CSE 5542 - Real Time Rendering

TBT

(Not So) Real Time Rendering

Where ?

Time - TR 11:10 AM – 12:30 pM
Place - DL 0264

Labs ?

- Your own machine …

- Graphics PC Lab – CL 112D ?

- Platforms: PC (visual studio), Mac OS X or Linux

Who Am I ?

The Instructor

Name: Raghu Machiraju

Email: machiraju.1@osu.edu

Office hours: M: 3-4 PM
TR12:30 PM – 1:30 PM

Grading

The Grader

Name: Tzi-Husan Wei

Email: TBA

Office hours:

Monday: 4:30-5:30 PM
Wed: 3:00-5:00 PM
Fri: 3:00-5:00 PM

Grading

- Labs: 45% (10+12+13+10) - Strict deadlines.
- Final Project: 15% - No Final Exam
- Quizzes: 20%(5x4)
- (Take Home) Midterm:15%

Course
Information

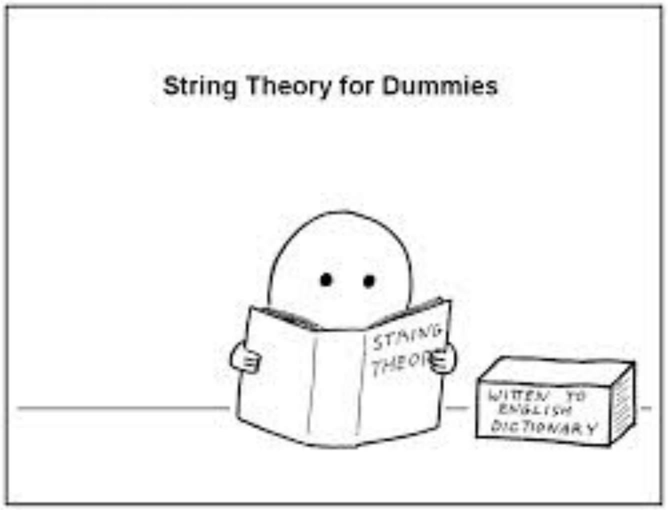
Information

Web: http://www.cse.ohio-state.edu/~raghu/5542

Piazza: https://piazza.com/osu/spring2015/cse5542/home

Prerequisite: 3901 (560) or 3902 or 3903
math 2568 (568) or 571
permission from the instructor

Pre-reqs



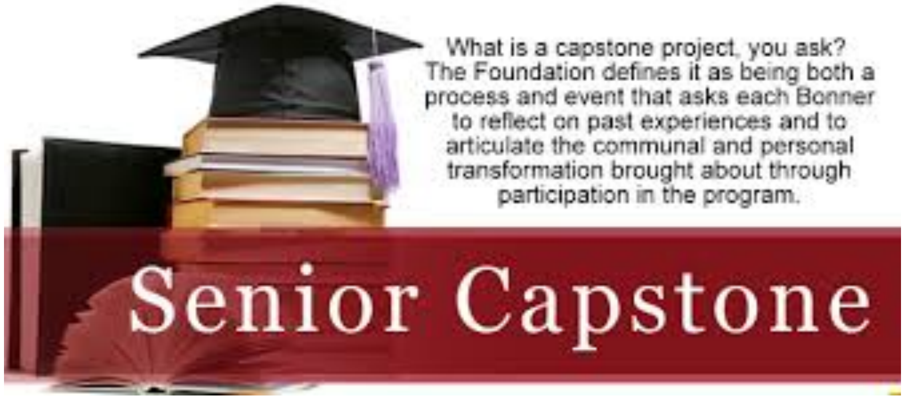
STEP 1: begin

Must Haves

- ✓ You need to be enthusiastic about Computer Graphics
- ✓ You need to be fluent in C/C++/Java programming
- ✓ You need to be comfortable with linear algebra
- ✓ You need to be willing to get hands-dirty: OpenGL, WebGL, 3D Printing, GLSL, hardware

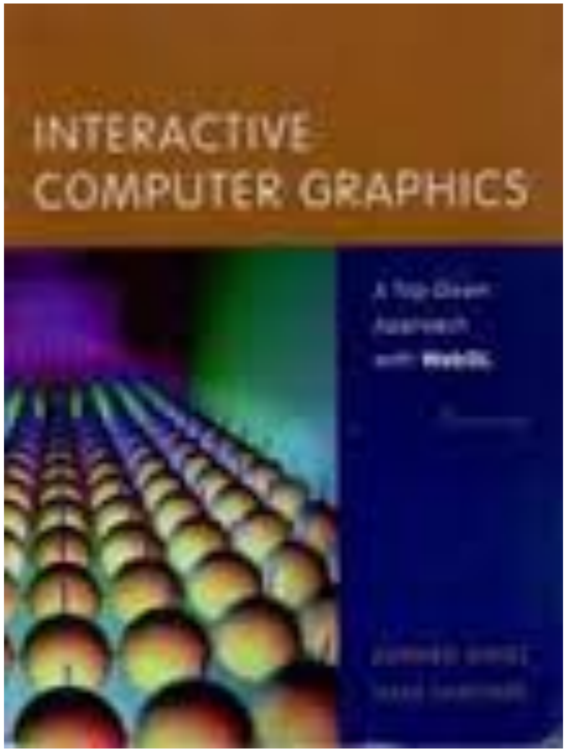
CAVEATS – Not a …

Closer to …

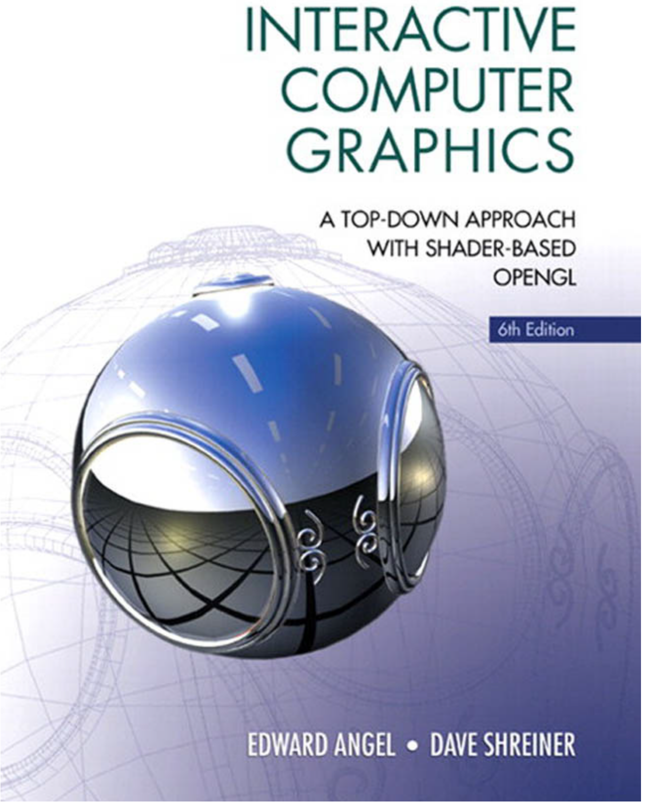


What is a capstone project, you ask? The Foundation defines it as being both a process and event that asks each Bonner to reflect on past experiences and to articulate the communal and personal transformation brought about through participation in the program.

The Book

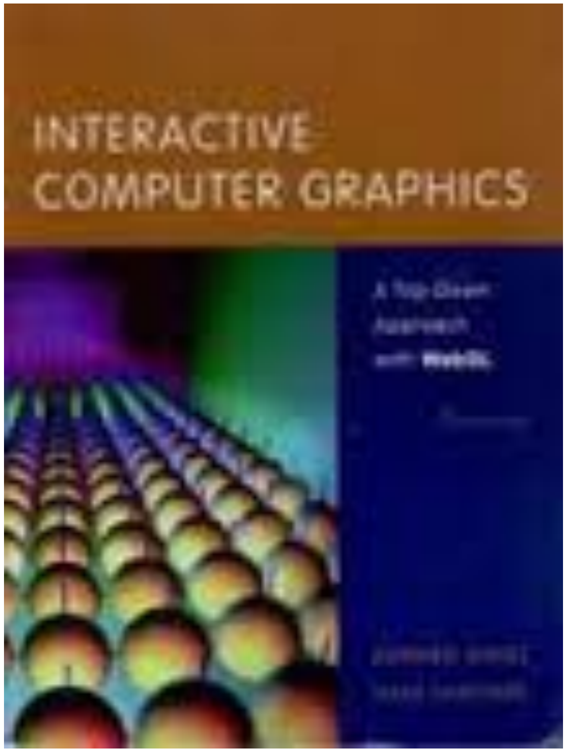

INTERACTIVE COMPUTER GRAPHICS
A Top-Down Approach with WebGL
EDWARD ANGEL
DAVE SHREINER

Earlier …


INTERACTIVE COMPUTER GRAPHICS
A TOP-DOWN APPROACH WITH SHADER-BASED OPENGL
6th Edition
EDWARD ANGEL • DAVE SHREINER

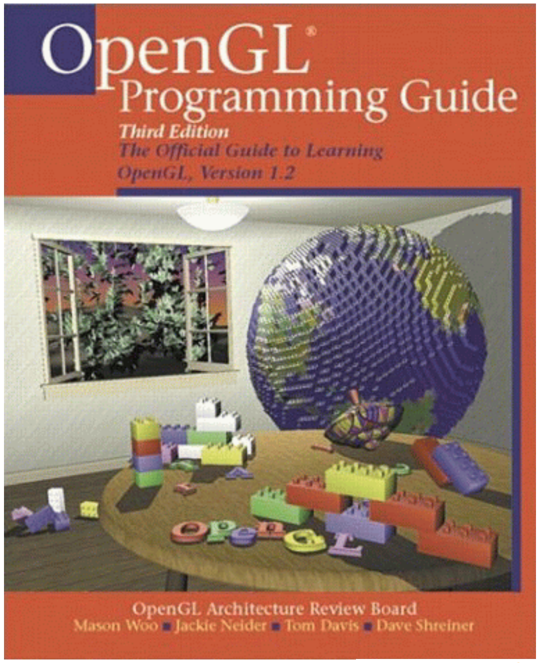
Diff 7e 6e

Will Follow Text Closely

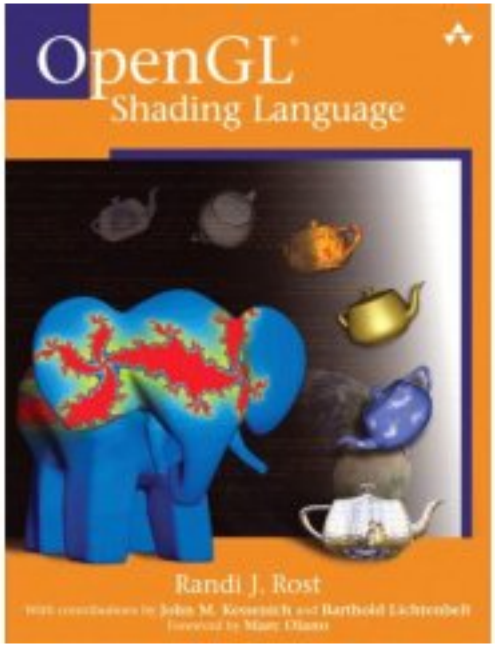


Useful Books – OpenGL, GLSL

http://proquest.safaribooksonline.com

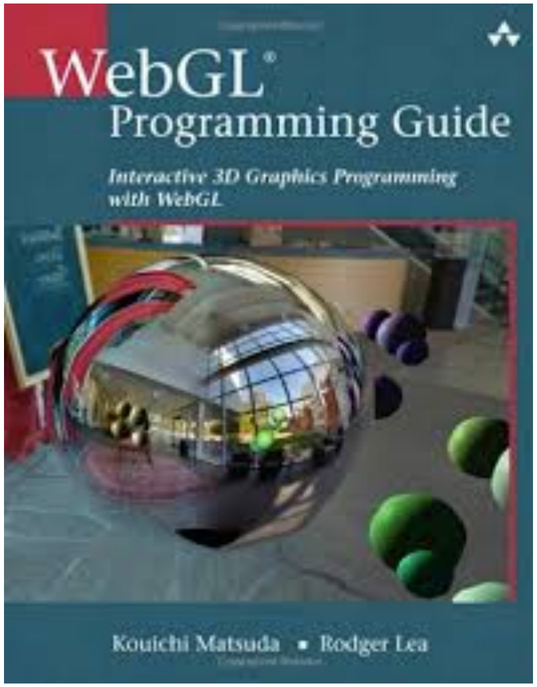

OpenGL®
Programming Guide
Third Edition
The Official Guide to Learning OpenGL, Version 1.2
OpenGL
OpenGL Architecture Review Board
Mason Woo ■ Jackie Neider ■ Tom Davis ■ Dave Shreiner

OpenGL programming Guide , 8th edition

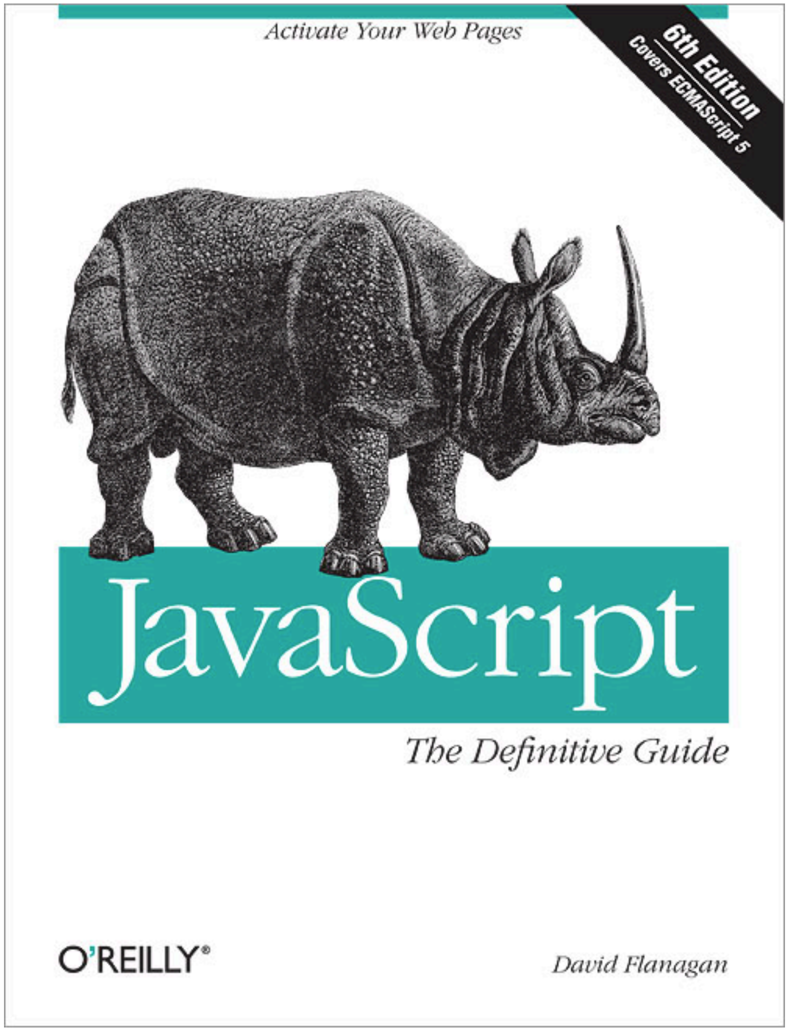

OpenGL®
Shading Language
Randi J. Rost
With contributions by John M. Kessenich and Barthold Lichtenbelt
Foreword by Marc Olano

