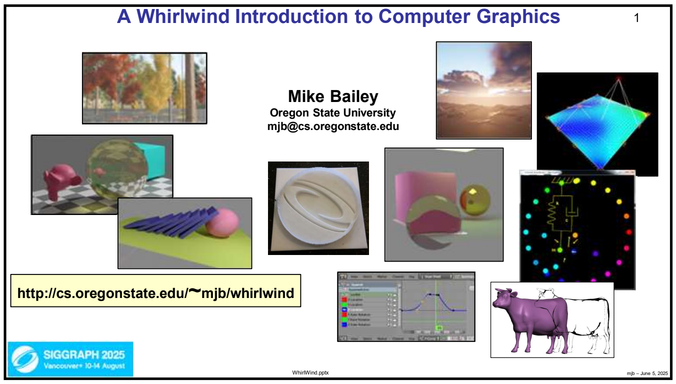
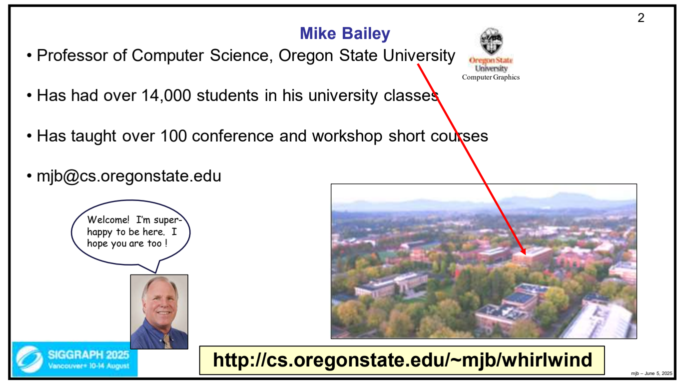

A Whirlwind Introduction to Computer Graphics
1
Mike Bailey
Oregon State University
mjb@cs.oregonstate.edu
http://cs.oregonstate.edu/~mjb/whirlwind
SIGGRAPH 2025
Vancouver+ 10-14 August

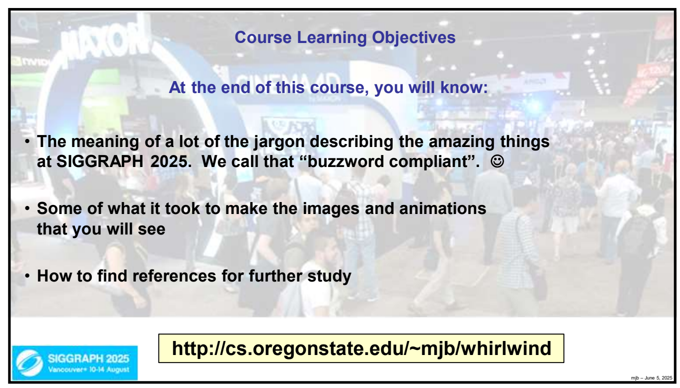
1


2
Mike Bailey
• Professor of Computer Science, Oregon State University
• Has had over 14,000 students in his university classes
• Has taught over 100 conference and workshop short courses
• mjb@cs.oregonstate.edu
Oregon State University Computer Graphics
Welcome! I'm super-happy to be here. I hope you are too !
http://cs.oregonstate.edu/~mjb/whirlwind
SIGGRAPH 2025
Vancouver+ 10-14 August

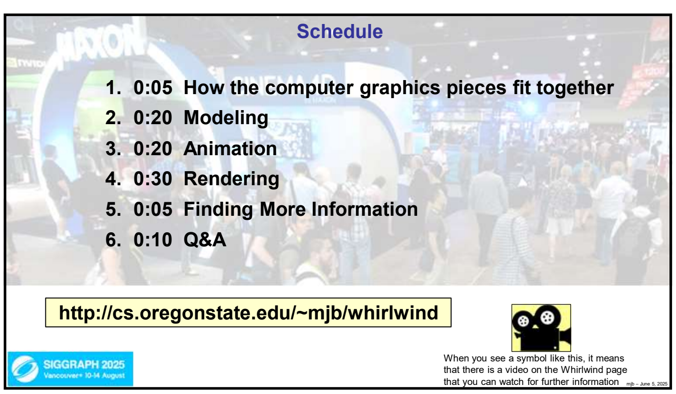
2


3
A Whirlwind Introduction to Computer Graphics
http://cs.oregonstate.edu/~mjb/whirlwind
SIGGRAPH 2025
Vancouver+ 10-14 August

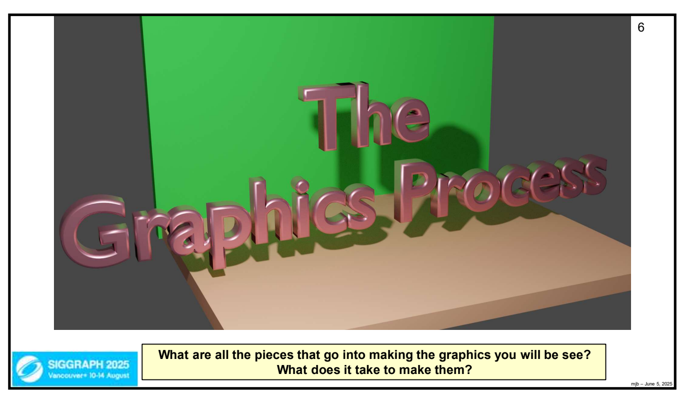
3


Course Learning Objectives
At the end of this course, you will know:
• The meaning of a lot of the jargon describing the amazing things at SIGGRAPH 2025. We call that "buzzword compliant". ☺
• Some of what it took to make the images and animations that you will see
• How to find references for further study
http://cs.oregonstate.edu/~mjb/whirlwind
SIGGRAPH 2025
Vancouver+ 10-14 August

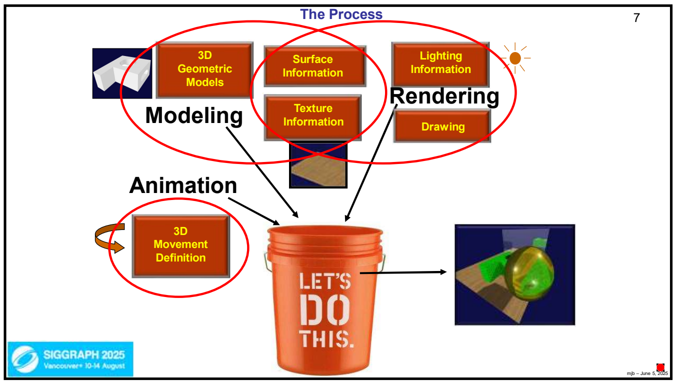
4


Schedule
1. 0:05 How the computer graphics pieces fit together
2. 0:20 Modeling
3. 0:20 Animation
4. 0:30 Rendering
5. 0:05 Finding More Information
6. 0:10 Q&A
http://cs.oregonstate.edu/~mjb/whirlwind
When you see a symbol like this, it means that there is a video on the Whirlwind page that you can watch for further information
SIGGRAPH 2025
Vancouver+ 10-14 August

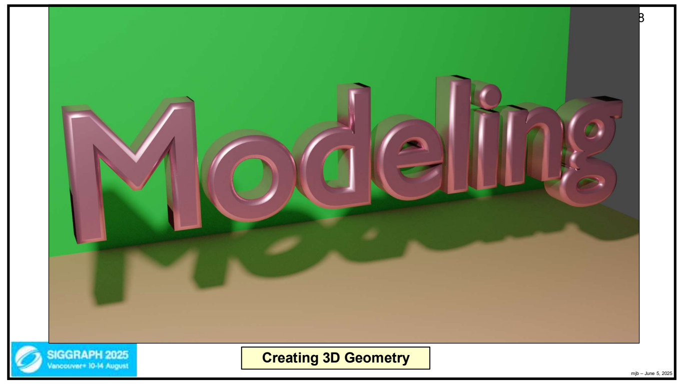
5


6
The Graphics Process
What are all the pieces that go into making the graphics you will be see?
What does it take to make them?
SIGGRAPH 2025
Vancouver+ 10-14 August

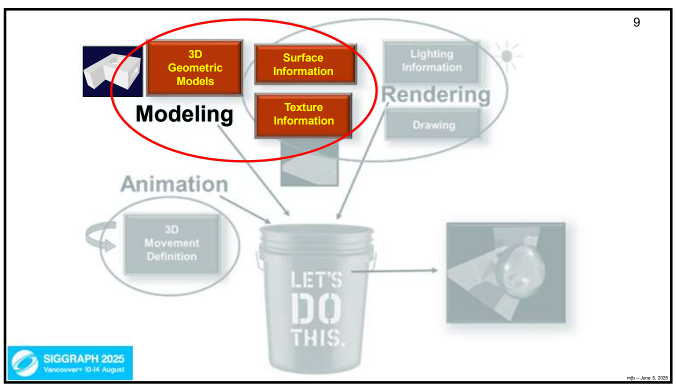
6


The Process
7
3D Geometric Models
Surface Information
Lighting Information
Texture Information
Drawing
Modeling
Rendering
Animation
3D Movement Definition
LET'S DO THIS.
SIGGRAPH 2025
Vancouver+ 10-14 August

