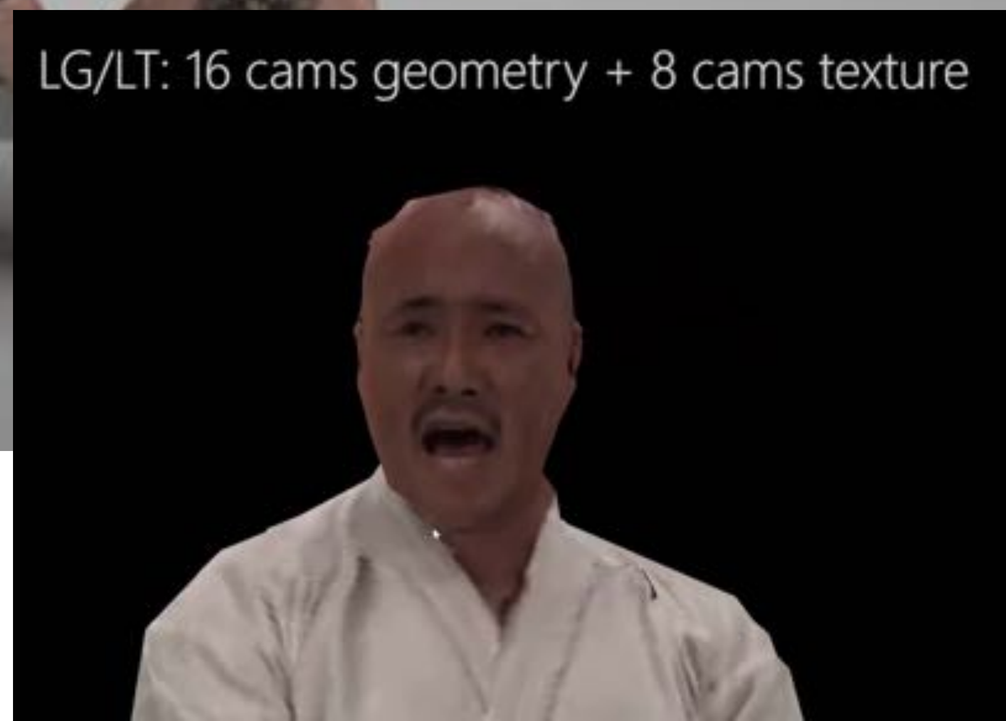
480 frames
512^3 grid
20 sec
@24Hz

5.2 MiB
0.9
GiB/hour
2.1 Mbits/s

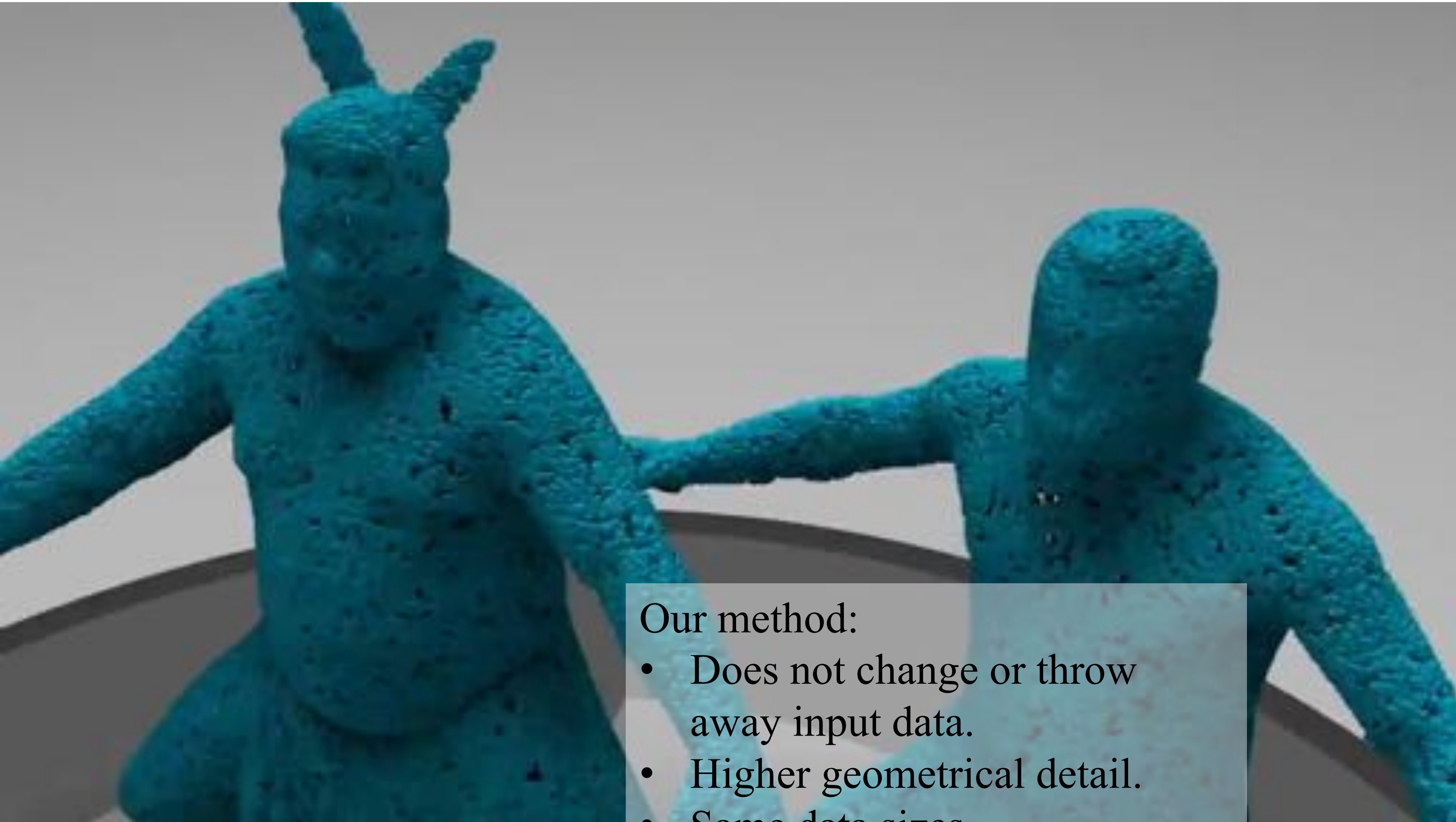
500 photos, static scene.
Precomputation time: hours



We: 3 cameras, dynamic scene.
Precomputation time: 5 minutes.



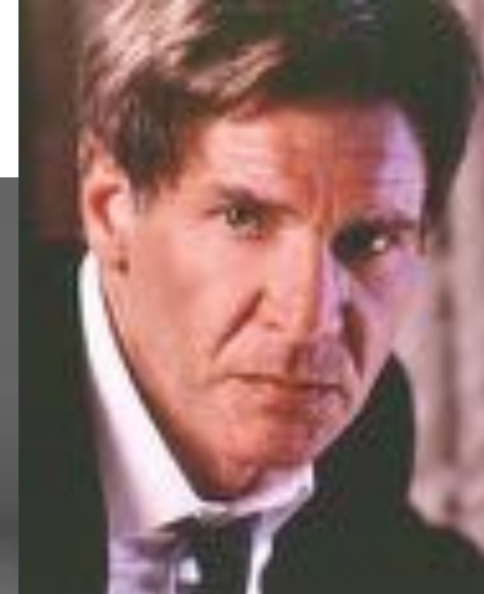
Microsoft: ~100 cameras, triangles
Precomputation time: ~100 hours.



Our method:

- Does not change or throw away input data.
- Higher geometrical detail.
- Same data sizes.
- Faster computation times
- Much simpler and also robust

But we still have no view-dependent colors.
I.e., no reflections that change with the view position



Deadpool (to demonstrate view-dependent reflections)

Deadpool (to demonstrate desired quality of FVV with view-dependent reflections)

Framtidens mediatekniker

- Filma (4-10 kameror)
- 3D-rekonstruera varje frame
  - För varje tidssteg i filmen:
    - 3D rekonstruera scenen från kamerornas foton.
- Komprimera från TB till GB.
  - streambar över internet
- Spela upp filmen från valfri synvinkel
  - Dvs vi kan gå omkring i filmen medan den spelar.