OpenGL shading language 3rd edition

Useful Books - WebGL

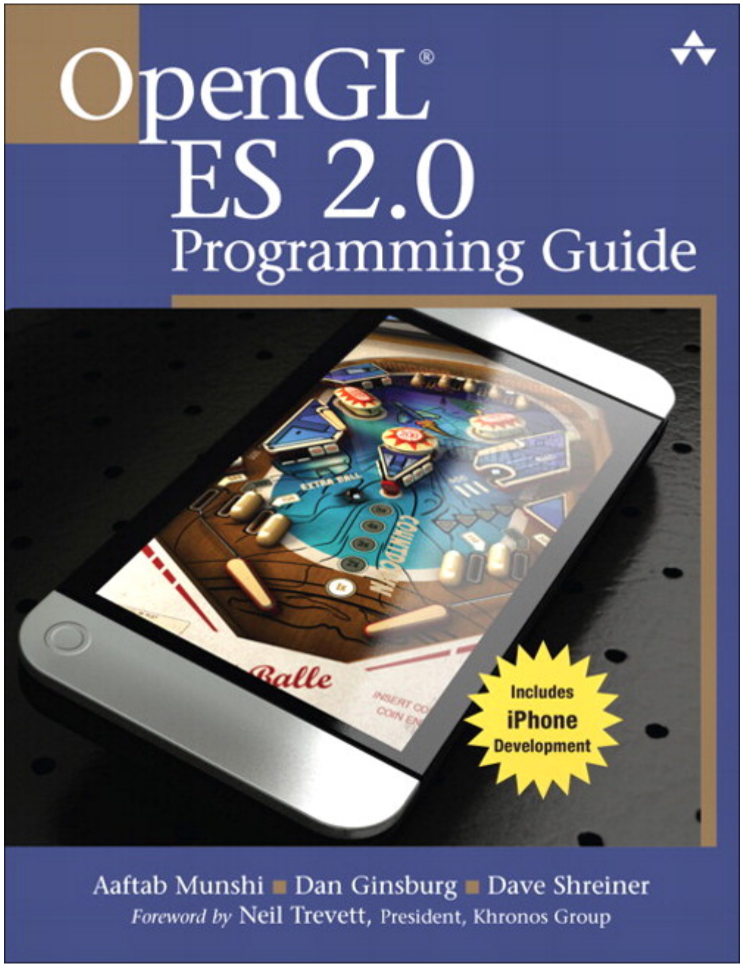

WebGL®
Programming Guide
Interactive 3D Graphics Programming with WebGL
Kouichi Matsuda • Rodger Lea

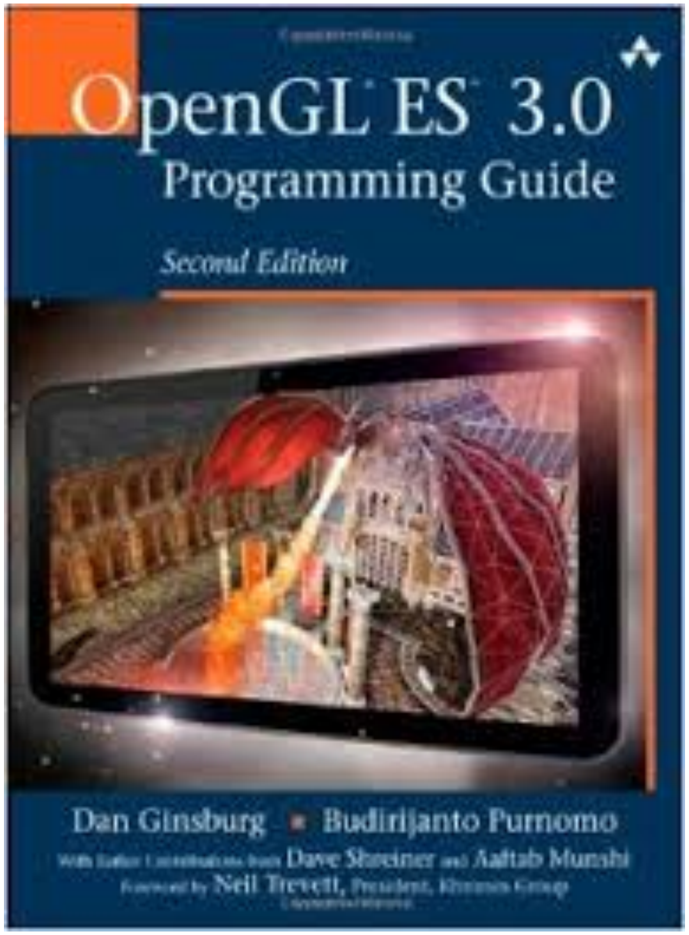
Useful Books - JavaScript



Useful Tool - Blender

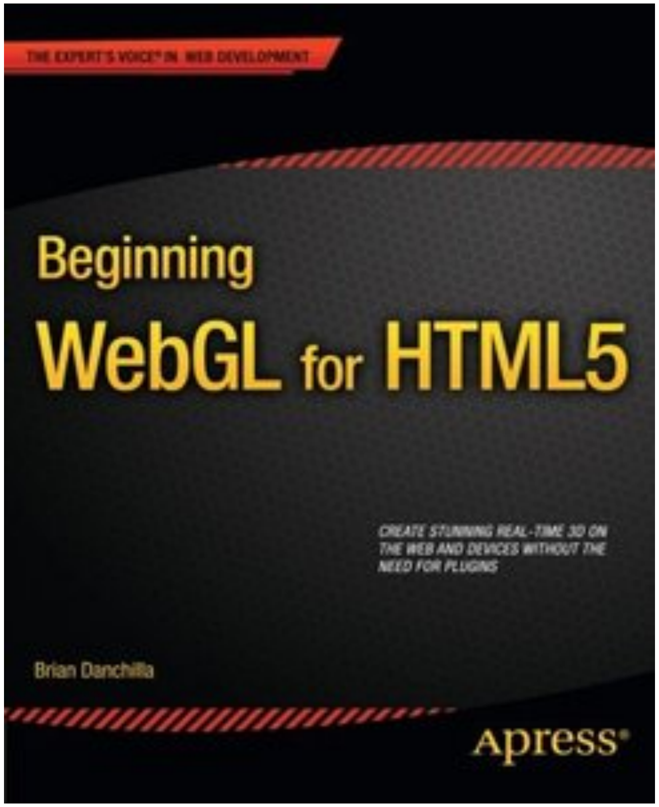
Reference Books – OpenGL/ES


OpenGL® ES 2.0
Programming Guide
Includes iPhone Development
Aaftab Munshi ■ Dan Ginsburg ■ Dave Shreiner
Foreword by Neil Trevett, President, Khronos Group


OpenGL® ES™ 3.0
Programming Guide
Second Edition
Dan Ginsburg ■ Budirijanto Purnomo
With Earlier Contributions from Dave Shreiner and Aaftab Munshi
Foreword by Neil Trevett, President, Khronos Group

https://www.khronos.org/

Reference Books – HTML5



THE EXPERT'S VOICE® IN WEB DEVELOPMENT

Beginning

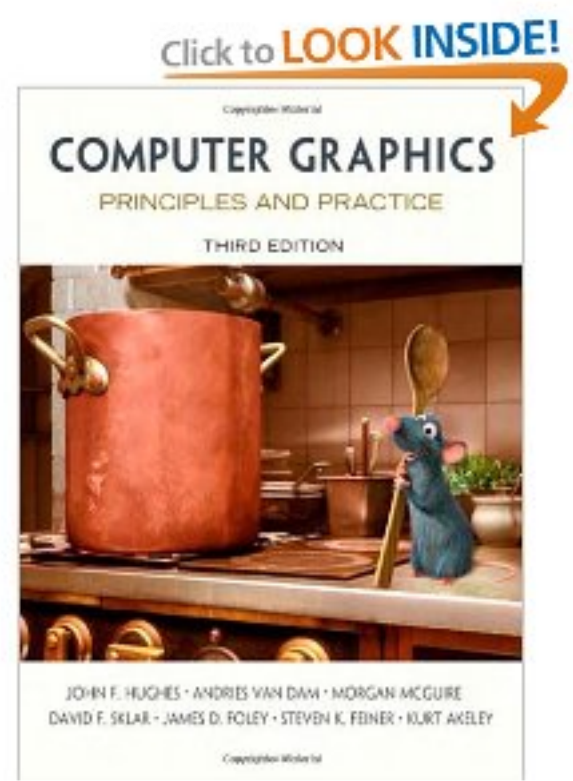
WebGL for HTML5

CREATE STUNNING REAL-TIME 3D ON THE WEB AND DEVICES WITHOUT THE NEED FOR PLUGINS

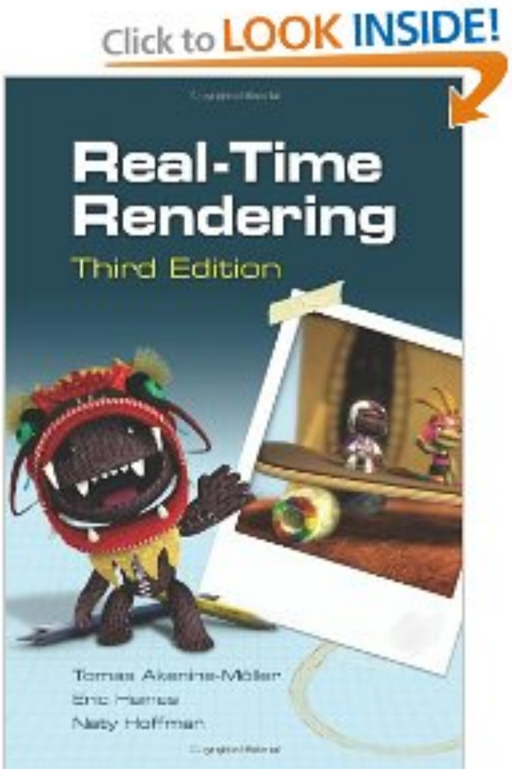
Brian Danchilla

Apress®

Reference Books

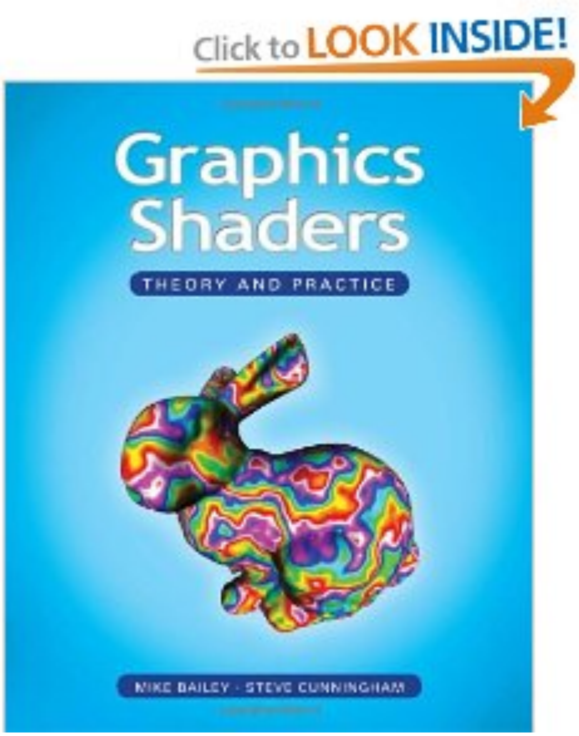

Click to LOOK INSIDE!
COMPUTER GRAPHICS
PRINCIPLES AND PRACTICE
THIRD EDITION
JOHN F. HUGHES · ANDRIES VAN DAM · MORGAN MCGUIRE
DAVID F. SKLAR · JAMES D. FOLEY · STEVEN K. FEINER · KURT AKELEY

Computer Graphics
Principle and Practice
3rd edition



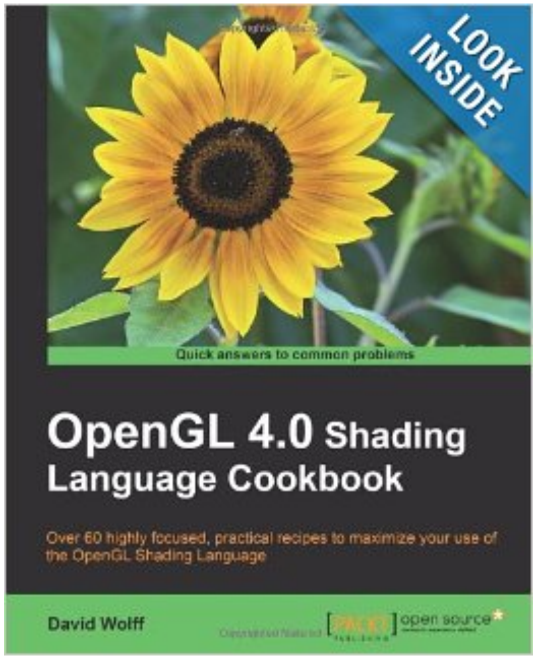
Real-Time Rendering
3rd edition

Reference Books



Safari Books Online

Graphics shaders: Theory and Practice, 2nd edition


LOOK INSIDE
Quick answers to common problems
OpenGL 4.0 Shading Language Cookbook
Over 60 highly focused, practical recipes to maximize your use of the OpenGL Shading Language
David Wolff
PACKT
open source

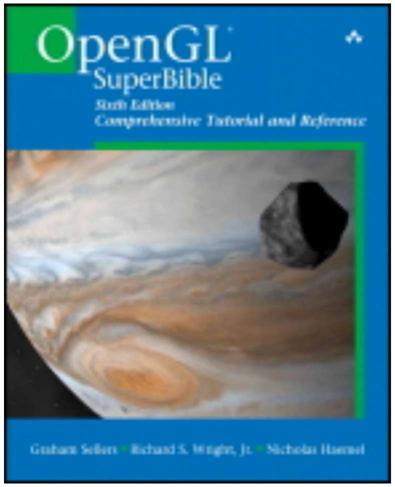
Safari Books Online

OpenGL 4.0 Shading Language Cookbook

Reference - Und Others

Quick answers to common problems
OpenGL Development Cookbook
Over 40 recipes to help you learn, understand, and implement modern OpenGL in your applications
Muhammad Mobeen Movania
PACKT

Safari Books Online


OpenGL
SuperBible
Sixth Edition
Comprehensive Tutorial and Reference
Graham Sellers • Richard S. Wright, Jr. • Nicholas Haemel

Safari Books Online

Community Experience Distilled
GLSL Essentials
Enrich your 3D scenes with the power of GLSL!
Jacobo Rodriguez
PACKT

Safari Books Online

Reference

Internet Tutorials

your basic guide to the Internet

What do we Study ?