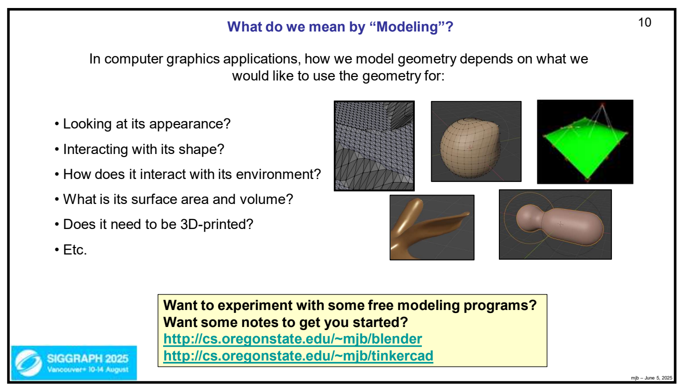
7


Modeling
Creating 3D Geometry
SIGGRAPH 2025
Vancouver+ 10-14 August

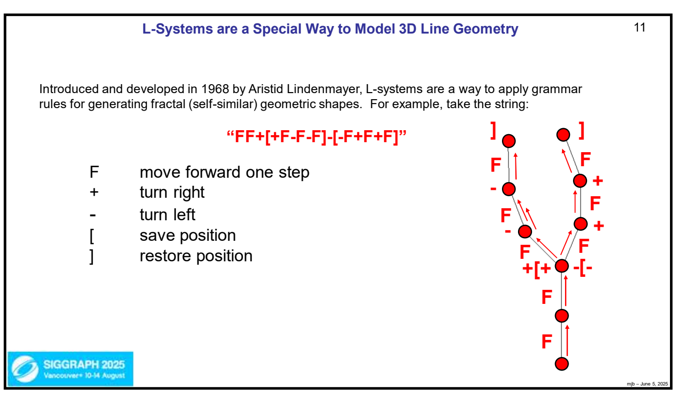
8


9
3D Geometric Models
Surface Information
Lighting Information
Texture Information
Drawing
Modeling
Rendering
Animation
3D Movement Definition
LET'S DO THIS.
SIGGRAPH 2025
Vancouver+ 10-14 August

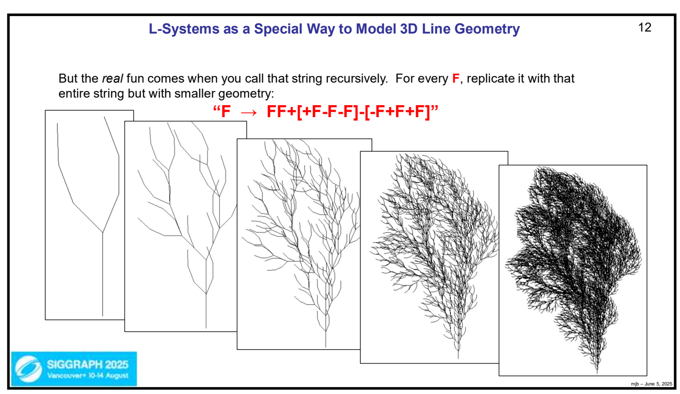
9


What do we mean by "Modeling"?
10
In computer graphics applications, how we model geometry depends on what we would like to use the geometry for:
• Looking at its appearance?
• Interacting with its shape?
• How does it interact with its environment?
• What is its surface area and volume?
• Does it need to be 3D-printed?
• Etc.
Want to experiment with some free modeling programs?
Want some notes to get you started?
http://cs.oregonstate.edu/~mjb/blender
http://cs.oregonstate.edu/~mjb/tinkercad
SIGGRAPH 2025
Vancouver+ 10-14 August

10


L-Systems are a Special Way to Model 3D Line Geometry
11
Introduced and developed in 1968 by Aristid Lindenmayer, L-systems are a way to apply grammar rules for generating fractal (self-similar) geometric shapes. For example, take the string:
"FF+[+F-F-F]-[-F+F+F]"
F move forward one step
+ turn right
- turn left
[save position
] restore position
SIGGRAPH 2025
Vancouver+ 10-14 August

11


L-Systems as a Special Way to Model 3D Line Geometry
12
But the real fun comes when you call that string recursively. For every F, replicate it with that entire string but with smaller geometry:
"F → FF+[+F-F-F]-[-F+F+F]"
SIGGRAPH 2025
Vancouver+ 10-14 August

12

L-Systems as a Special Way to Model 3D Line Geometry

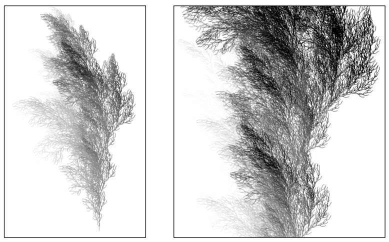
13

And, of course we can introduce more grammar to swing it into 3D

"F → FF+[+F-<F->F]-[-F+^F+vF]"

lsystems.mp4

F	move forward one step
+	rotate + about Z
-	rotate - about Z
<	rotate + about Y
>	rotate – about Y
v	rotate + about X
^	rotate – about X
[	save position
]	restore position

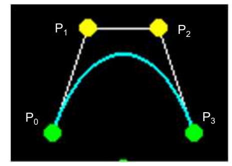


SIGGRAPH 2025 Vancouver+ 10-14 August

13

Another way to Model Geometry: Curve Sculpting

14



This equation is for a cubic Bezier curve:

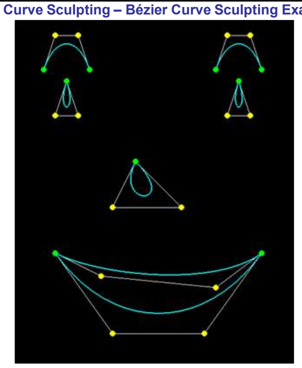
$$P(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t)P_2 + t^3 P_3$$
$$0. \le t \le 1.$$

SIGGRAPH 2025 Vancouver+ 10-14 August

14

Curve Sculpting – Bézier Curve Sculpting Example

15



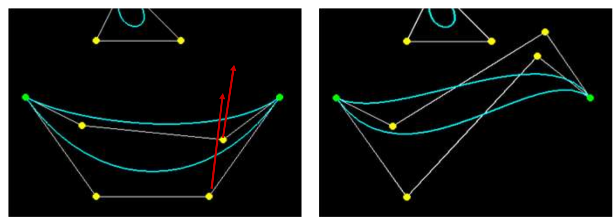
curves.mp4

SIGGRAPH 2025 Vancouver+ 10-14 August

15

Curve Sculpting – Bézier Curve Sculpting Example

16



A *Small* Amount of Input Change Results in a *Large* Amount of Output Change

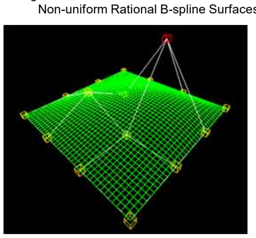
SIGGRAPH 2025 Vancouver+ 10-14 August

16

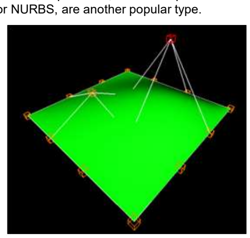
Another way to Model: Surface Sculpting

17

In general, these are referred to as ***Patches***. These, in particular, are Beziér patches. Non-uniform Rational B-spline Surfaces, or NURBS, are another popular type.



Wireframe



Surface

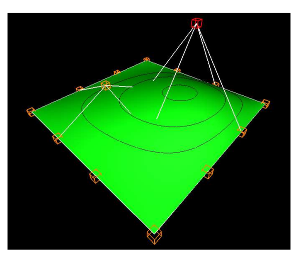
Like the curve sculpting, a *Small* Amount of Input Change Results in a *Large* Amount of Output Change

SIGGRAPH 2025 Vancouver+ 10-14 August

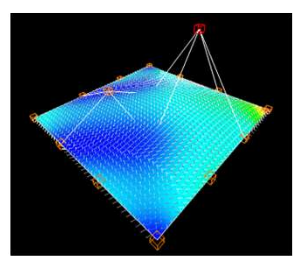
17

Surface Equations can also be used for Mathematical Analysis

18



Showing Contour Lines



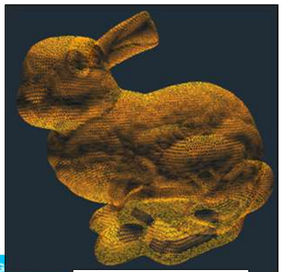
Showing Curvature

SIGGRAPH 2025 Vancouver+ 10-14 August

18

Explicitly Listing Geometry (3D Points) and Topology (How They Are Connected)

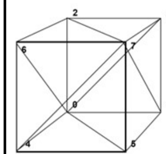
Models defined this way can consist of thousands of vertices and faces – we need some way to describe them



This is often called a **Mesh**, or sometimes a **Triangular Irregular Network** (TIN).