Framtidens mediatekniker

- Science Fiction visar vägen
  - Visar vad vi vill ha
  - Människan skaffar det hon vill ha (bland annat…)

Star Trek

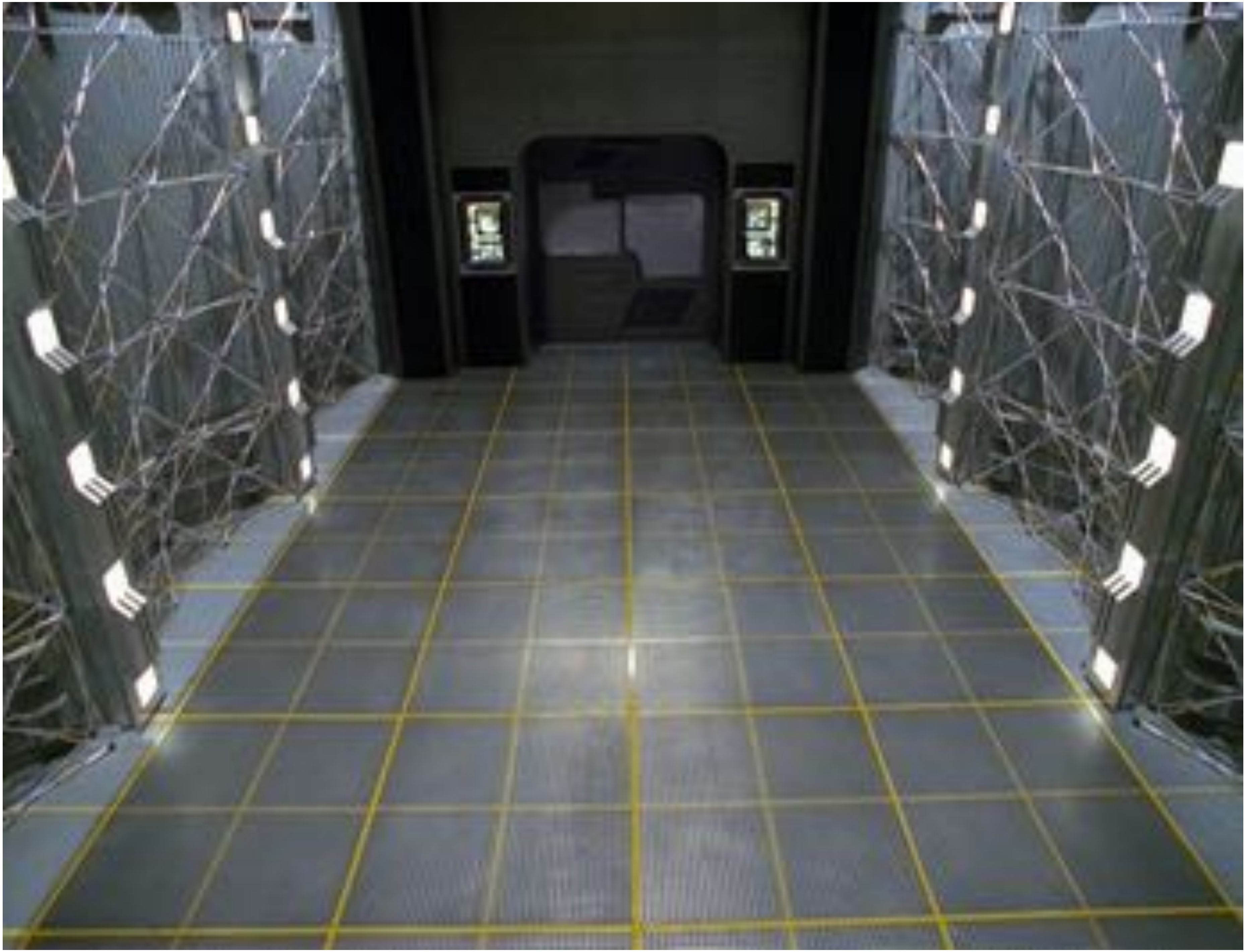
60:ies

2000

Star Trek - Tablets

80:ies 2010

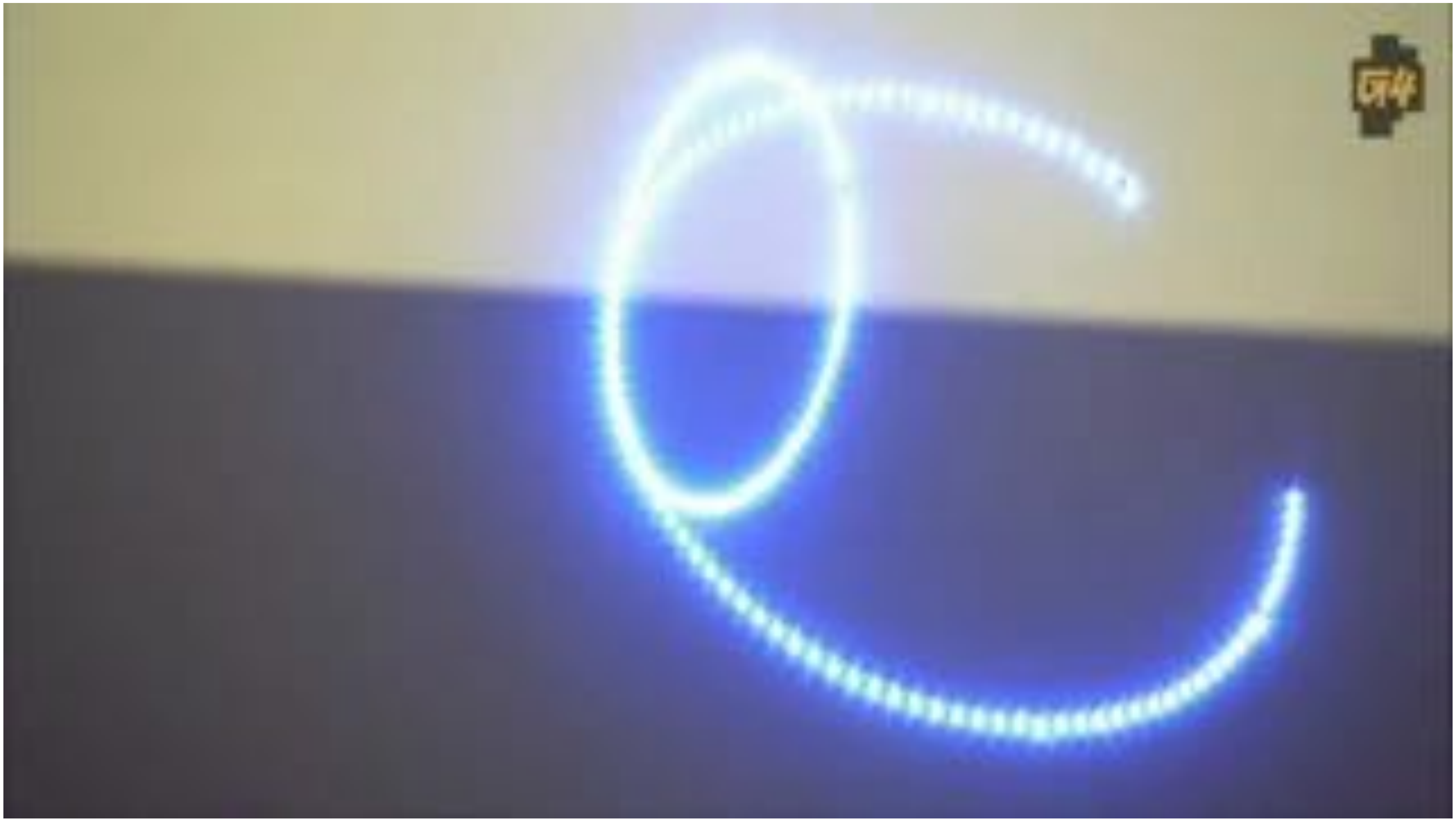