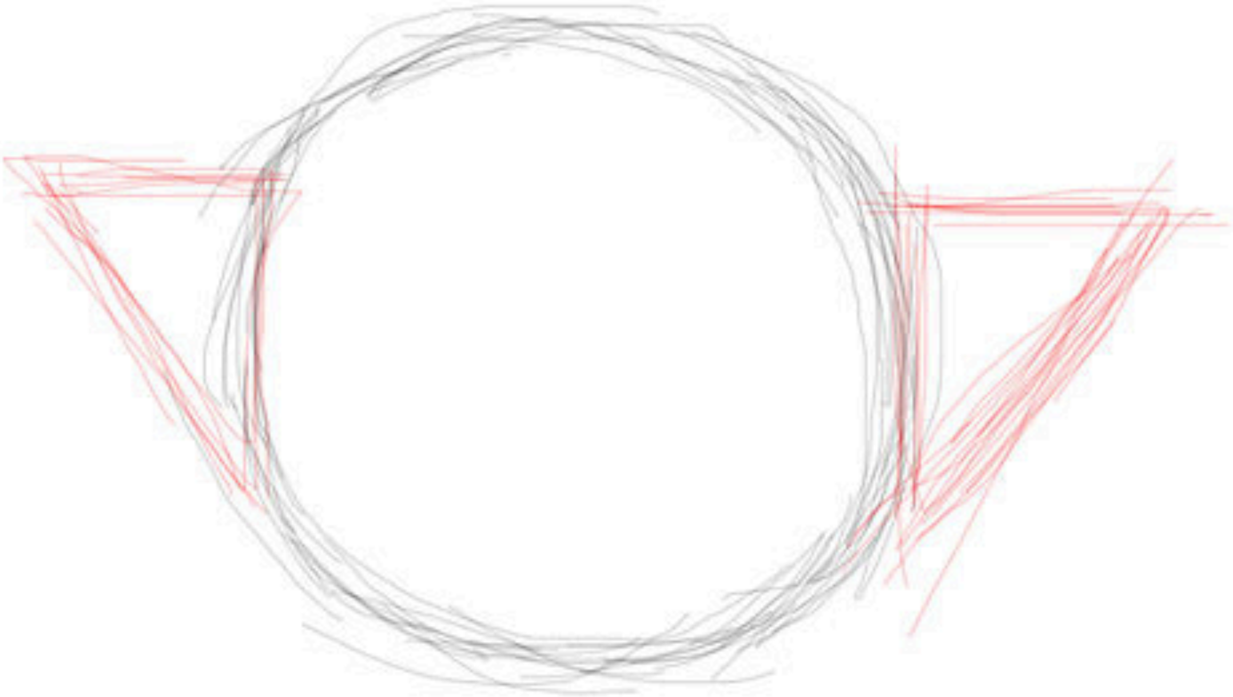
Teapots

A Real One

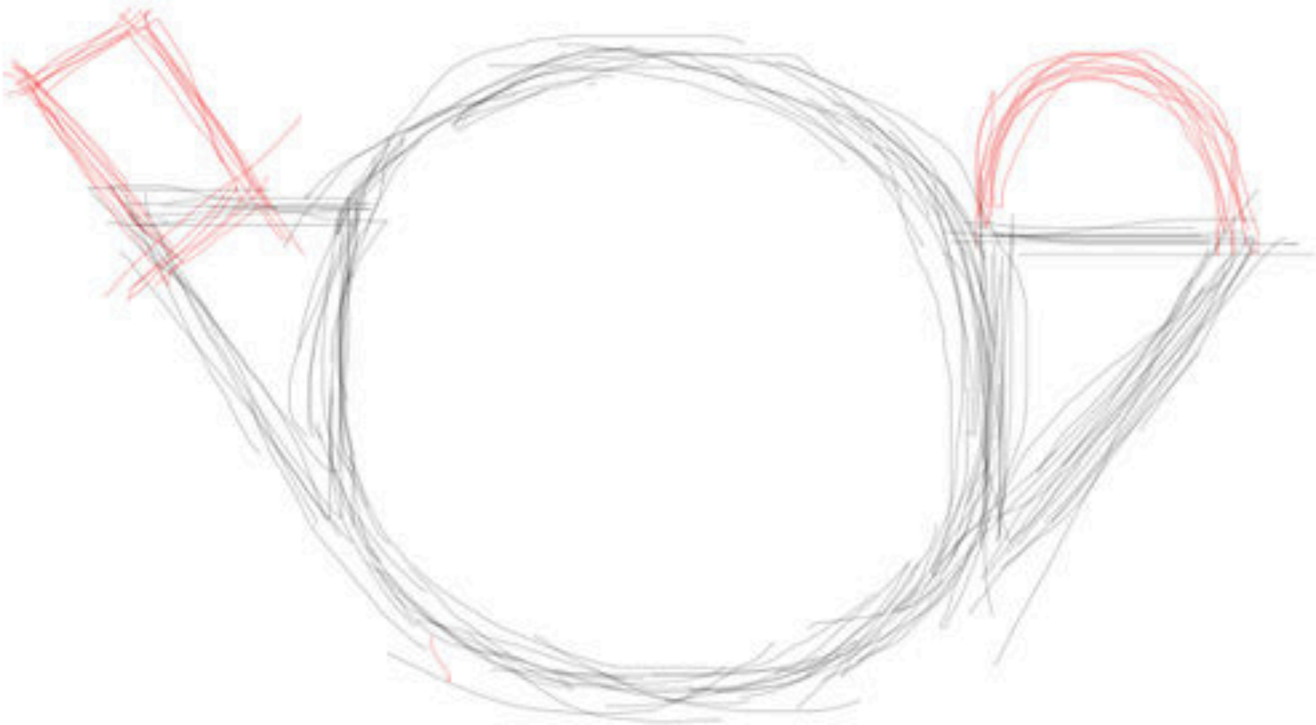
Drawing Teapots

Step 1

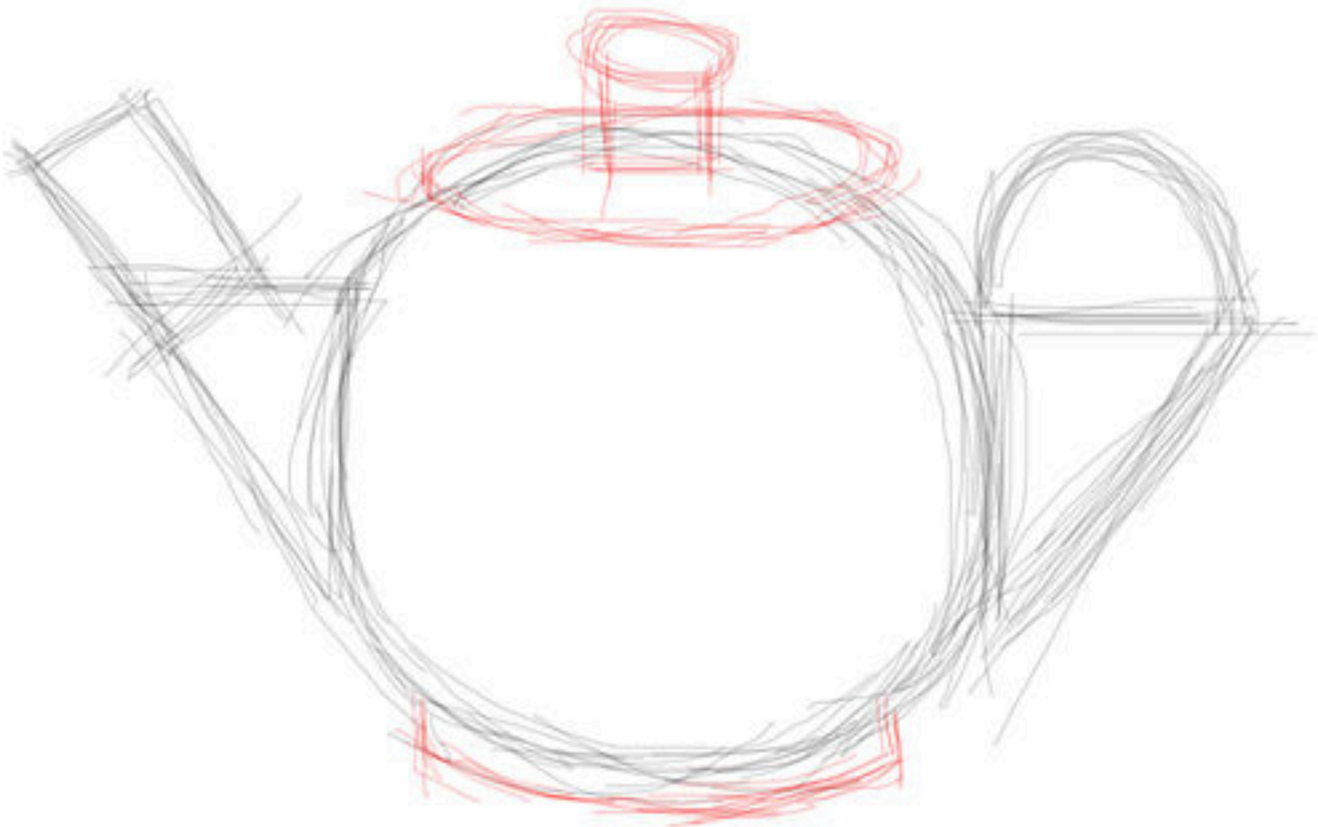
Step 2



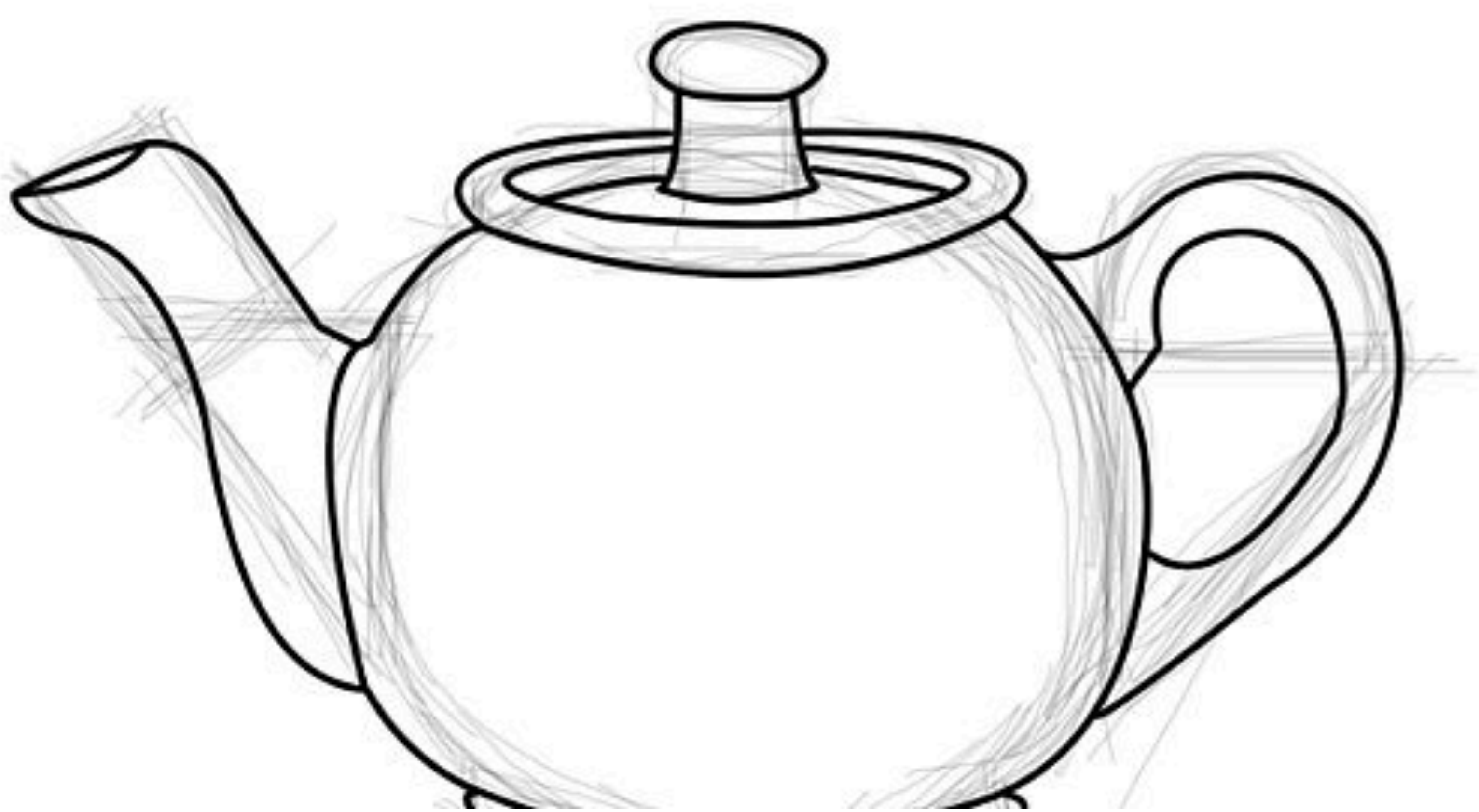
Step 3



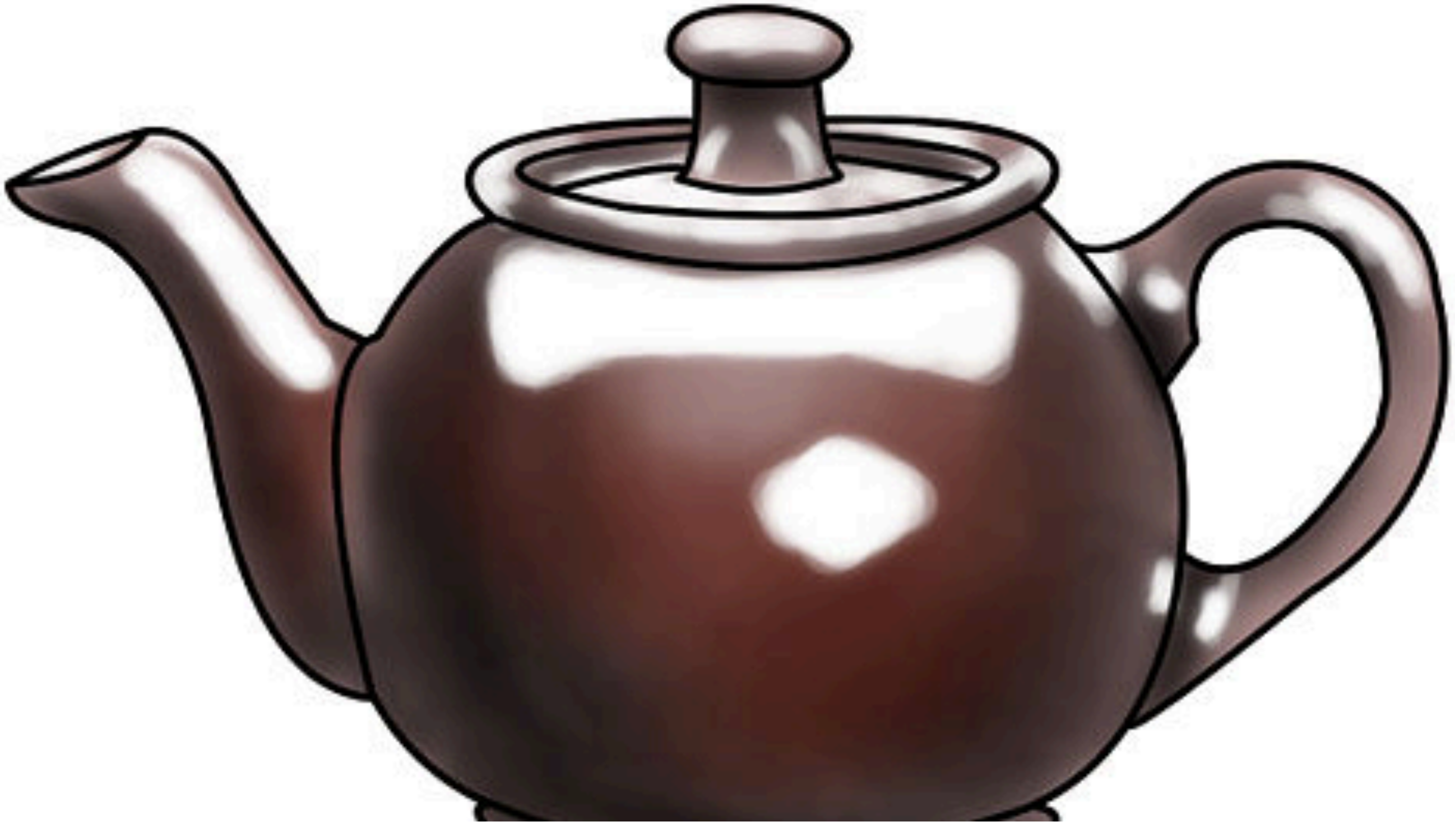
Step 4



Step 5



Step 6



Manufacturing Teapots & Minions

Manufacturing Teapots

http://oncampus.osu.edu/replicating-in-3d/

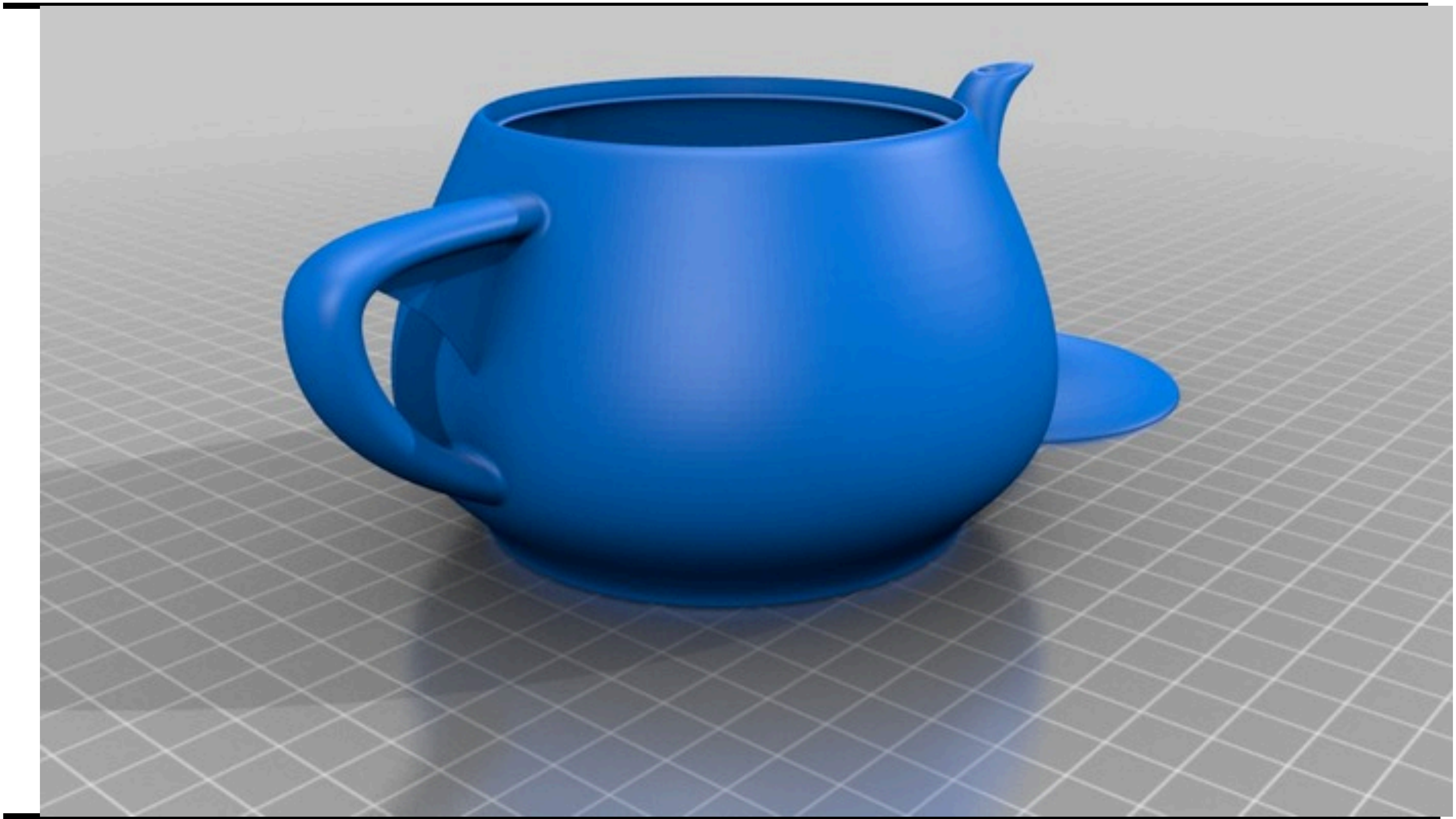
https://www.ted.com/talks/lisa_harouni_a_primer_on_3d_printing

http://news.cornell.edu/stories/2013/04/rapid-reality-students-design-3-d-printed-products

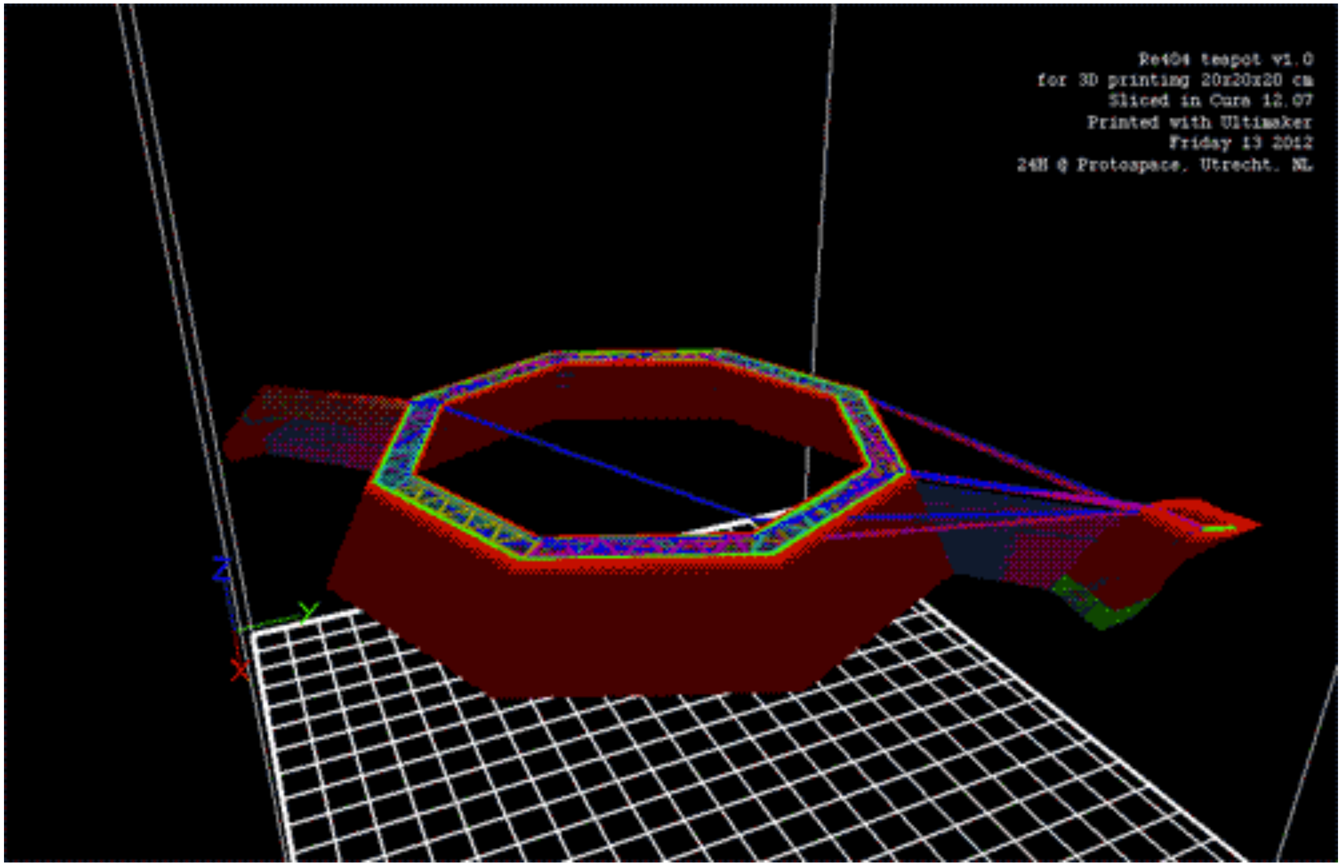