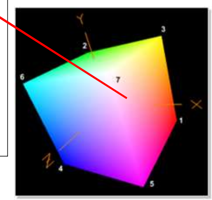
http://graphics.stanford.edu/data/3Dscanrep

Explicitly Listing Triangular Mesh Coordinates and Connections in Tables

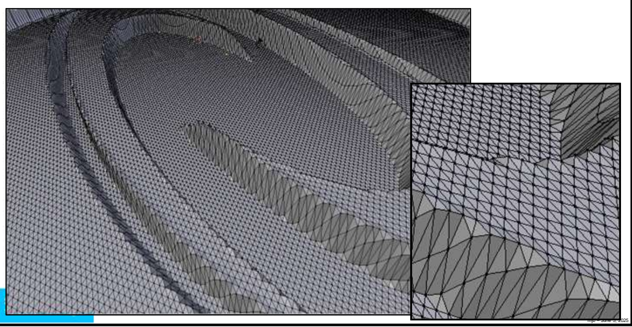



```
float VertexCoordinates[ ][3] =
{
        { -1., -1., -1. },
        {  1., -1., -1. },
        { -1.,  1., -1. },
        {  1.,  1., -1. },
        { -1., -1.,  1. },
        {  1., -1.,  1. },
        { -1.,  1.,  1. },
        {  1.,  1.,  1. }
};
```

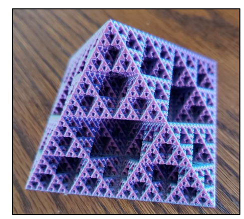
```
int Indexes[ ][3] =
{
        { 0, 2, 3 },
        { 0, 3, 1 },
        { 4, 5, 7 },
        { 4, 7, 6 },
        { 1, 3, 7 },
        { 1, 7, 5 },
        { 0, 4, 6 },
        { 0, 6, 2 },
        { 2, 6, 7 },
        { 2, 7, 3 },
        { 0, 1, 5 }
        { 0, 5, 4 }
};
```



Triangular Meshes are Super Important These Days Because *3D Printing* Requires a Triangular Mesh Data Format



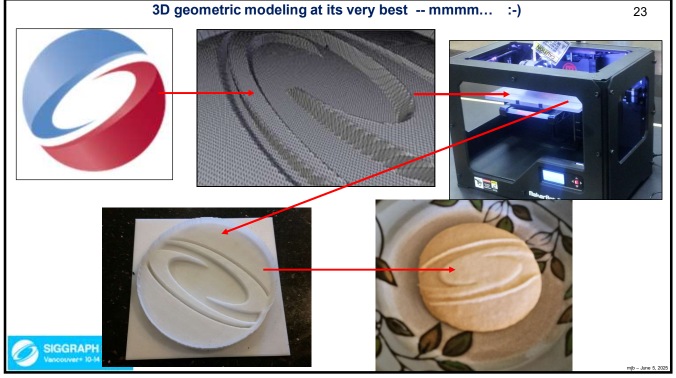
3D Printing Meshes Don't Always Look Very Mesh-ish Anymore – But They Are



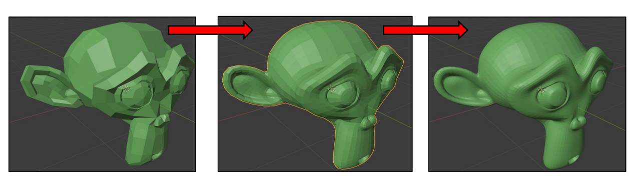
Models are from https://www.printables.com

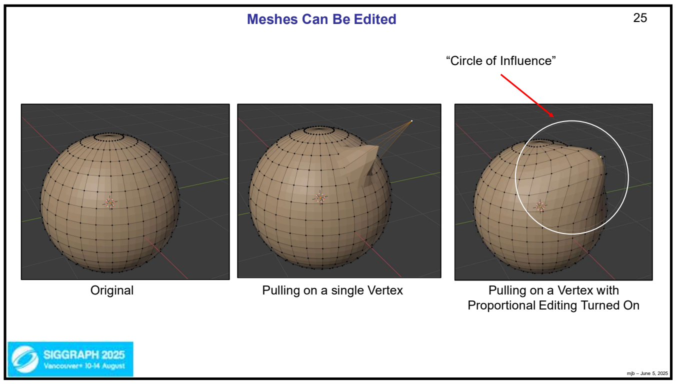
3D Printed by Ryan Bailey
Images used by permission

3D geometric modeling at its very best -- mmmm… :-)