Star Trek - Holodeck



Oculus Rift

HTC Vive

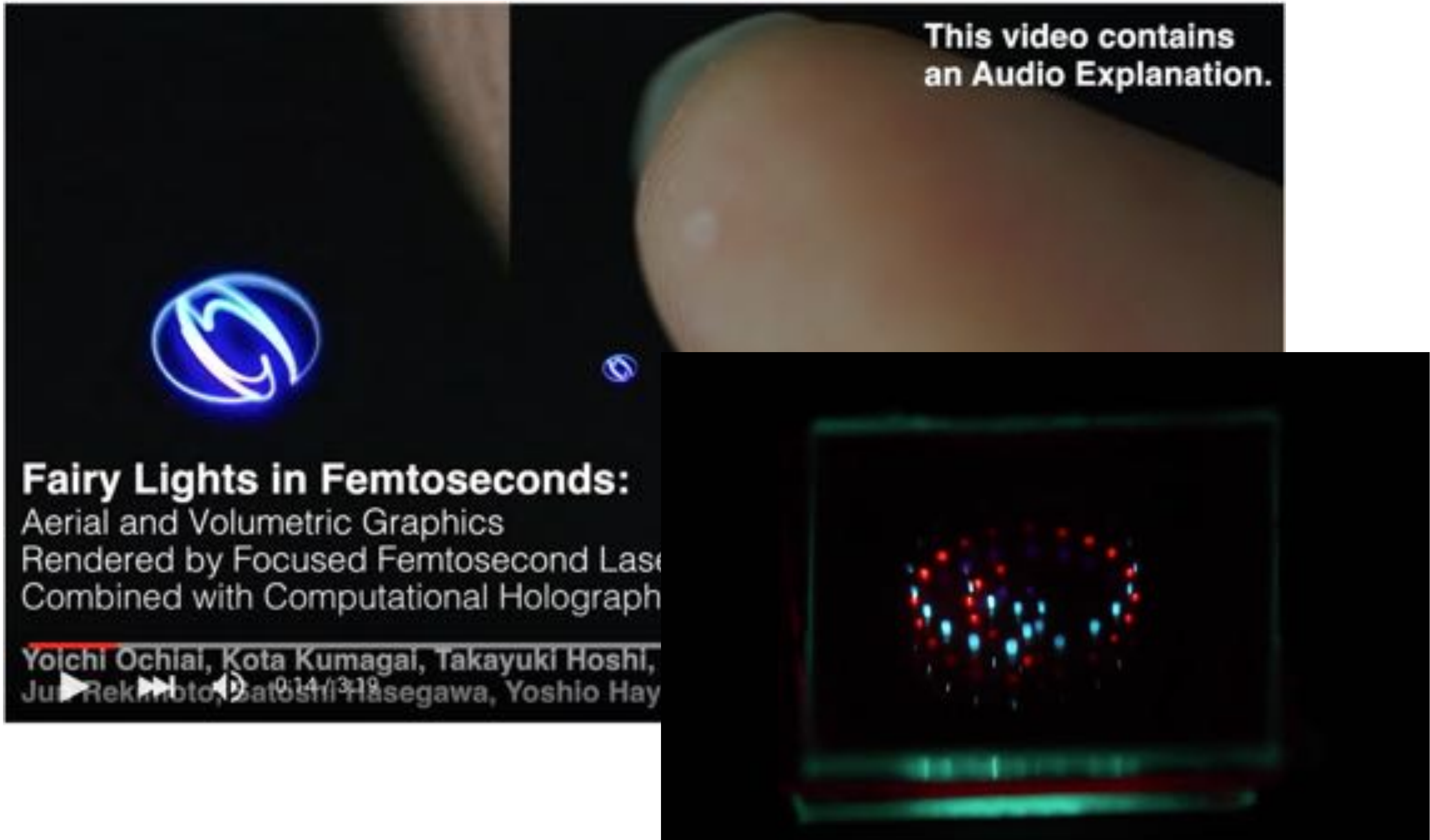
Mid air display



Mid air display

Mid air displays 2015

Mid air displays 2015



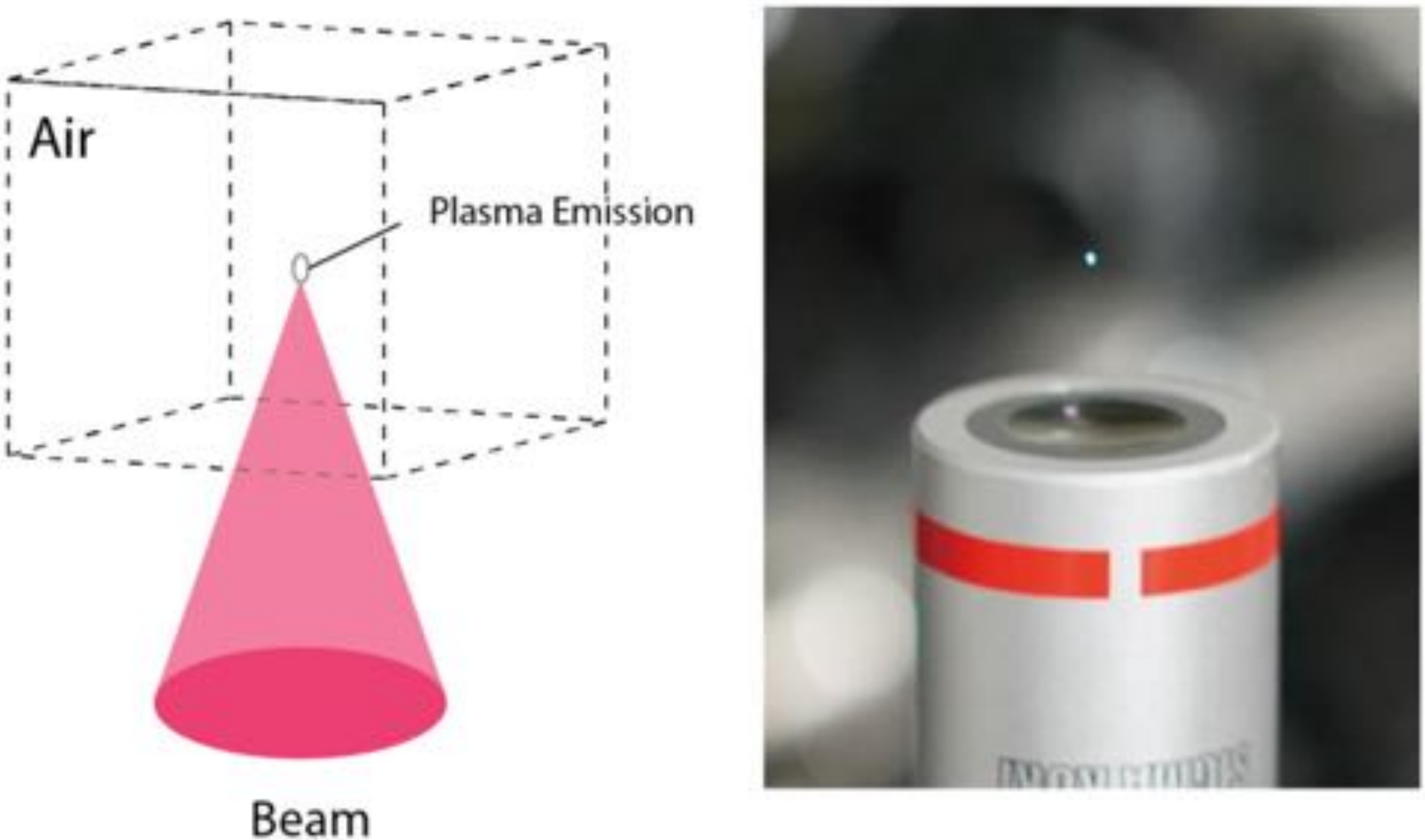


Figure 4: *Laser plasma induced by focused femtosecond laser.*

Rendering Volumetric Haptic Shapes in Mid-Air using Ultrasound

Inget nytt under solen

- Titta på Automan från 1983-1984.

Inget nytt under solen

- Titta på Automan från 1983-1984.

Inget nytt under solen

- T


AUTOMAN

The Future?

- Much science fiction will become possible
- We want to enter computer-generated virtual worlds
  - Holodeck (maybe in a few decades)
  - Or plug into brain like in Matrix…
- We want computer-generated objects to enter our real world
  - 3D printers
  - Mid-air displays
  - Virtual matter (particles)