You will all make these ☺

http://www.thingiverse.com/thing:68880

http://www.thingiverse.com/thing:68880



Manufacturing Teapots



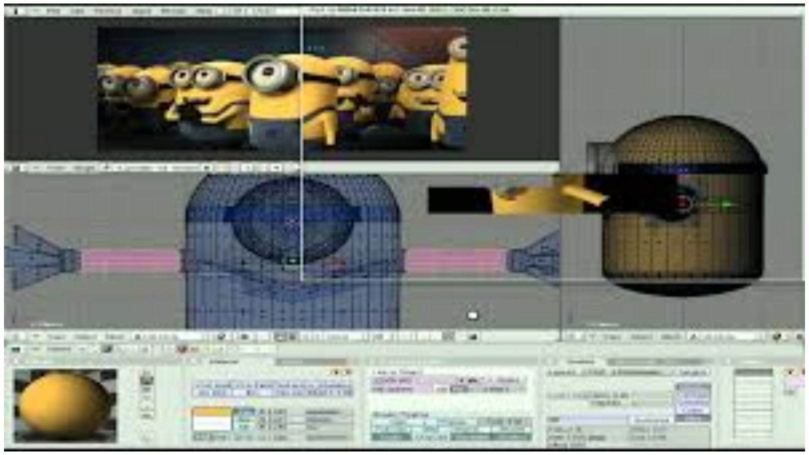
http://onderin.de.buro.la/cura-ani-500-64c.gif

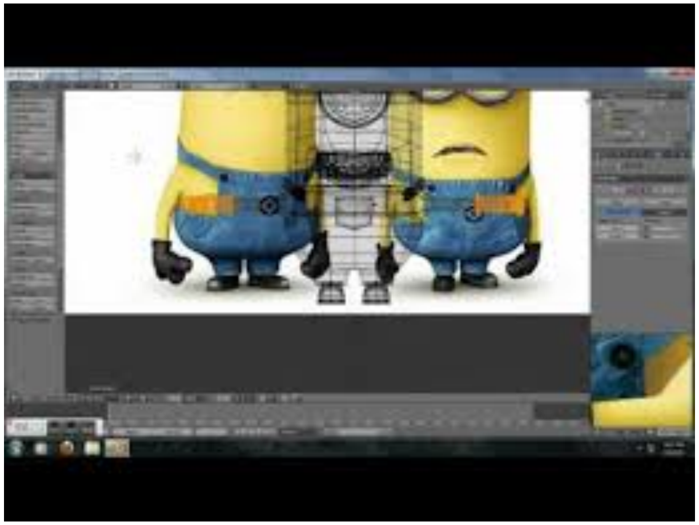
And These

MEET
STUART
KEVIN
&BOB
I'M WITH
STUPID
ILLUMINATION
minions
SUMMER 2015

Maybe Not This

Manufacturing Minions





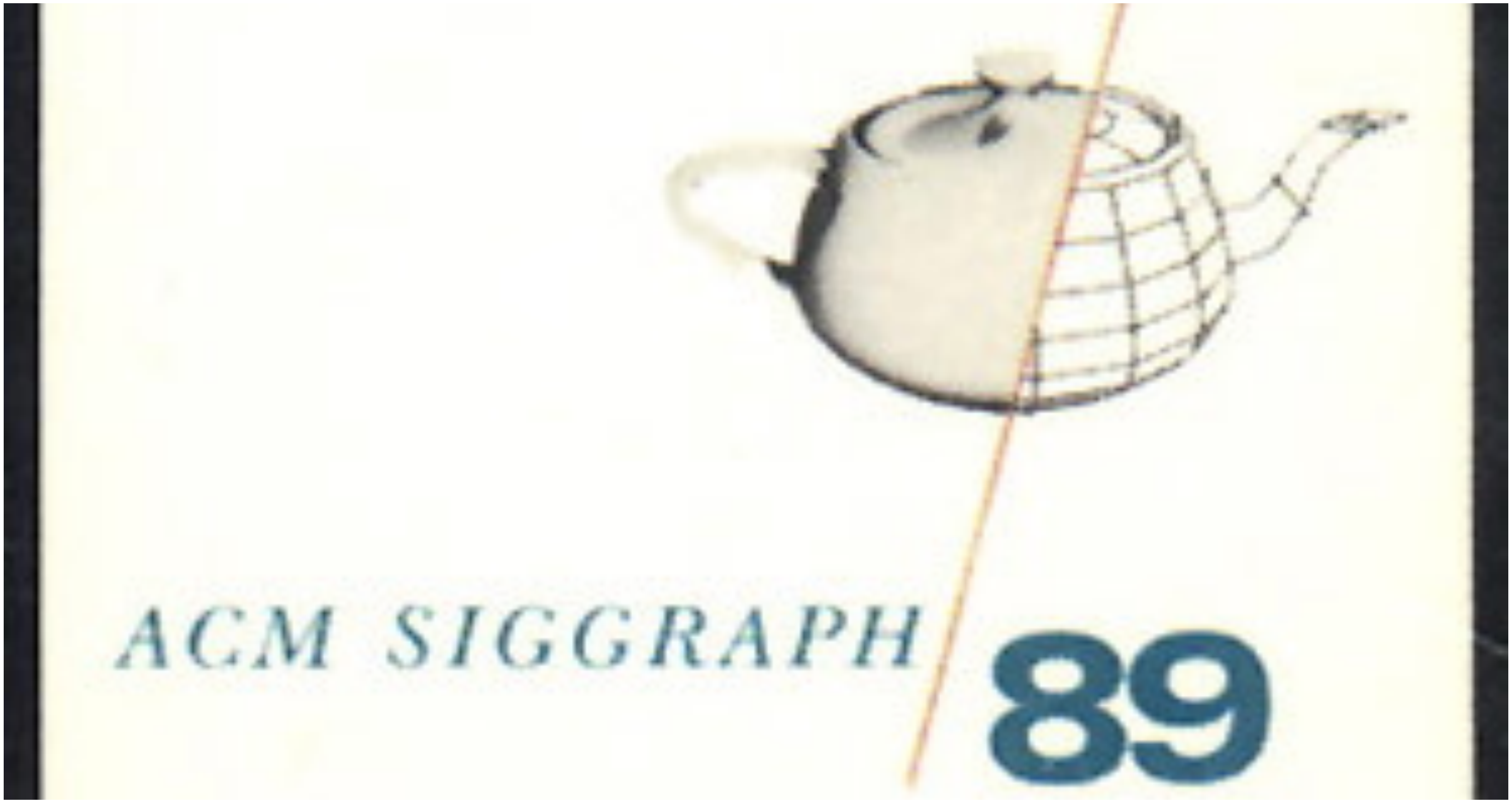
https://www.youtube.com/watch?v=-2uY7rjhhMs

We will not do this …

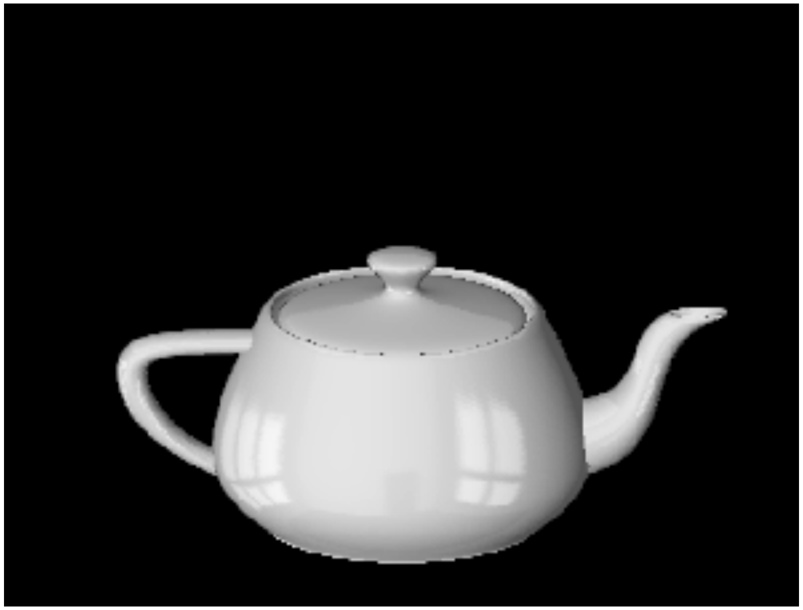
Kevin Wolf, 3D Printer, Mechanical Eng., OSU

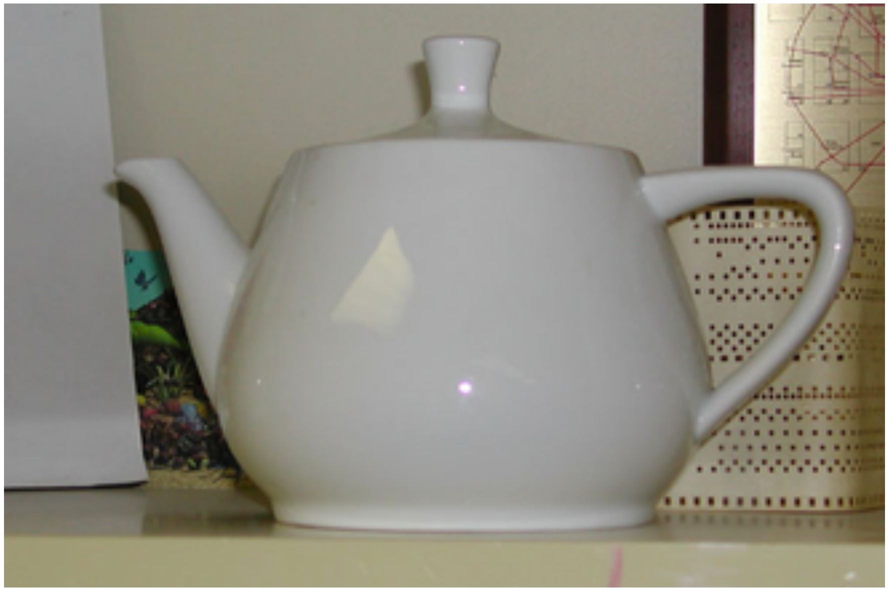
Virtual Teapots

An Icon !