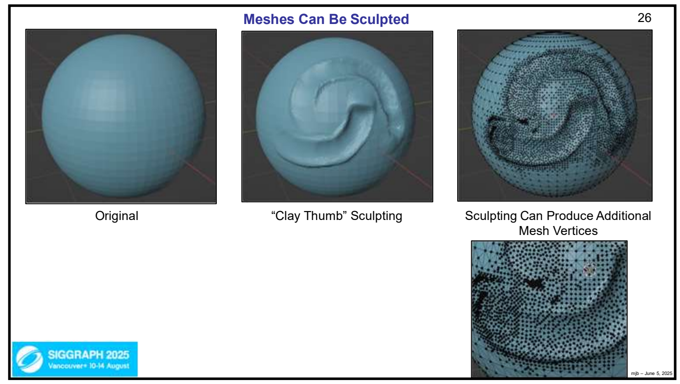


Meshes Can Be Smoothed

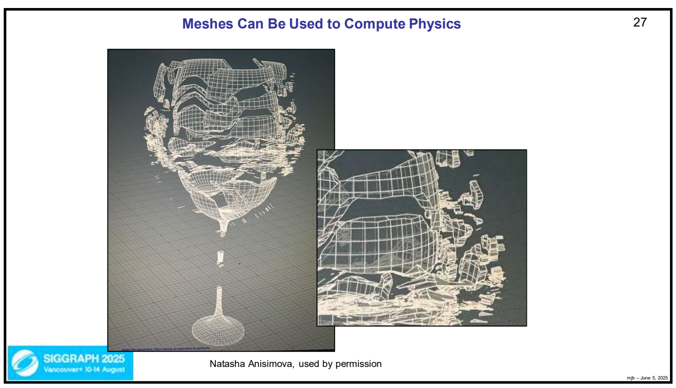




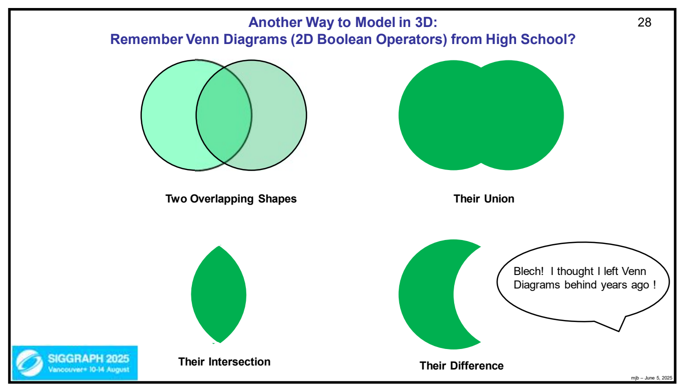
25



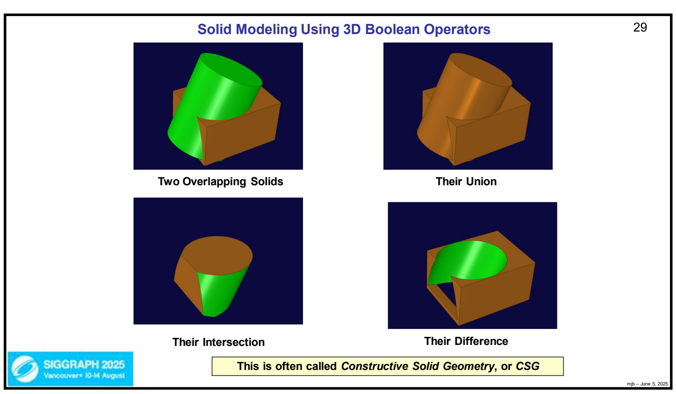
26



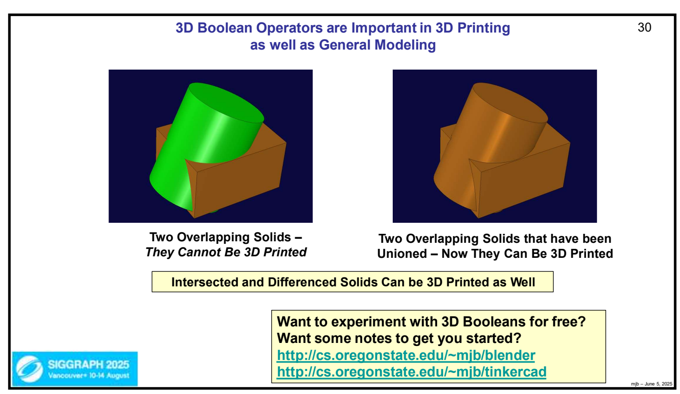
27



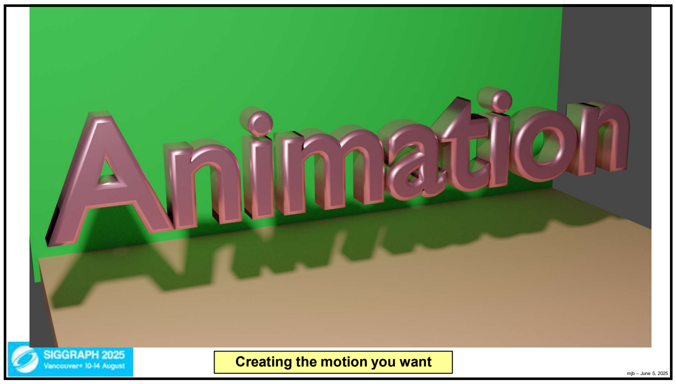
28



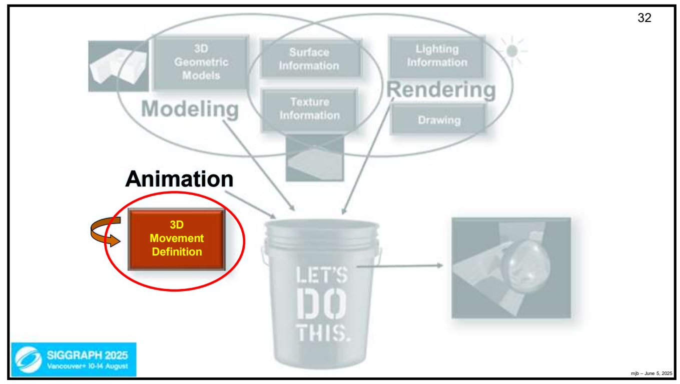
29



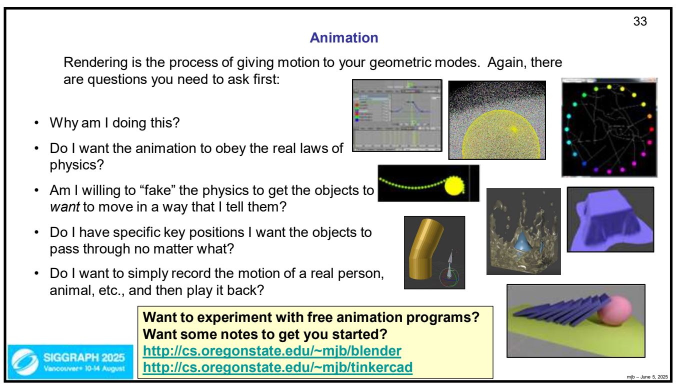
30



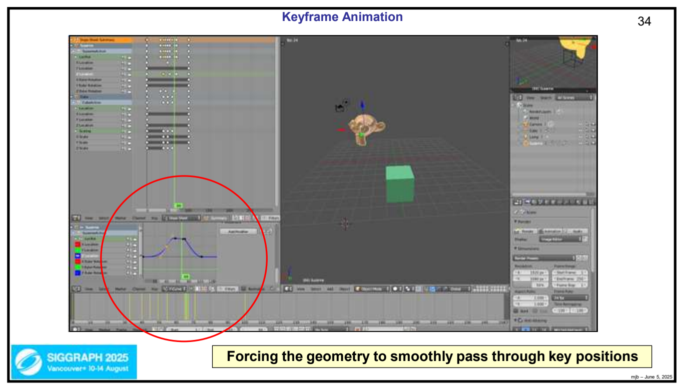
31



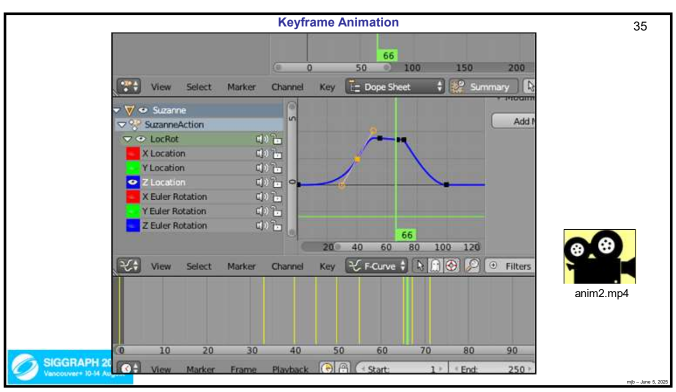
32



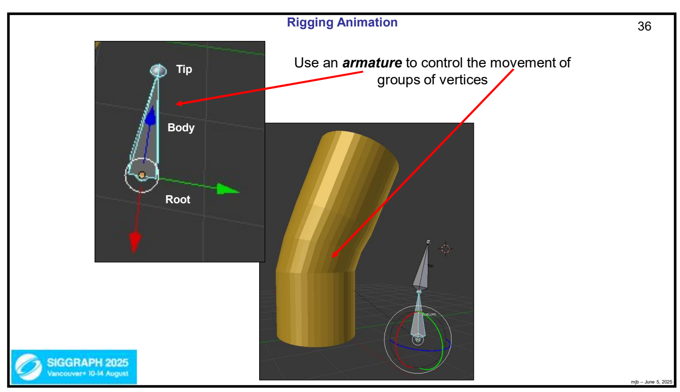
33



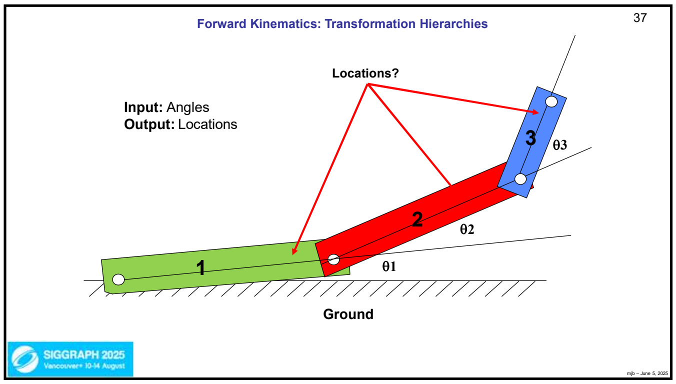
34



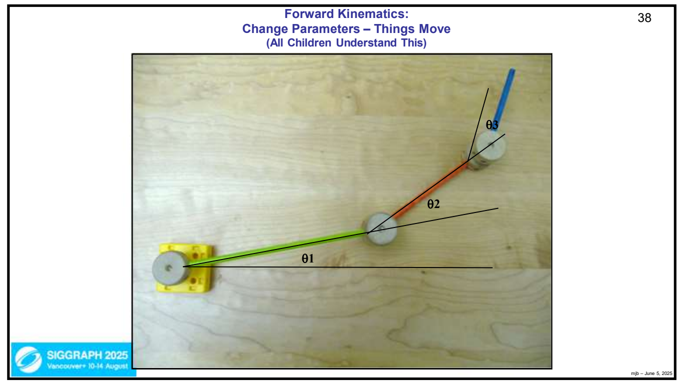
35



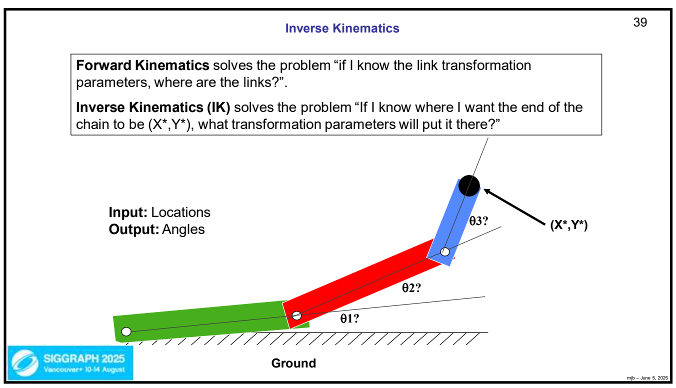
36



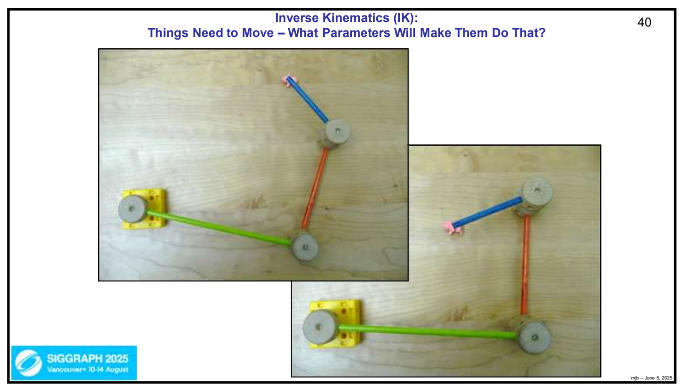
37



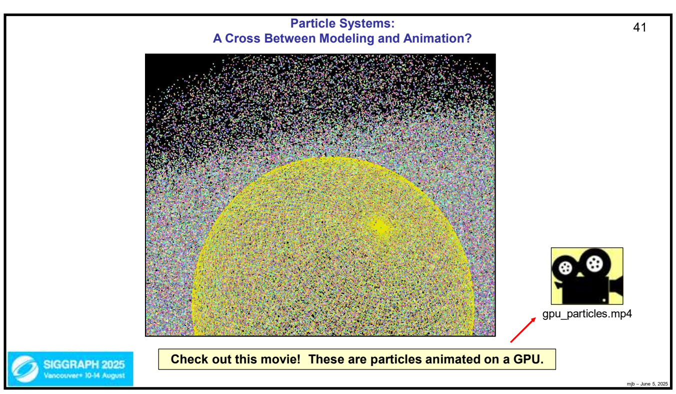
38



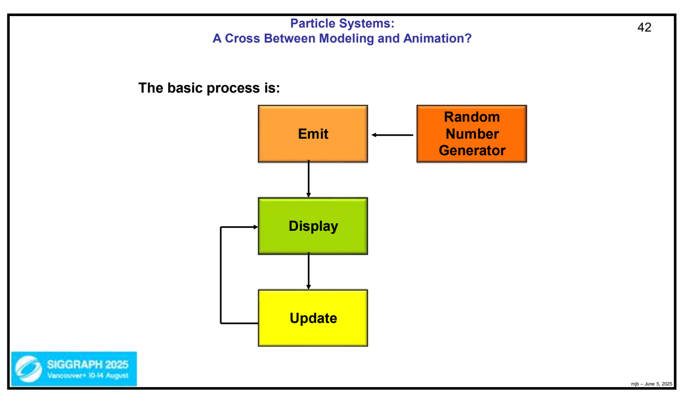
39



40



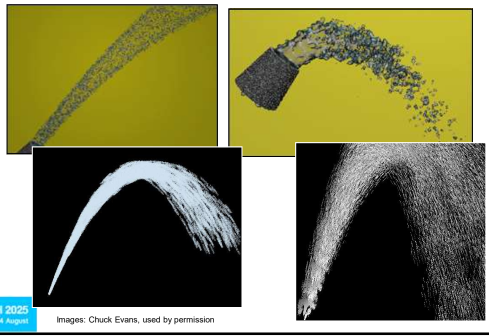
41



42

Particle Systems Examples

43



Images: Chuck Evans, used by permission

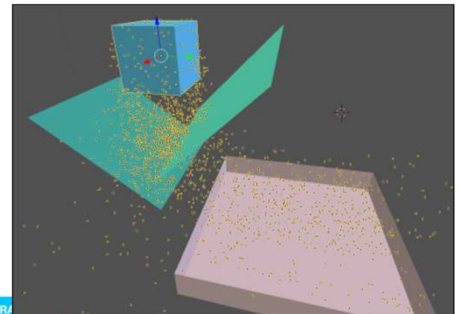
SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

43

Particle Systems Examples

44



particles.mp4

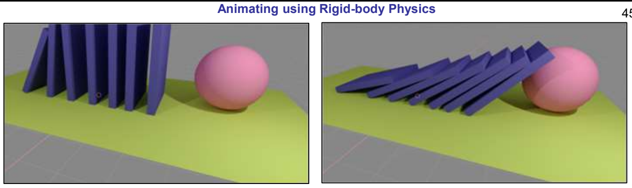
SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

44

Animating using Rigid-body Physics

45



Newton's second law:

force = mass * acceleration

or:

acceleration = $\{\ddot{x}\}$ = force / mass

Newton's Second Law

dominos2.mp4

To make this work, you need to supply physical properties such as mass, center of mass, moment of inertia, coefficients of friction, coefficients of restitution, etc.