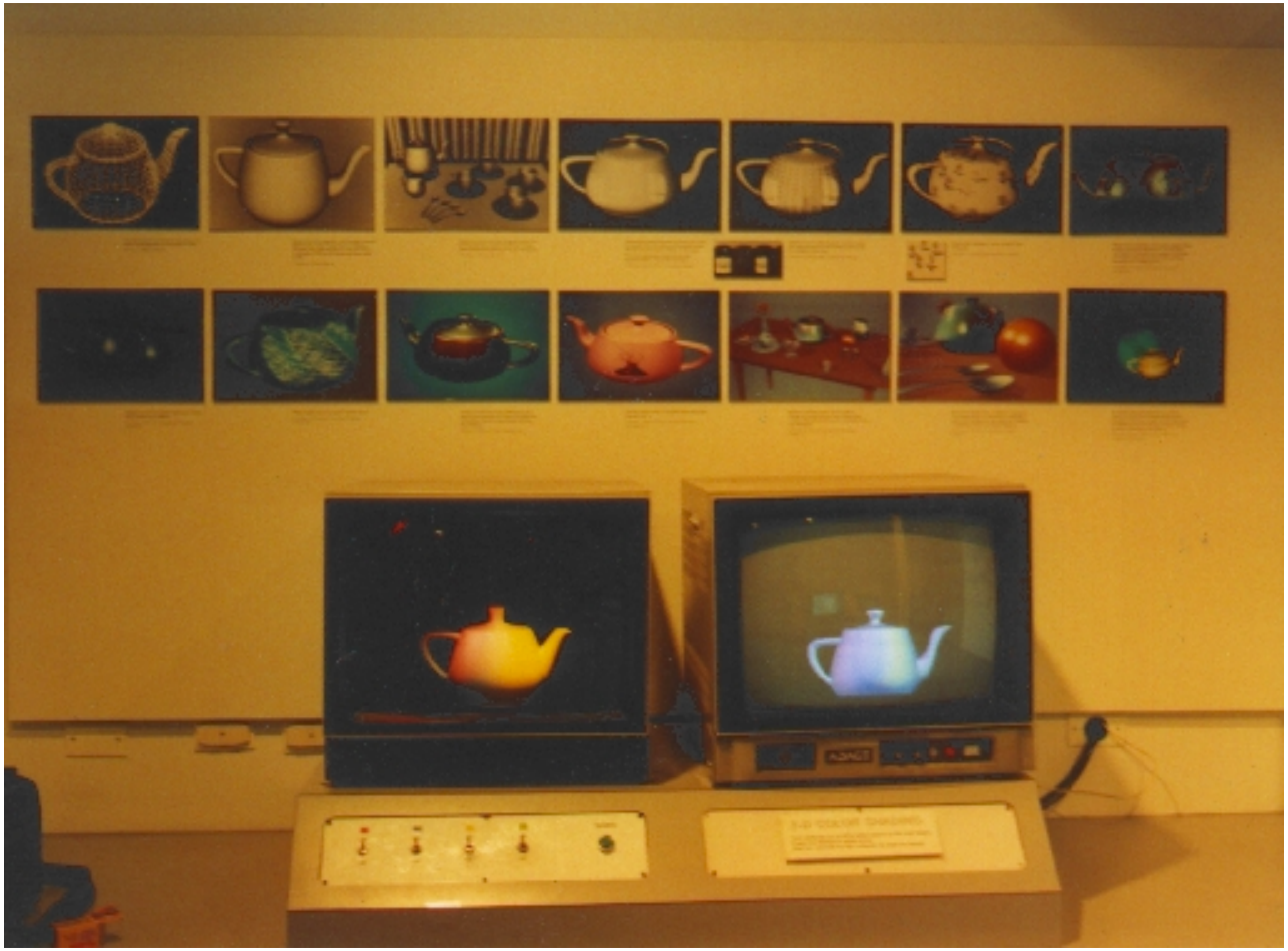


Which one is real ?





Real vs Virtual – Boston Museum



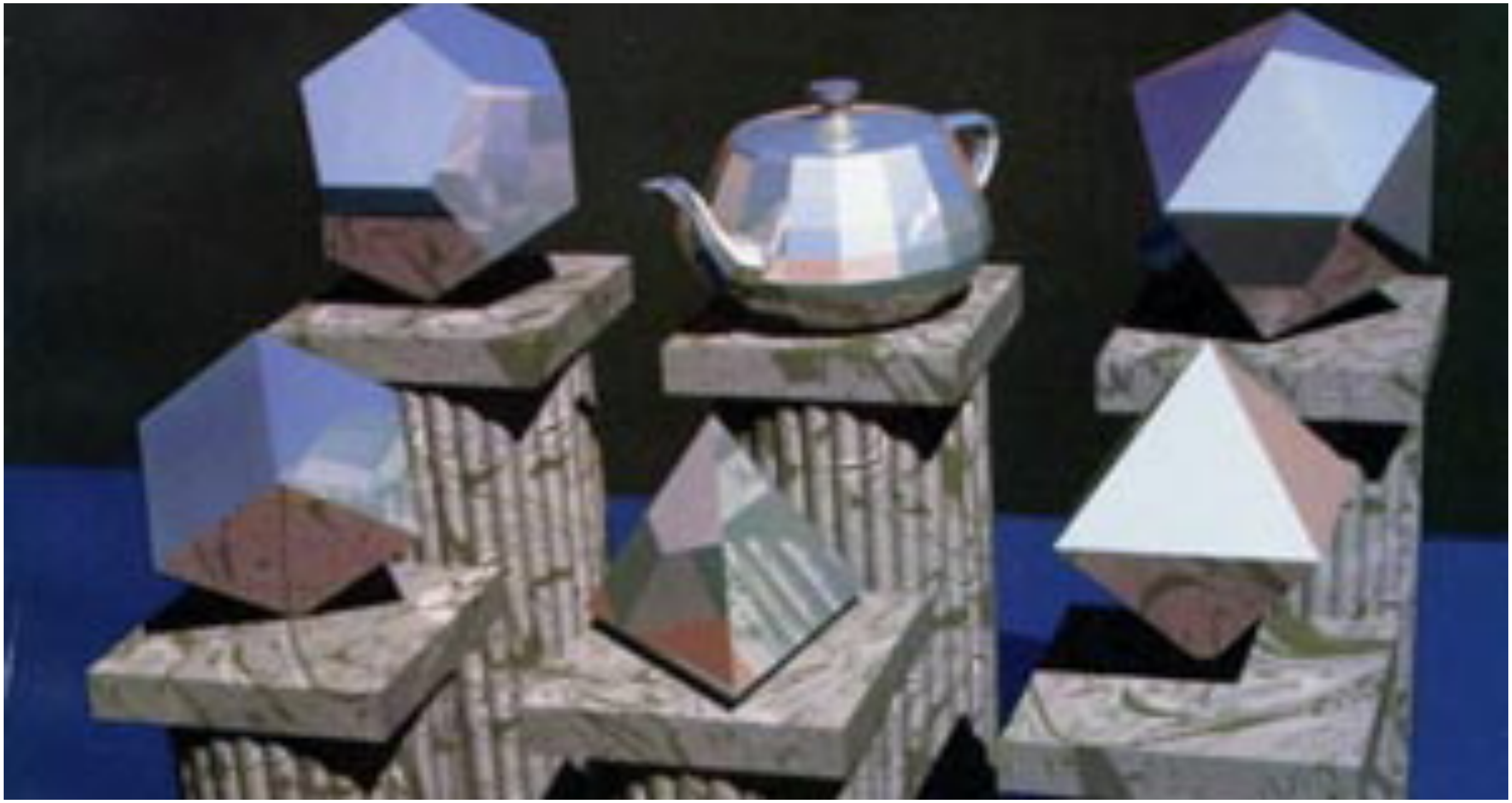
More fake ones !

http://codegolf.stackexchange.com/questions/22620/draw-the-utah-teapot

Shutterbug !

Toy Story

A Platonic Relationship !



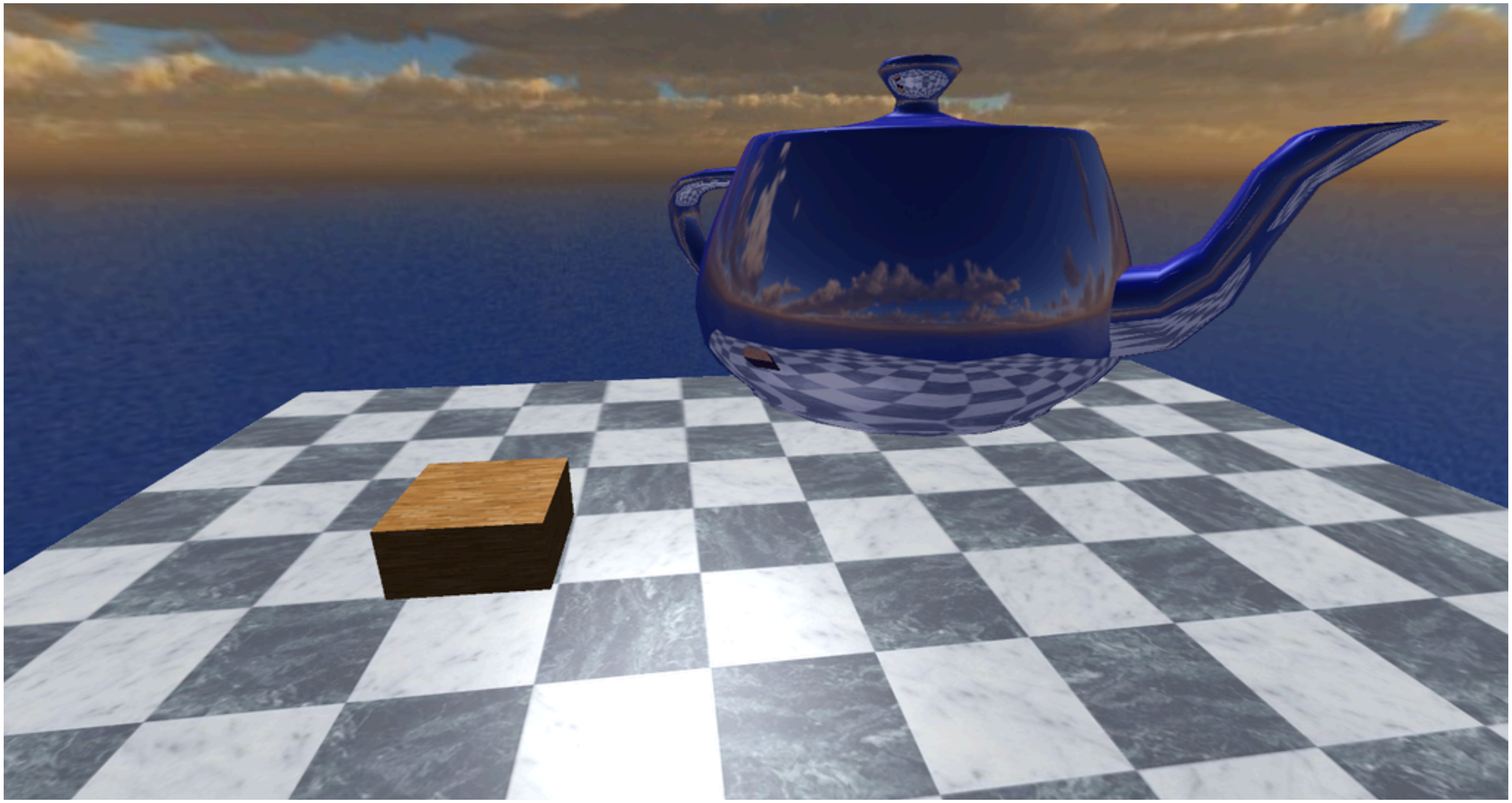
Building with Blender

https://www.youtube.com/watch?v=QyiBwL2Scec

Another Method

https://www.youtube.com/watch?v=x8AiEi4aJ4g

You will be also doing this …



And better than this !

OpenGL, GLSL

OpenGL, GLSL

Topics and Outcomes

To Reiterate

Geometric Modeling

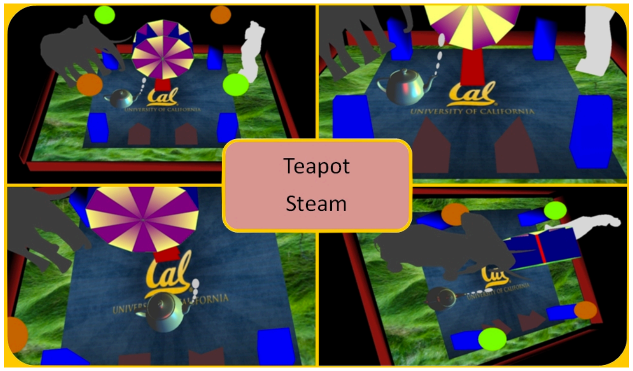
blender™

Software Tools

3D Printed Version

WebGL™

GLSL



Learning Outcomes - Familiarity

- Basic understanding of graphics hardware/software

- Basics of modeling and 3D Printing

- Basics of interaction

Learning Outcomes - Skills

- OpenGL/GLSL to control graphics hardware

- WebGL/OpenGL-ES to allow interactions

- Blender to build models

Learning Outcomes - Cognitive

- Advanced real time rendering algorithms

- Integrating three different software suites – WebGL, GLSL, OpenGL

- Solid and curve modeling nuances

Specific Topics

- Overview of Graphics Hardware and Software
- Coordinate systems
- WebGL Interaction modes
  - HTML 5
  - Immediate vs. retained mode
- OpenGL geometry drawing
  - OpenGL vertex buffer objects
- OpenGL Shading Language
  - Vertex and fragment shaders
- 3D transformation and Viewing

Specific Topics

- Illumination
  - – Flat, Gourad, Phong shading models
  - – Fixed function pipeline and shaders
- Visibility and Z-buffering
- Texture Mapping
  - – Image and procedural textures
- Bump, environment, & projective texture mapping

Specific Topics

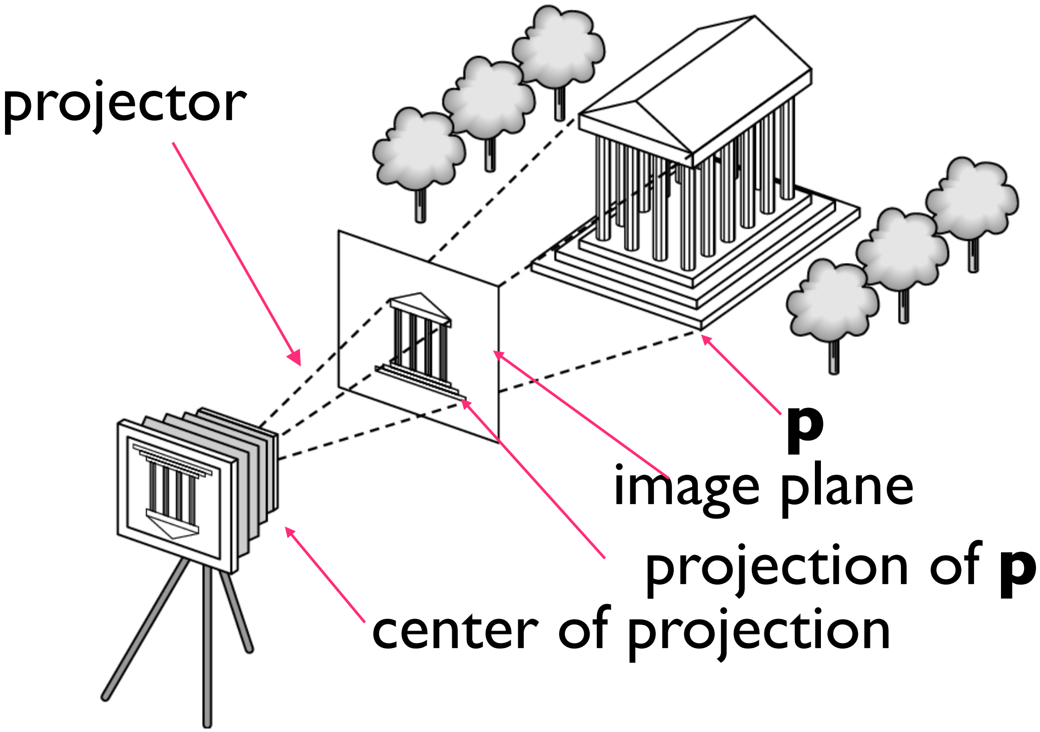
- Real time shadows
- Particle Methods
- Advanced topics in shaders
  - Geometry shader
  - Tessellation shader
- Advanced topics in rendering and graphics

Image Formation

Image Formation

- – Cameras
- – Microscopes
- – Telescopes
- – Human visual system

Synthetic Camera Model



projector

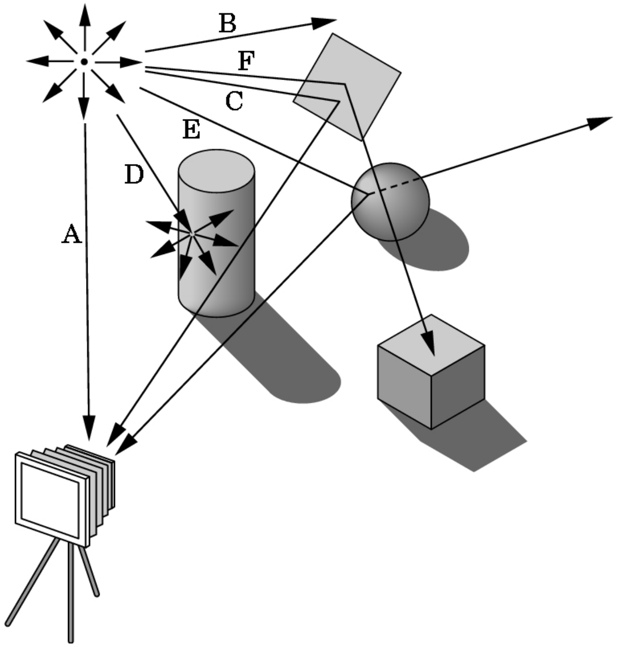
p

image plane

projection of **p**

center of projection

Image Formation (Eine Explanation)



Ray Tracing and Geometric Optics

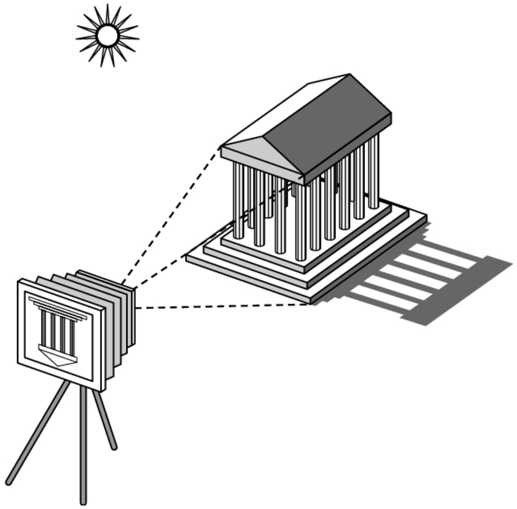
Problemo ?

The Real Thing !

http://www.uccs.edu/~rtirado/Astronomy_Texts/Light_Image_Formation.pdf

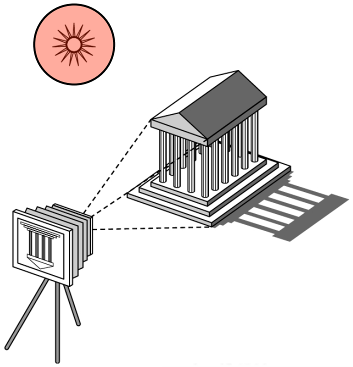
Essentials of Image Formation

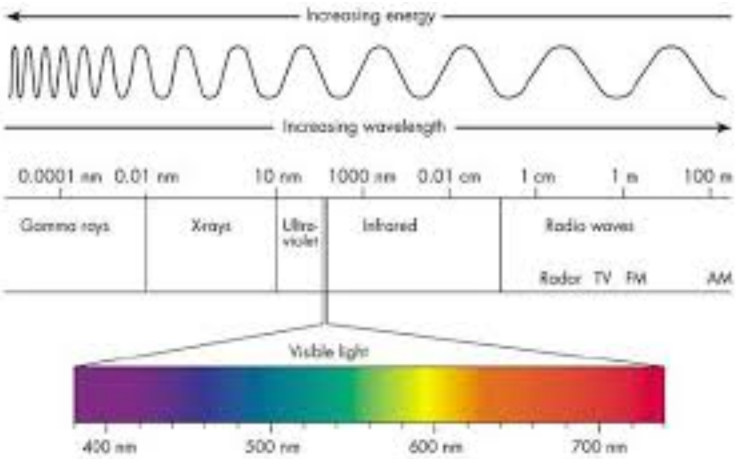
- Objects
- Camera
- Light source(s)

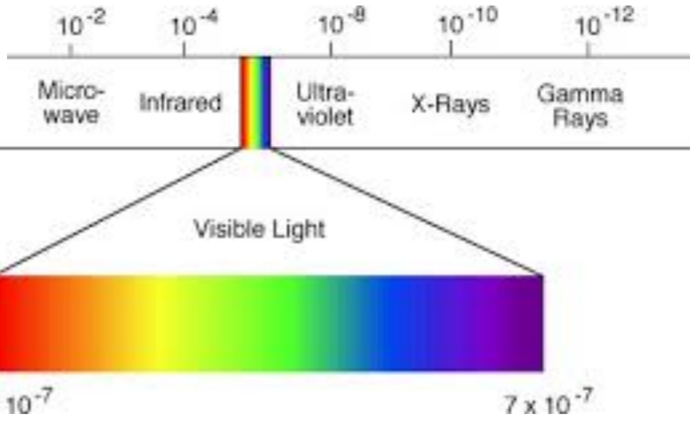


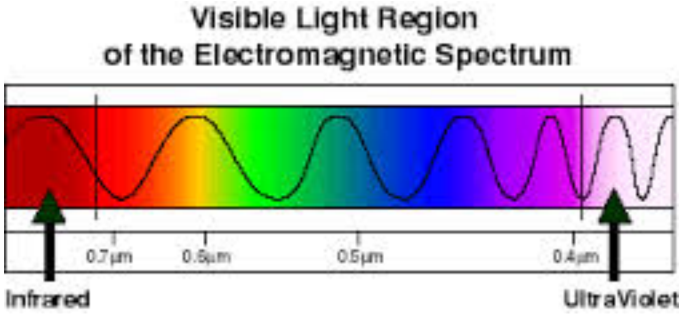
- Light-material interaction
- Independence of objects, viewer, and light source(s)

Light

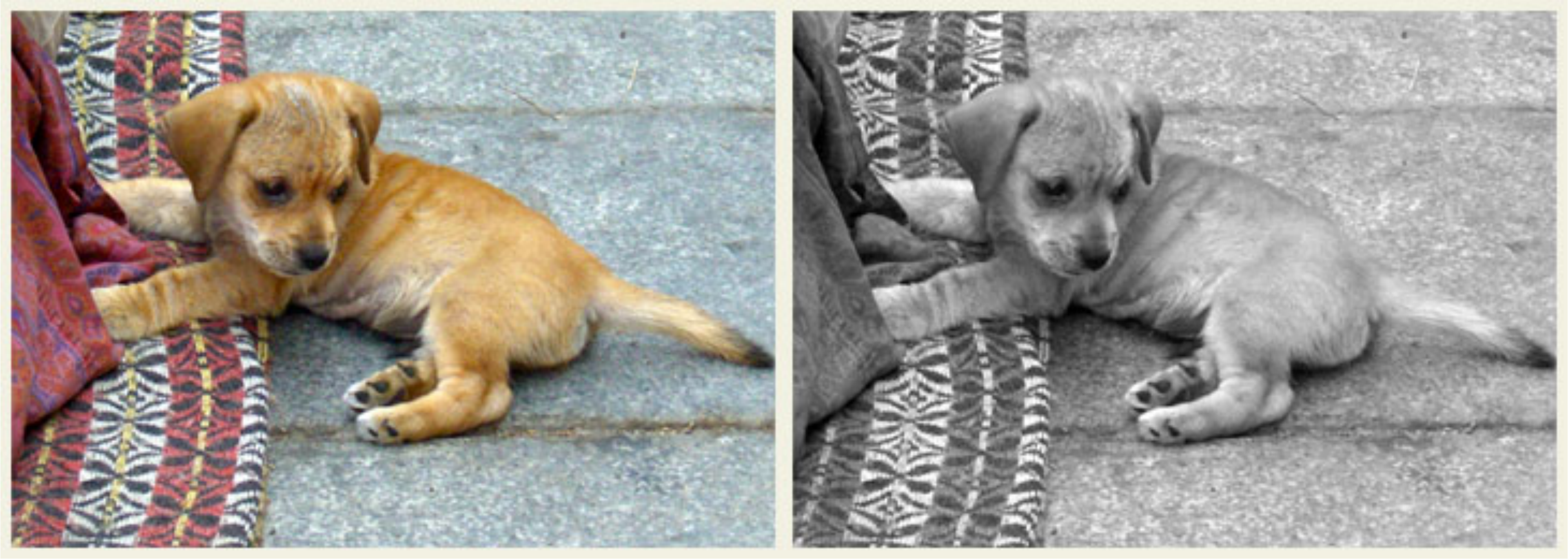



Increasing energy
Increasing wavelength
0.0001 nm
0.01 nm
10 nm
1000 nm
0.01 cm
1 cm
1 m
100 m
Gamma rays
X-rays
Ultra-violet
Infrared
Radio waves
Radar
TV
FM
AM
Visible light
400 nm
500 nm
600 nm
700 nm


10^{-2}
10^{-4}
10^{-8}
10^{-10}
10^{-12}
Micro-wave
Infrared
Ultra-violet
X-Rays
Gamma Rays
Visible Light
4×10^{-7}
7×10^{-7}


Visible Light Region
of the Electromagnetic Spectrum
0.7μm
0.6μm
0.5μm
0.4μm
Infrared
UltraViolet

Luminance vs. Color

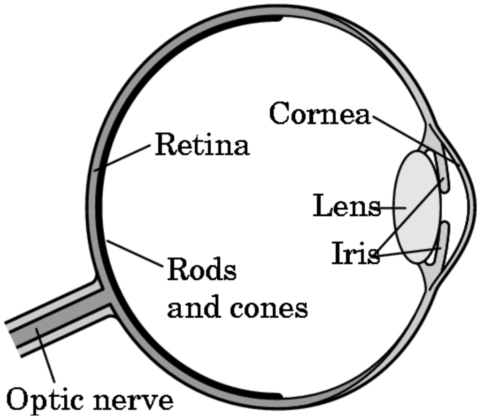


http://www.workwithcolor.com/color-luminance-2233.htm

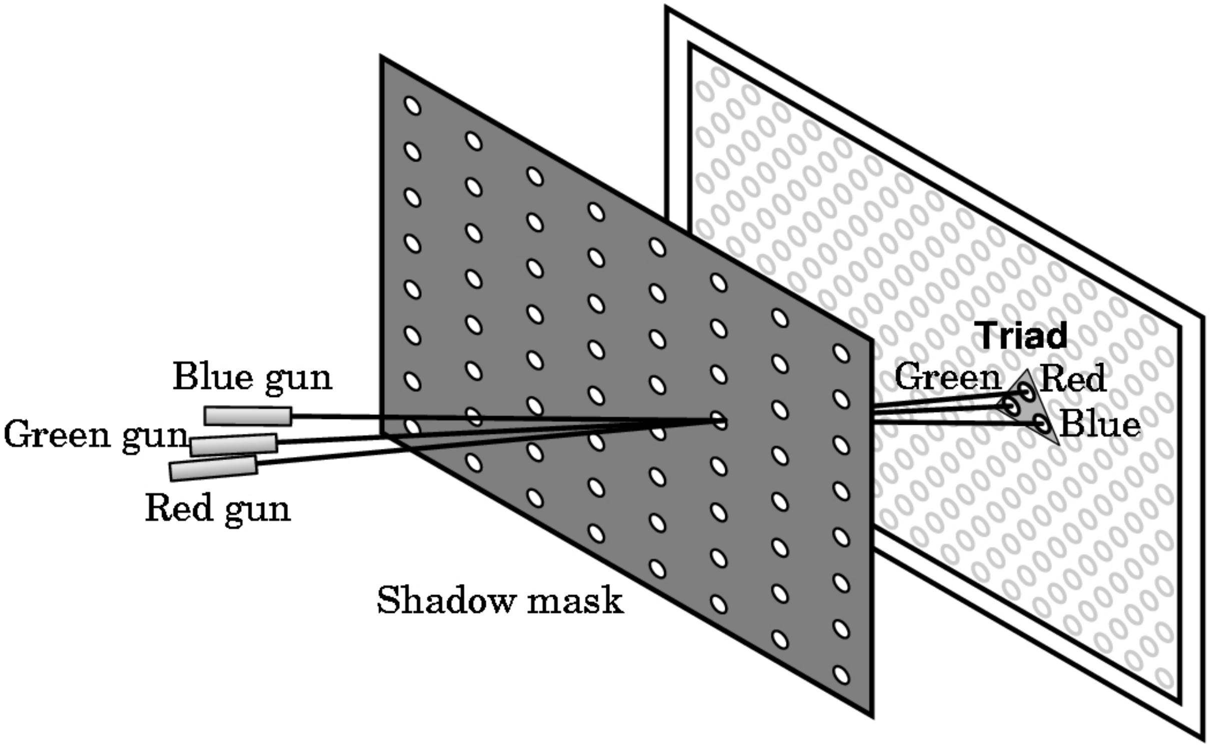
How Many Colors ?