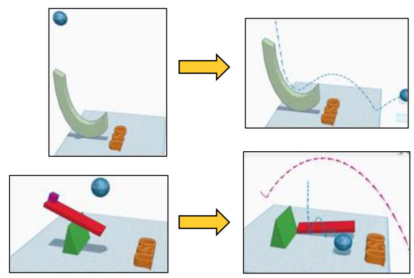
SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

45

Even TinkerCad Now Has Rigid Body Physics Animation

46



Want to experiment with TinkerCad physics animation for free?
Want some notes to get you started?
http://cs.oregonstate.edu/~mjb/tinkercad

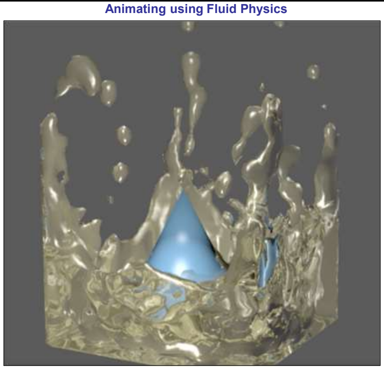
SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

46

Animating using Fluid Physics

47



fluid.avi

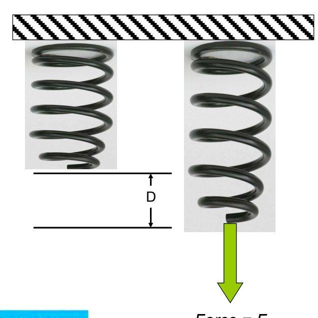
SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

47

Animating using Spring Physics

48



k = ***spring stiffness*** in newtons/cm or pounds/inch

If you know the force, the distance the spring stretches or compresses will be:

$$D = \frac{F}{k}$$

Or, if you know the distance the spring stretches or compresses, the force exerted by the spring will be:

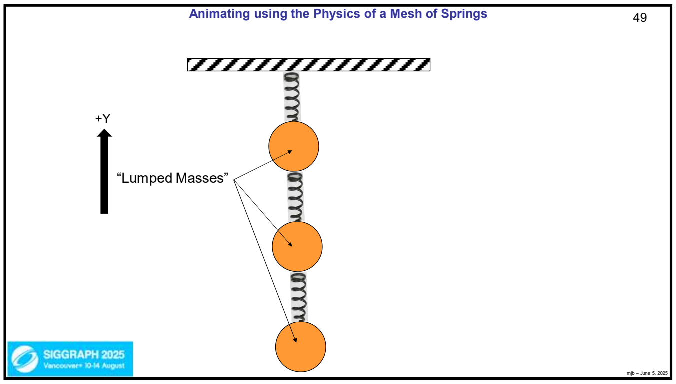
$$F = kD$$

This is known as *Hooke's Law*

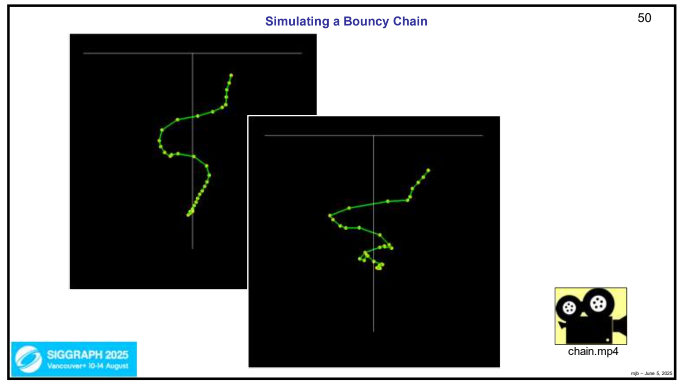
SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

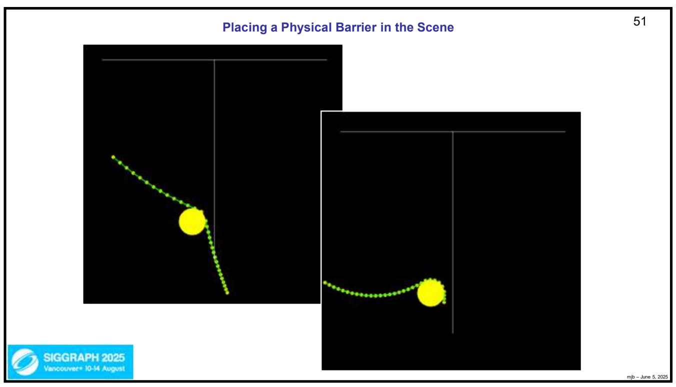
48



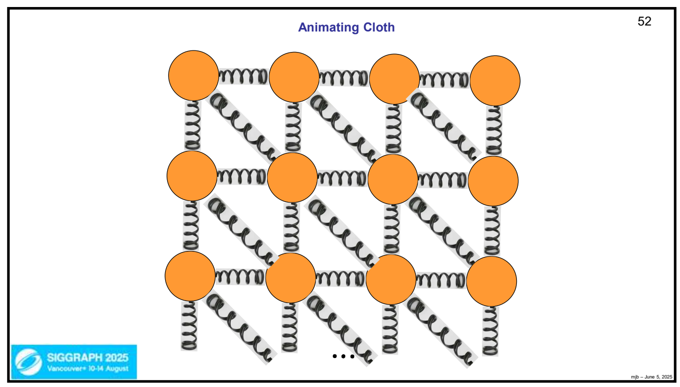
49



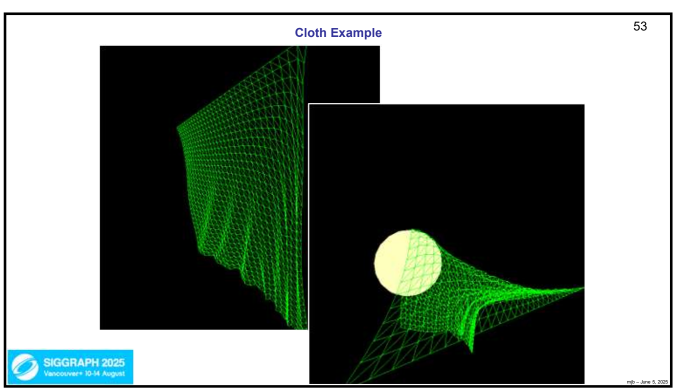
50



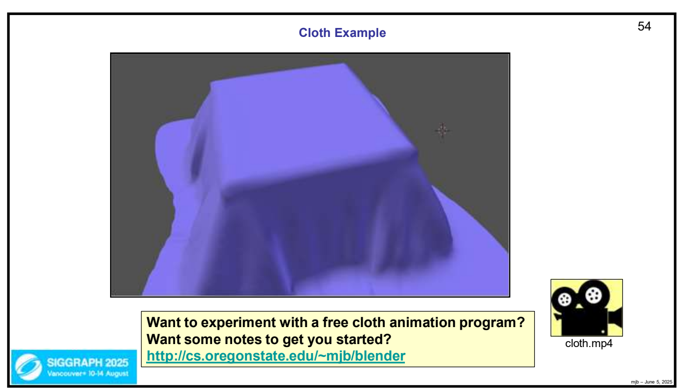
51



52



53



54

Functional Animation – "Fake Physics"

55

The Challenge: animate a collection of objects, each trying to move to a target, but without colliding with each other.

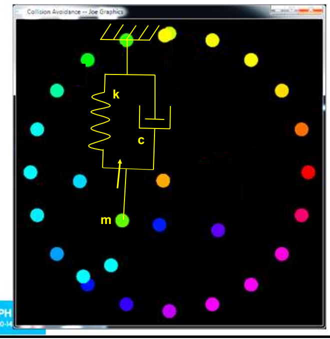
SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

55

Functional Animation: Make the Object *Want* to Move Towards a Goal Position . . .

56



$$m\ddot{x}+c\dot{x}+kx=0$$

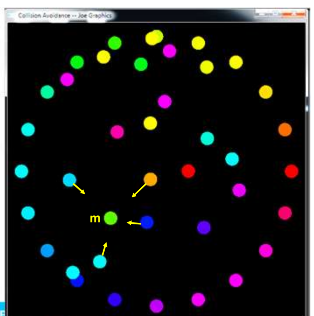
Actual equation of motion for a spring-mass-damper system

mjb – June 5, 2025

56

Functional Animation: . . . While Making it *Want* to Keep Away from all other Objects

57



$$m\ddot{x} = \sum F_{repulsive}$$

Repulsion Coefficient

$$F_{repulsive} = \frac{C_{repulse}}{d^{Power}}$$

Distance between the boundaries of the 2 bodies

Repulsion Exponent

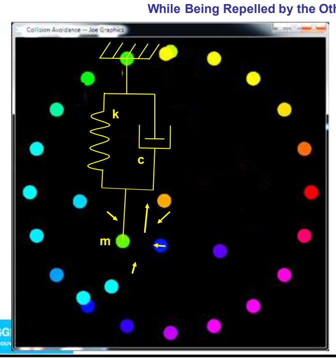
Fake equation of motion for two masses trying to push each other away – I just made this up…

mjb – June 5, 2025

57

Total Goal – Make the Free Body Move Towards its Final Position While Being Repelled by the Other Bodies

58



$$m\ddot{x}+c\dot{x}+kx = \sum F = \sum F_{repulsive}$$

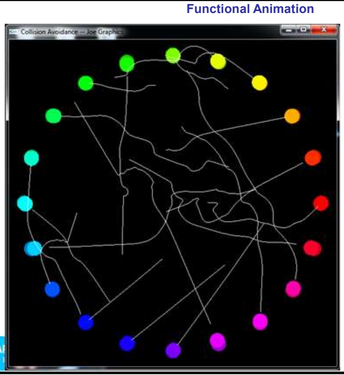
If we set the actual and fake equations in motion, what will happen?

mjb – June 5, 2025

58

Functional Animation

59



We get a collection of objects, each trying to move to a target, but without colliding with each other.

avoid.mp4

mjb – June 5, 2025

59

Motion Capture ("MoCap") as an Input for Animation

60

Natural Point

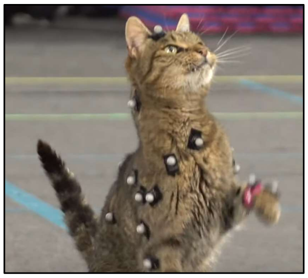
Photos: Mike Bailey

mjb – June 5, 2025

60

Even Animals can be MoCapped (if you dare…)

61



https://www.youtube.com/watch?v=zyq_LQrHpoo

Photo courtesy of: DIGIC Services' Mocap Studio, used by permission

SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

61

62

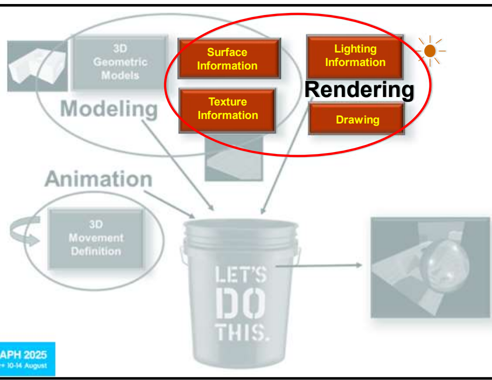
Creating an image of your scene

SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

62

63



SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

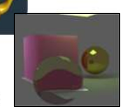
63

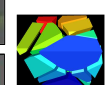
Rendering

64

Rendering is the process of creating an image of geometric models. Again, there are questions you need to ask first:

- Why am I doing this?
- How realistic do I want this image to be?
- How much compute time do I want this to take?
- Do I need to take lighting into account?
- Does the illumination need to be global or will local do?
- Do I need to create shadows?
- Do I need to create reflections and refractions?





Want to experiment with a free rendering program?
Want some notes to get you started?
http://cs.oregonstate.edu/~mjb/blender

SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

64

Non-Photorealistic Rendering: Toon Shading

65

Photo by Steve Cunningham, used with permission

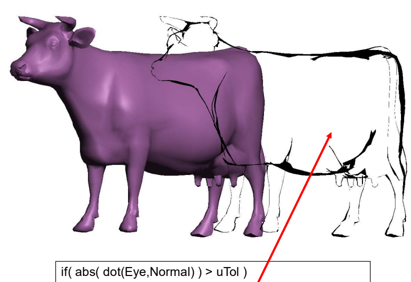
SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

65

Non-Photorealistic Rendering: Silhouettes using Shaders

66

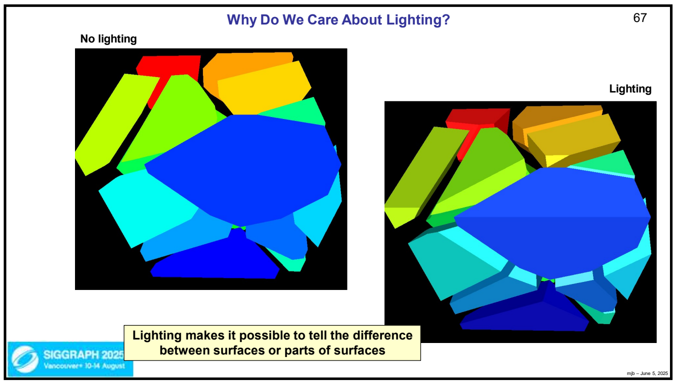


```
if( abs( dot(Eye,Normal) ) > uTol )
        discard;
else
        gl_FragColor = vec4( SILHCOLOR, 1. );
```