Human visual system

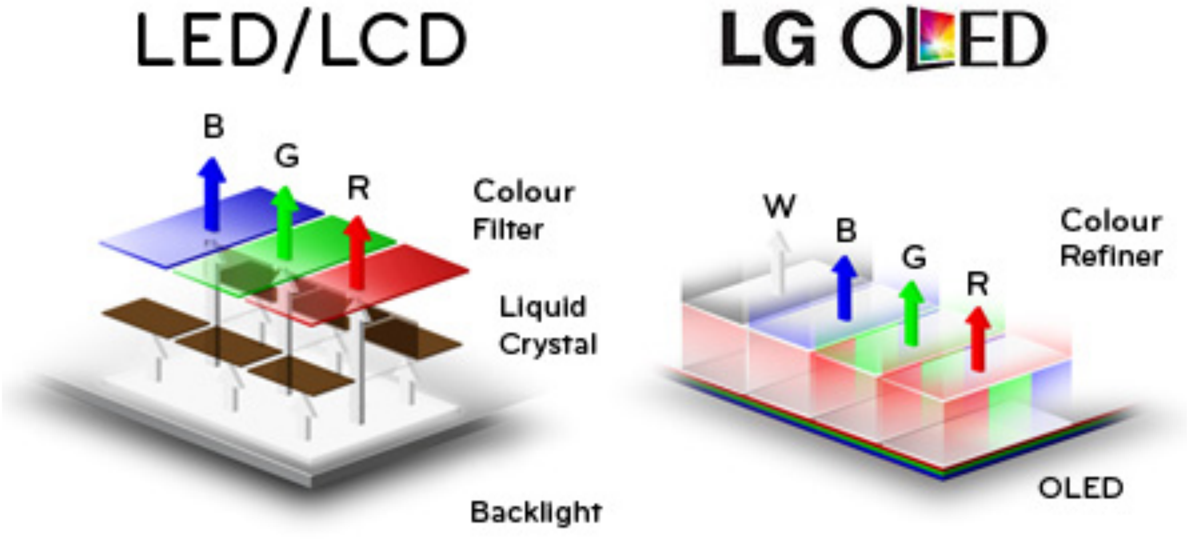
- Rods: monochromatic, night vision
- Cones
  - Color sensitive
  - Three types of cones
  - Only three values (the *tristimulus* values) are sent to the brain
- Three *primary* colors – R, G, B



In Days Long Gone



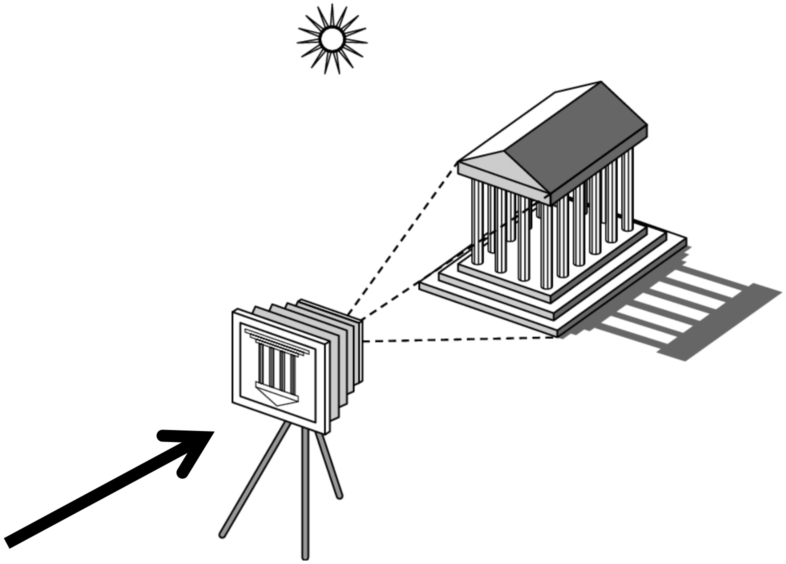
Even in these days !

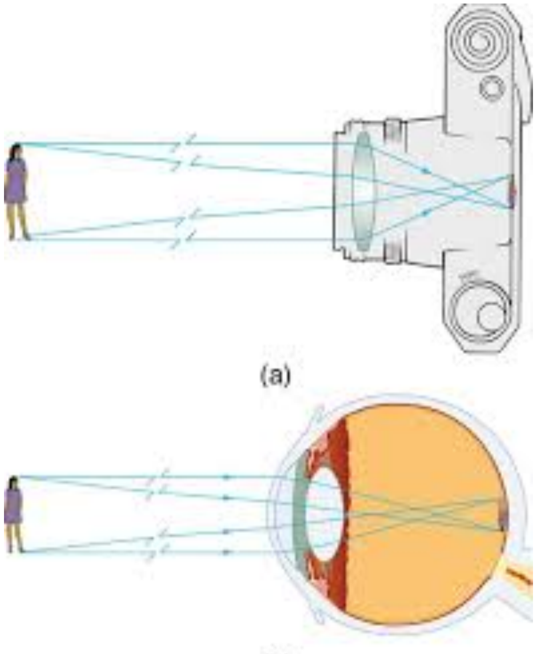


- Complex Structure
- BLU (Backlight Unit) CCFL, LED
- Lighting Unit = Pixel Unit

- Simple Structure
- Self-emissive
- Lighting Unit = Pixel Unit

The Camera

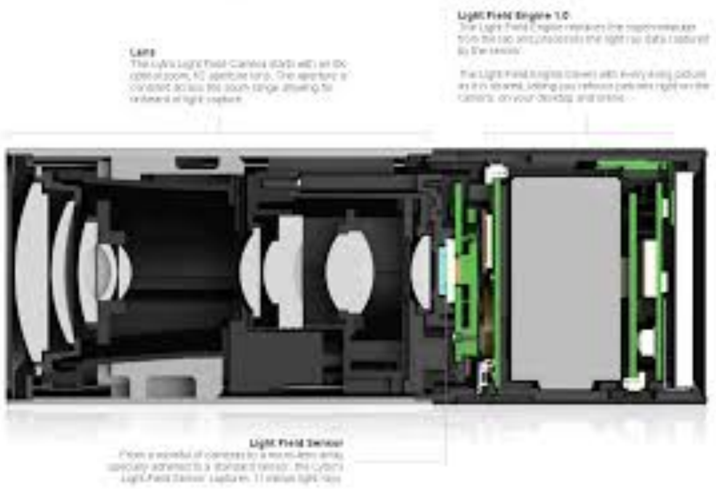


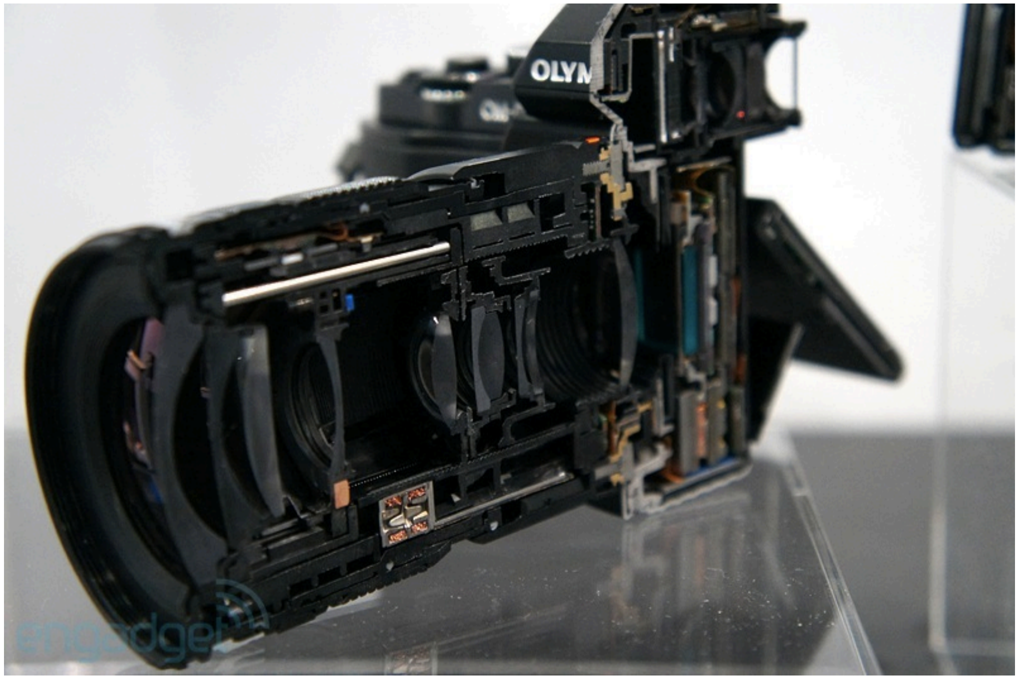


(a)

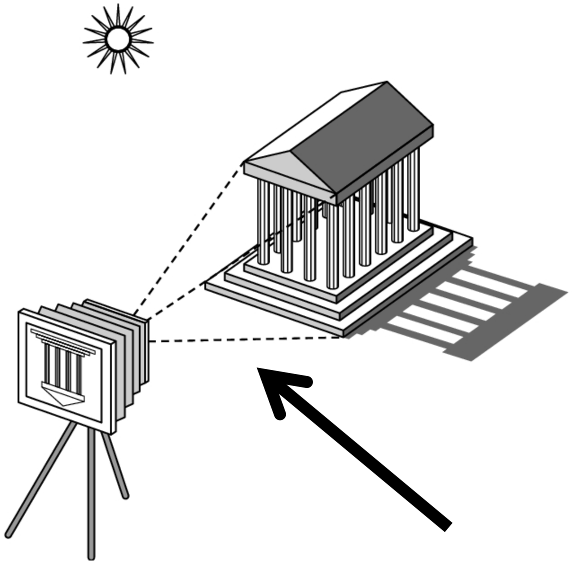
(b)

Real Cameras

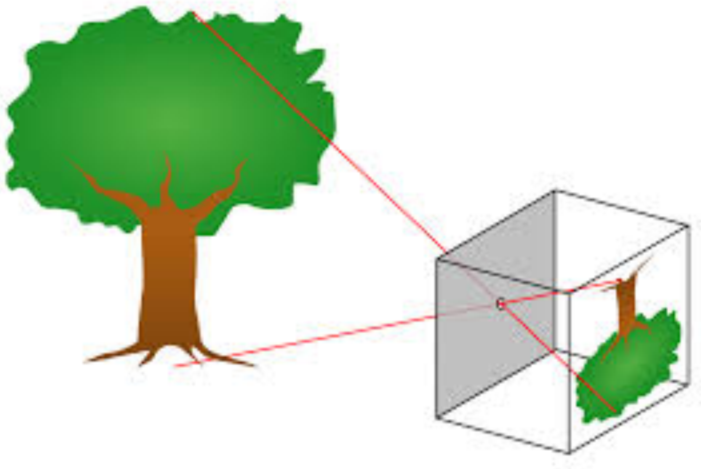

Lens
The Lytro's Light Field Camera starts with an 8x optical zoom, f/2 aperture lens. The aperture is constant across the zoom range allowing for unheard of light capture.
Light Field Engine 1.0
The Light Field Engine replaces the supercomputer from the lab and processes the light ray data captured by the sensor.
The Light Field Engine travels with every living picture as it is shared, letting you refocus pictures right on the camera, on your desktop and online.
Light Field Sensor
From a wheelful of cameras to a micro-lens array specially adhered to a standard sensor, the Lytro's Light Field Sensor captures 11 million light rays.

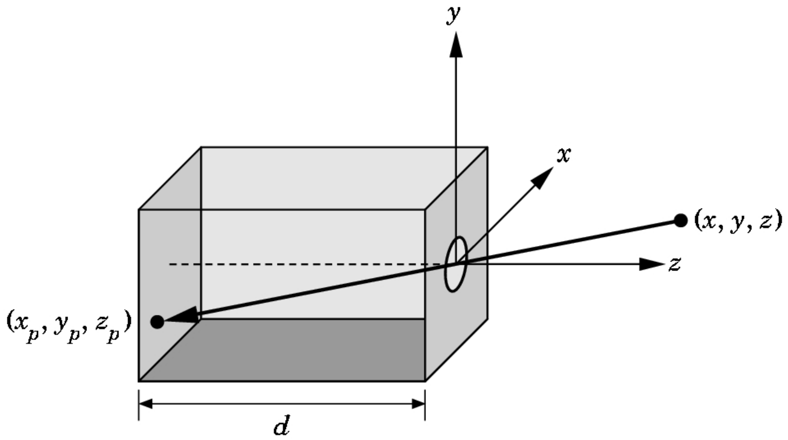


Viewing



Eine Simple Kamera – The Pinhole

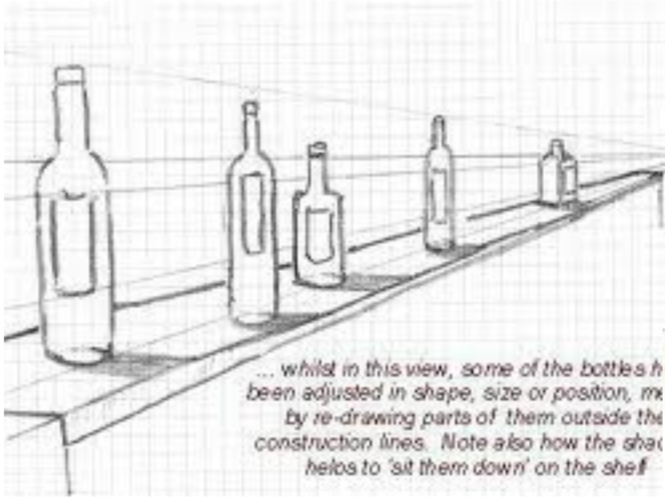




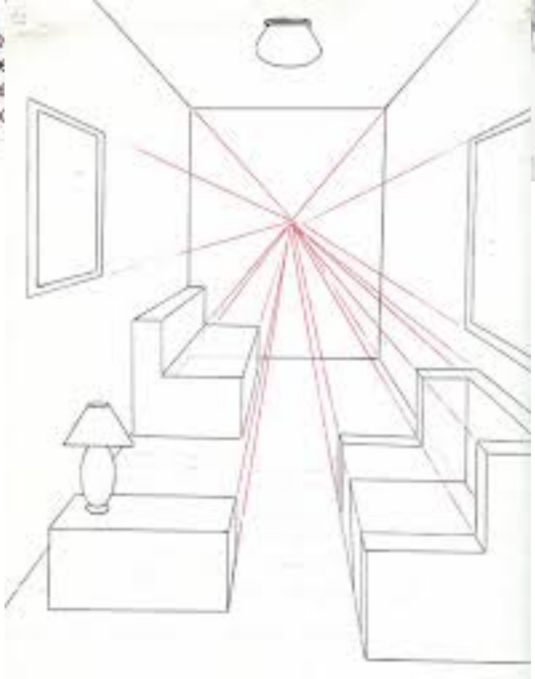
Simple Perspective - find projection of point at (x,y,z)