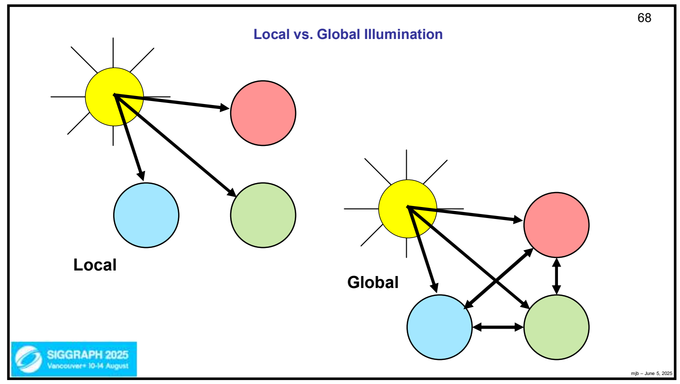
SIGGRAPH 2025 Vancouver+ 10-14 August

mjb – June 5, 2025

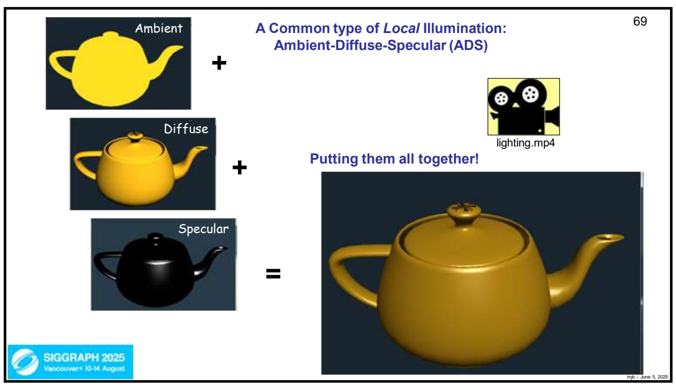
66



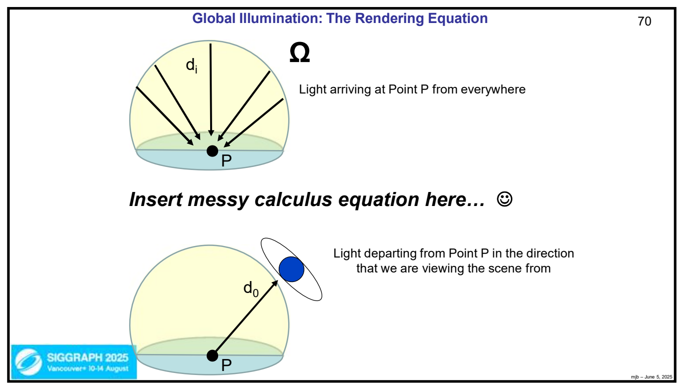
67



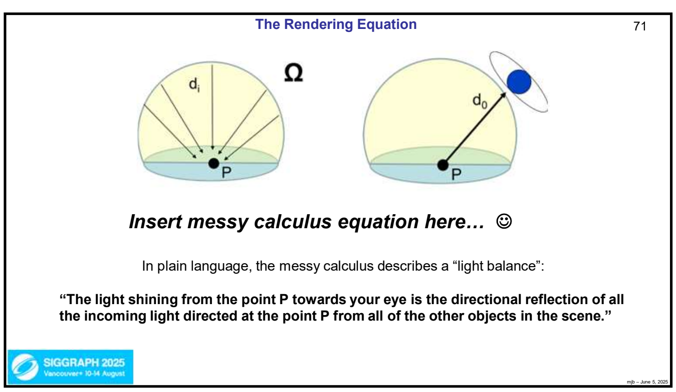
68



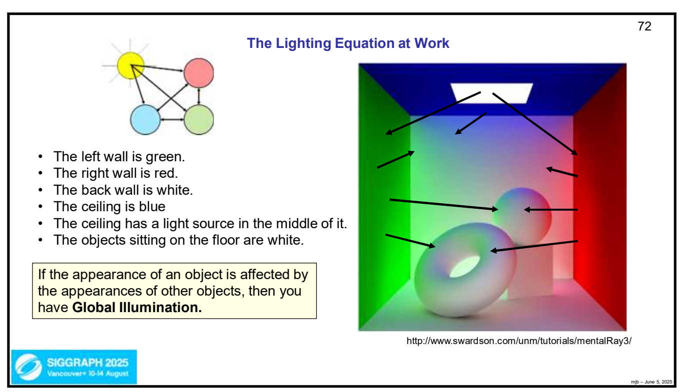
69



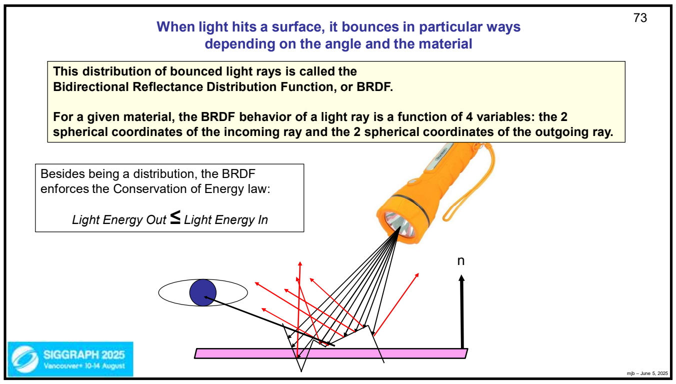
70



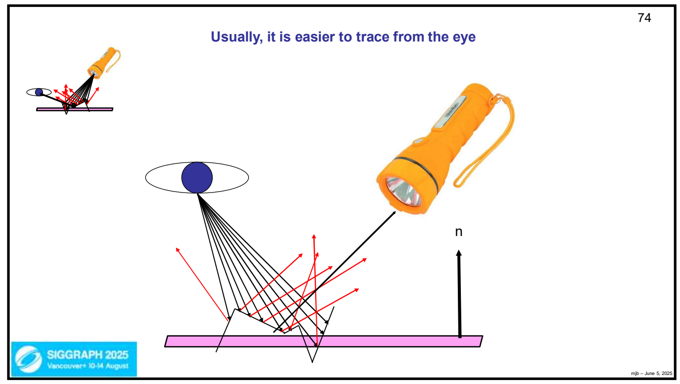
71



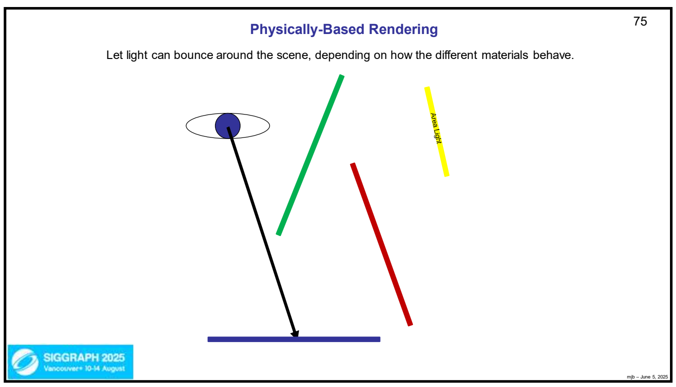
72



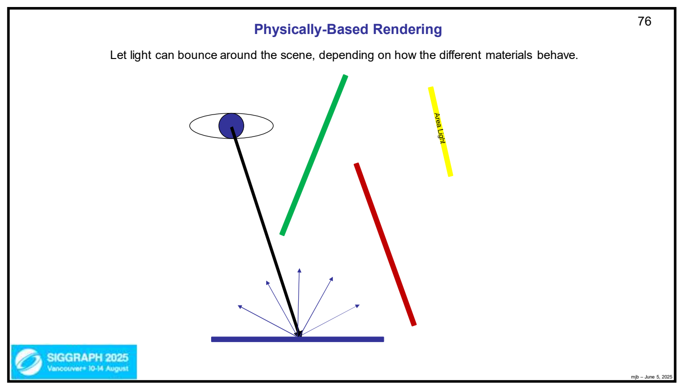
73



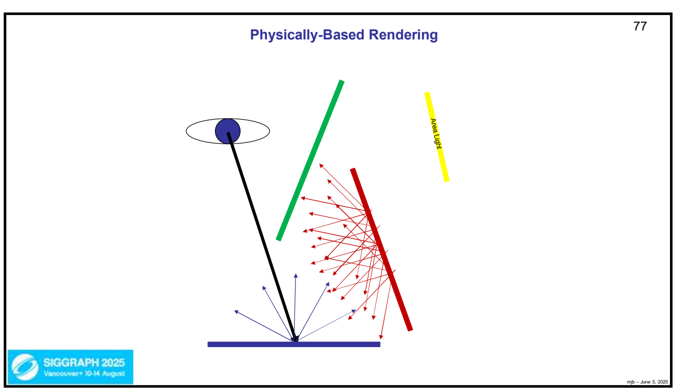
74



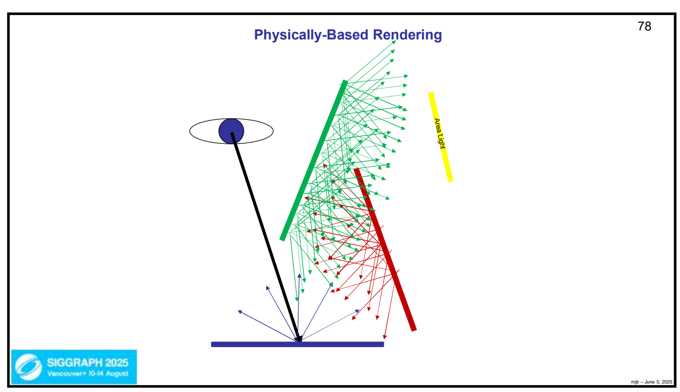
75



76



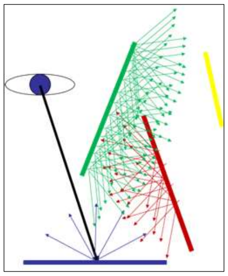
77



78

Physically-Based Rendering

79



Clearly this process can spawn an infinite number of rays. How do we handle this?

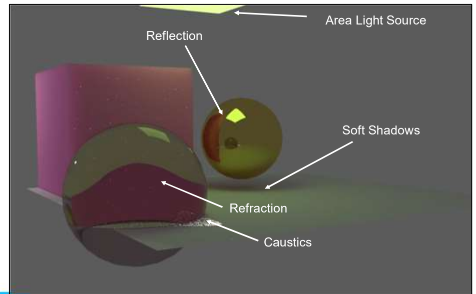
We can't use all rays that are possible, so we use a statistical subset of the rays.

This is known as **Monte Carlo** simulation.

79

Physically-Based Rendering using the Blender *Cycles* Renderer

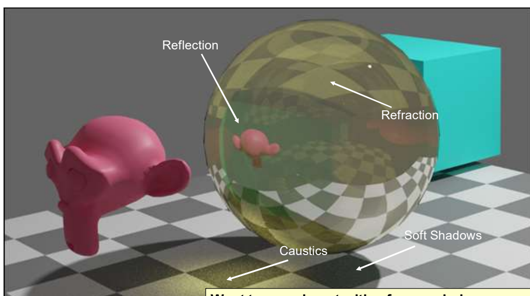
80



80

Physically-Based Rendering using the Blender *Cycles* Renderer

81

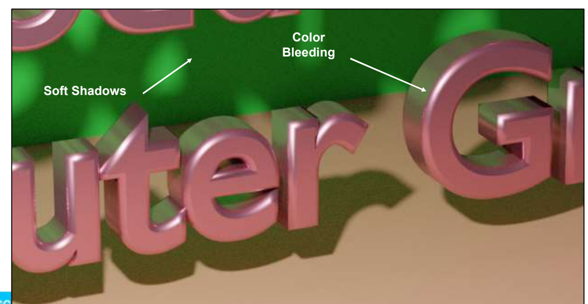


Want to experiment with a free rendering program?
Want some notes to get you started?
http://cs.oregonstate.edu/~mjb/blender

81

An Example from the Title Slide

82



82

More from the Blender *Cycles* Renderer

83

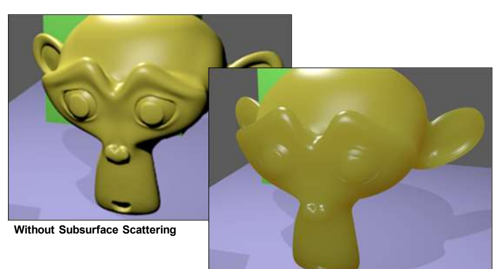
Want to experiment with a free rendering program?
Want some notes to get you started?
http://cs.oregonstate.edu/~mjb/blender

83

Tricky Lighting Situations -- Subsurface Scattering

84

Subsurface Scattering can model light bouncing around *within* an object before coming back out. This is a good way to represent skin, wax, milk, etc.

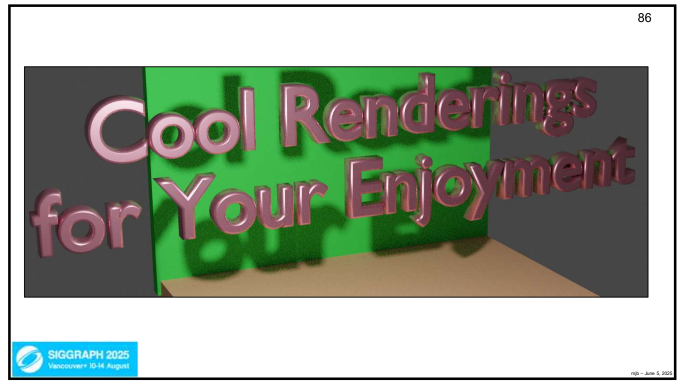


Without Subsurface Scattering

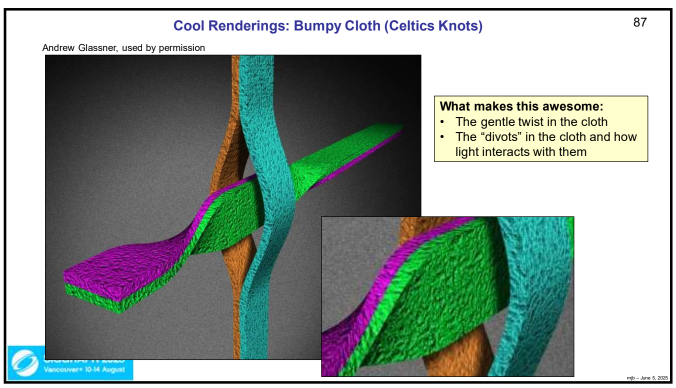
With Subsurface Scattering

84

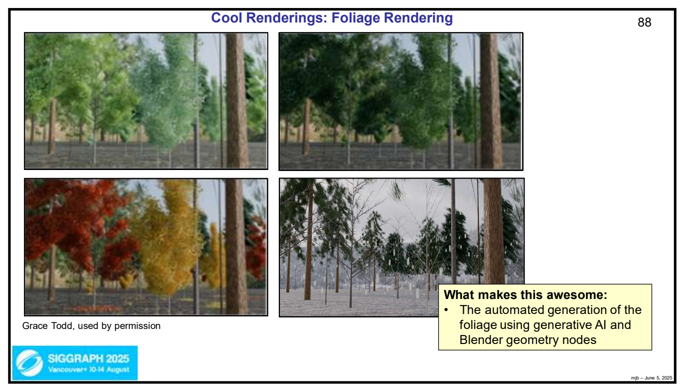
85



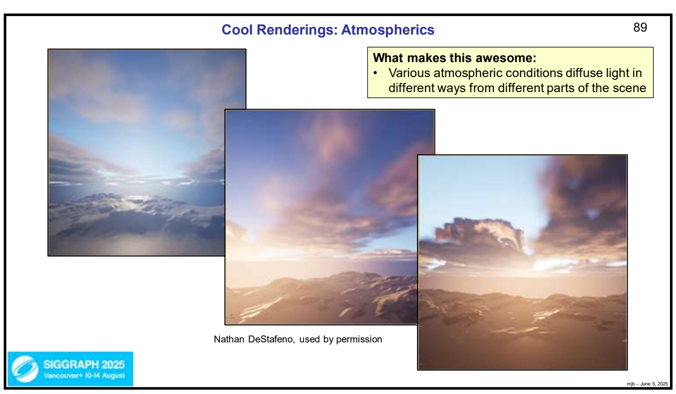
86



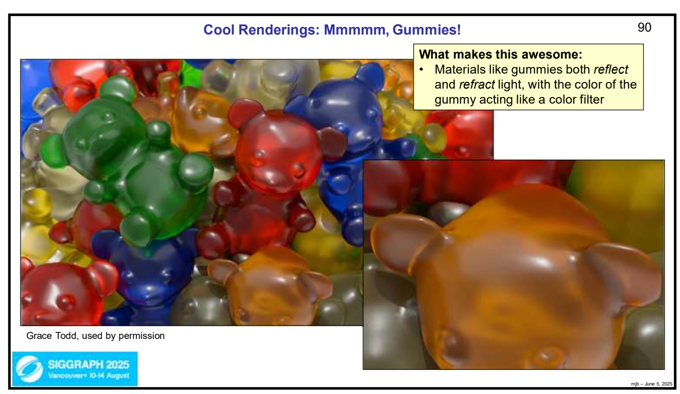
87



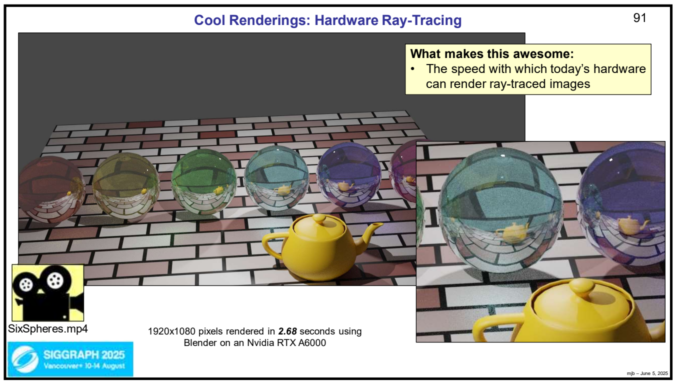
88



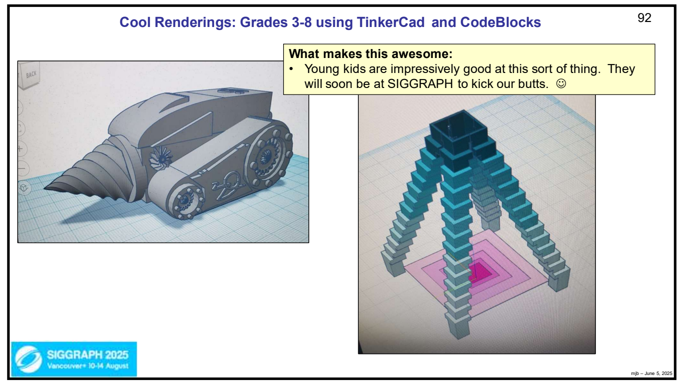
89



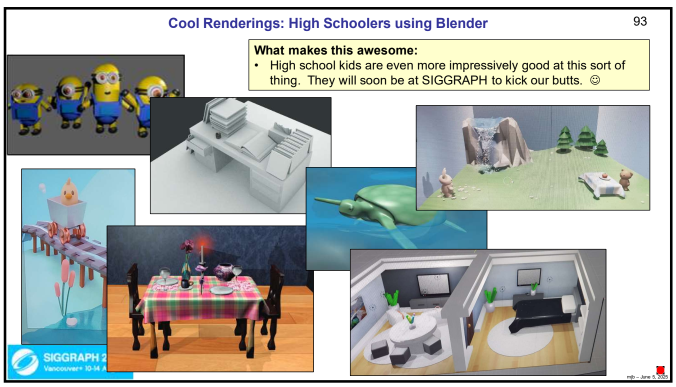
90



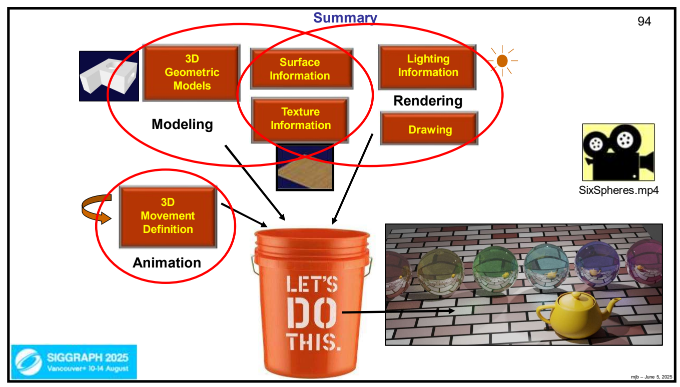
91



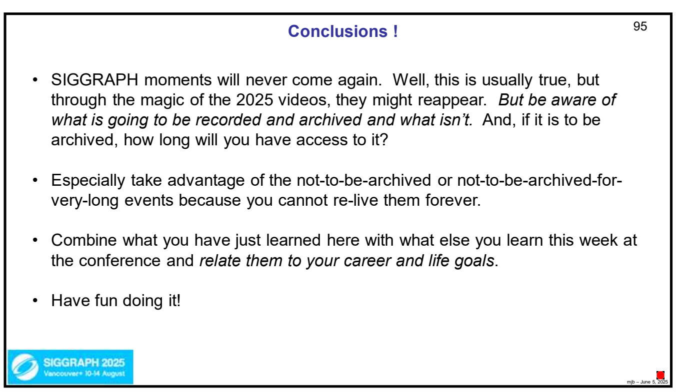
92



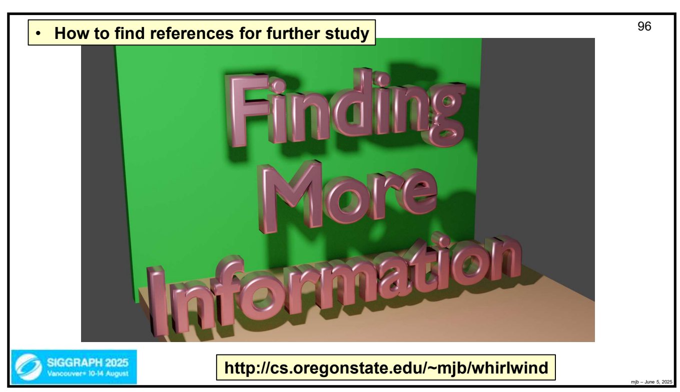
93



94



95



96

Check Out the *More Information* Document to be found at: 97

Where to Find More Information about Computer Graphics and Related Topics

Mike Bailey
Oregon State University

1. References

1.1 General Computer Graphics

SIGGRAPH Online Bibliography Database:
http://www.siggraph.org/learn/computer-graphics-bibliography-database

Edward Angel and Dave Shreiner, *Interactive Computer Graphics: A Top-down Approach with OpenGL*, 6th Edition, Addison-Wesley, 2011.

Francis Hill and Stephen Kelley, *Computer Graphics Using OpenGL*, 3rd Edition, Prentice Hall, 2006.

Steve Cunningham, *Computer Graphics: Programming in OpenGL for Visual Communication*, Prentice-Hall, 2007

Alan Watt, *3D*

Peter Shirley, *Fundamentals* of Computer Graphics, 2nd Edition, AK Peters, 2005.

Andrew Glassner, *Graphics Gems*, Academic Press, 1990.

http://cs.oregonstate.edu/~mjb/whirlwind

SIGGRAPH 2025 Vancouver+ 10-14 August

97

Check Out Other Sets of Free Notes: 98

http://cs.oregonstate.edu/~mjb/cgeducation

University course notes:
- Introduction to Computer Graphics
- Computer Graphics Shaders
- CS Skills for Simulation and Game Programming
- Parallel Programming
- Scientific Visualization
- Vulkan

SIGGRAPH course notes:
- A Whirlwind Introduction to Computer Graphics

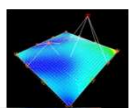
K-12 notes:
- Blender
- CodeBlocks
- Processing
- Scratch
- Scratch Jr.
- SimLab
- Tinkercad

SIGGRAPH 2025 Vancouver+ 10-14 August

98

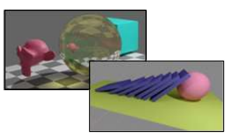
A Whirlwind Introduction to Computer Graphics 99

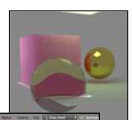
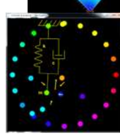




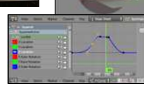
Thank You!

Mike Bailey
mjb@cs.oregonstate.edu



http://cs.oregonstate.edu/~mjb/whirlwind



SIGGRAPH 2025 Vancouver+ 10-14 August

99