$x_p = -x/z/d \qquad y_p = -y/z/d \qquad z_p = d$

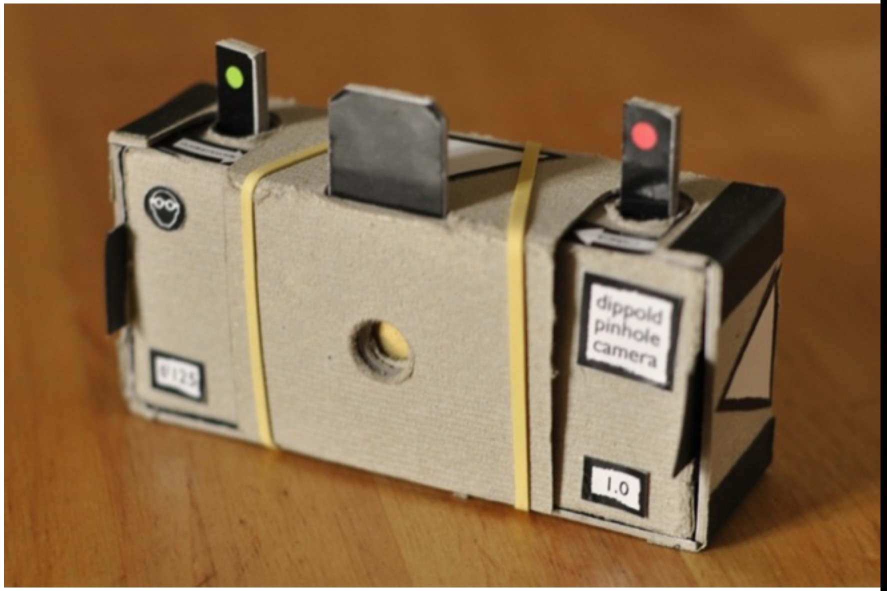
Perspective

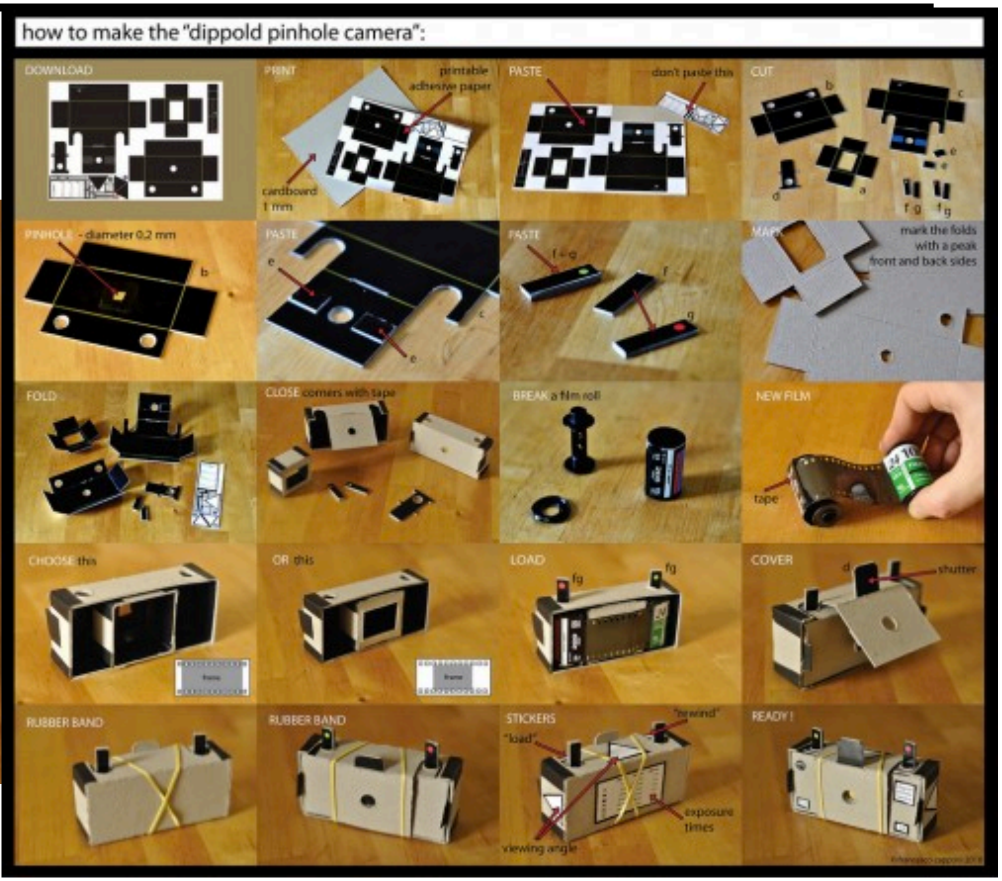


... whilst in this view, some of the bottles h
been adjusted in shape, size or position, m
by re-drawing parts of them outside the
construction lines. Note also how the shad
helps to 'sit them down' on the shelf



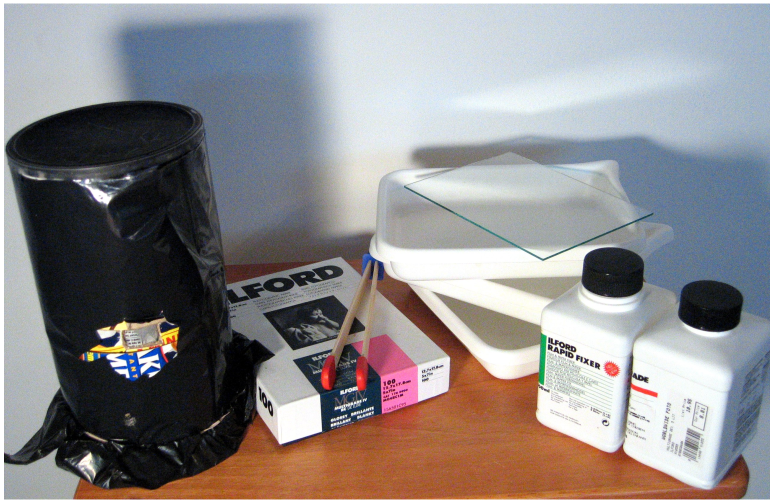
DYC



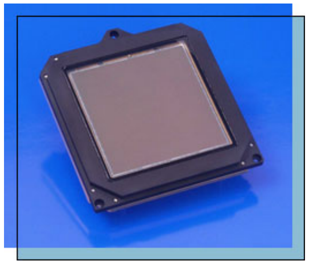


http://www.pinhole.cz/en/pinholecameras/dirkon_01.html

Not Bad ☺

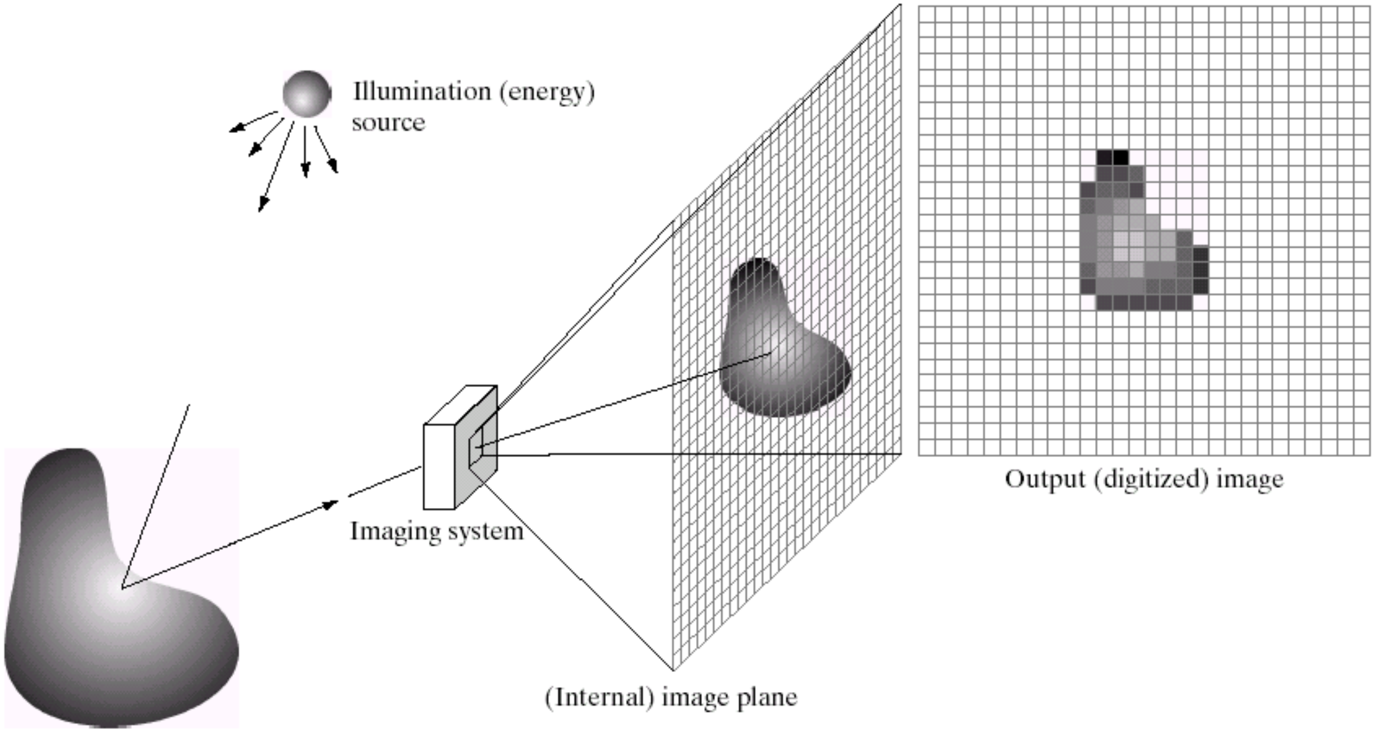


Let Us Go Digital

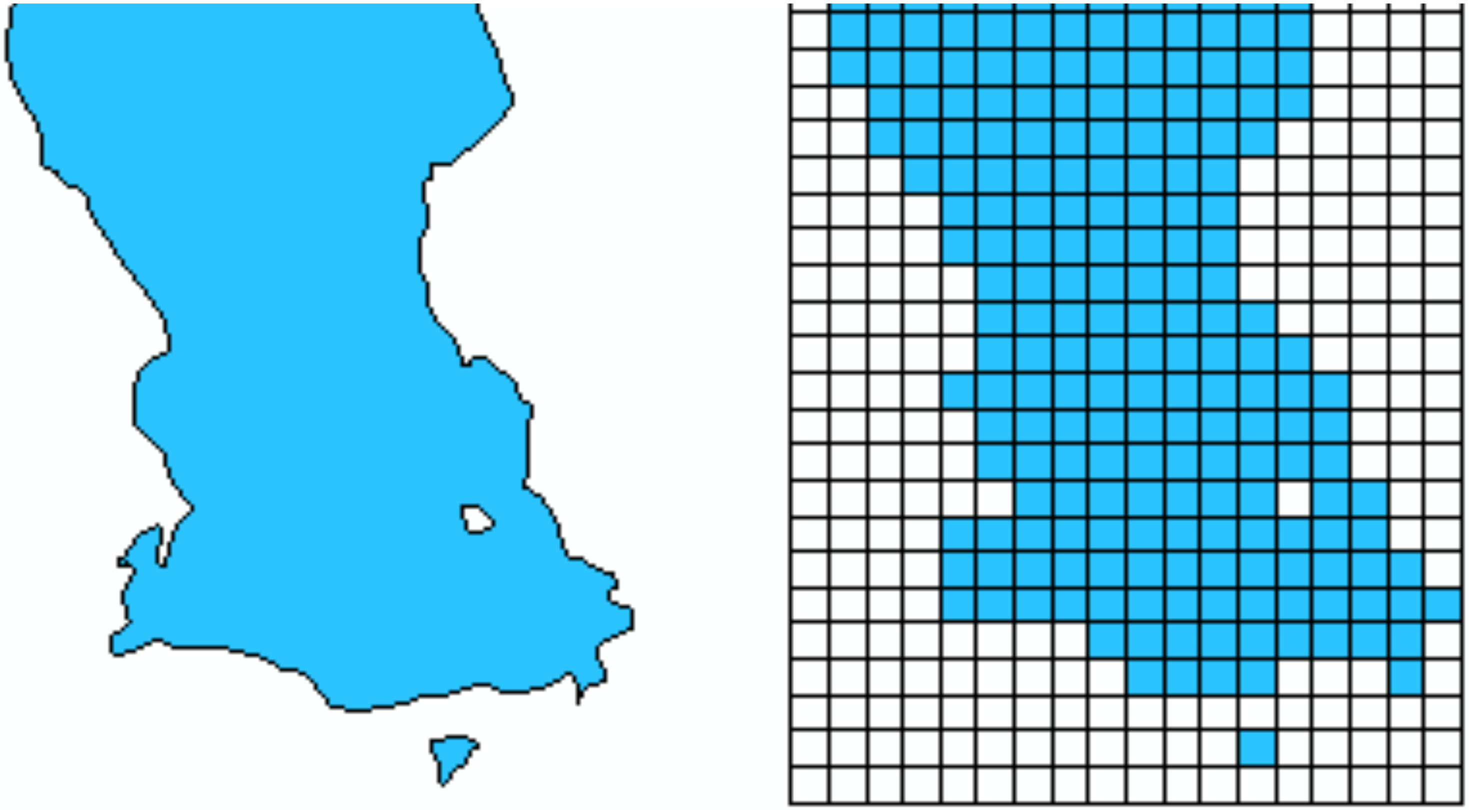


http://electronics.howstuffworks.com/digital-camera.htm

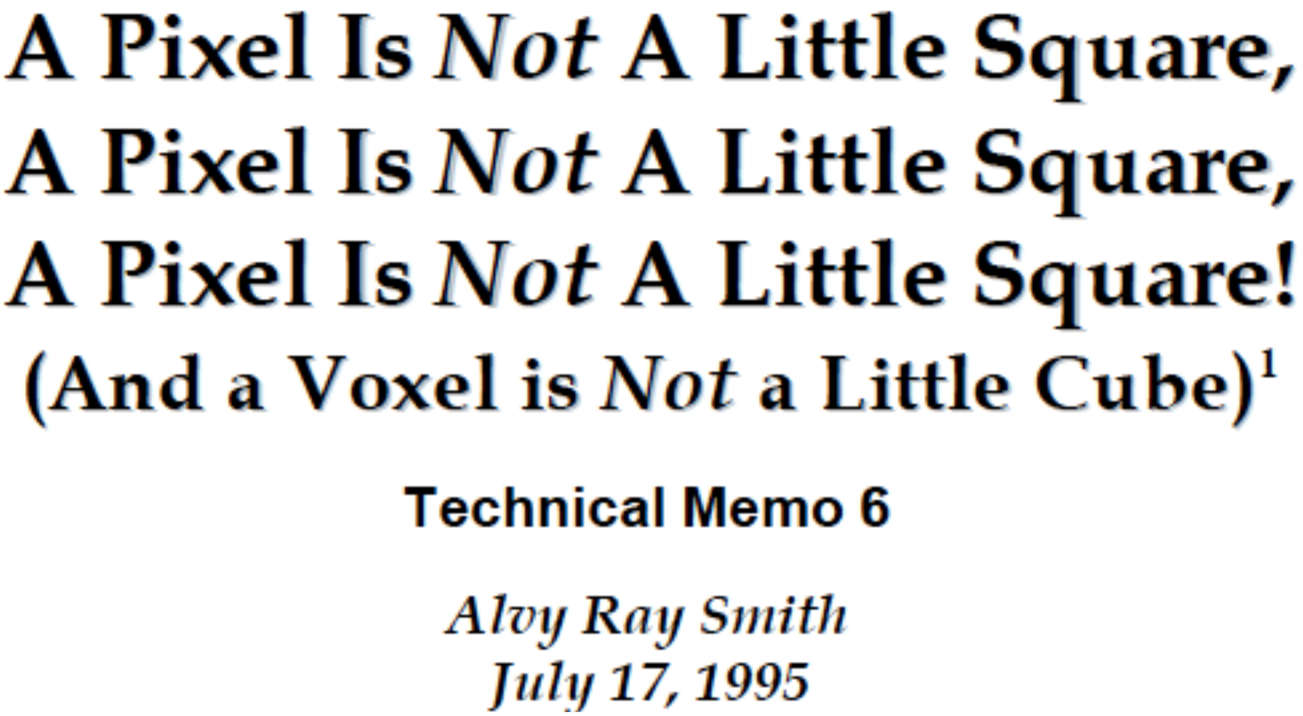
Rasters


Illumination (energy) source
Imaging system
(Internal) image plane
Output (digitized) image
Scene element

Pixels & Resolution



Is A Pixel Really A Square ?



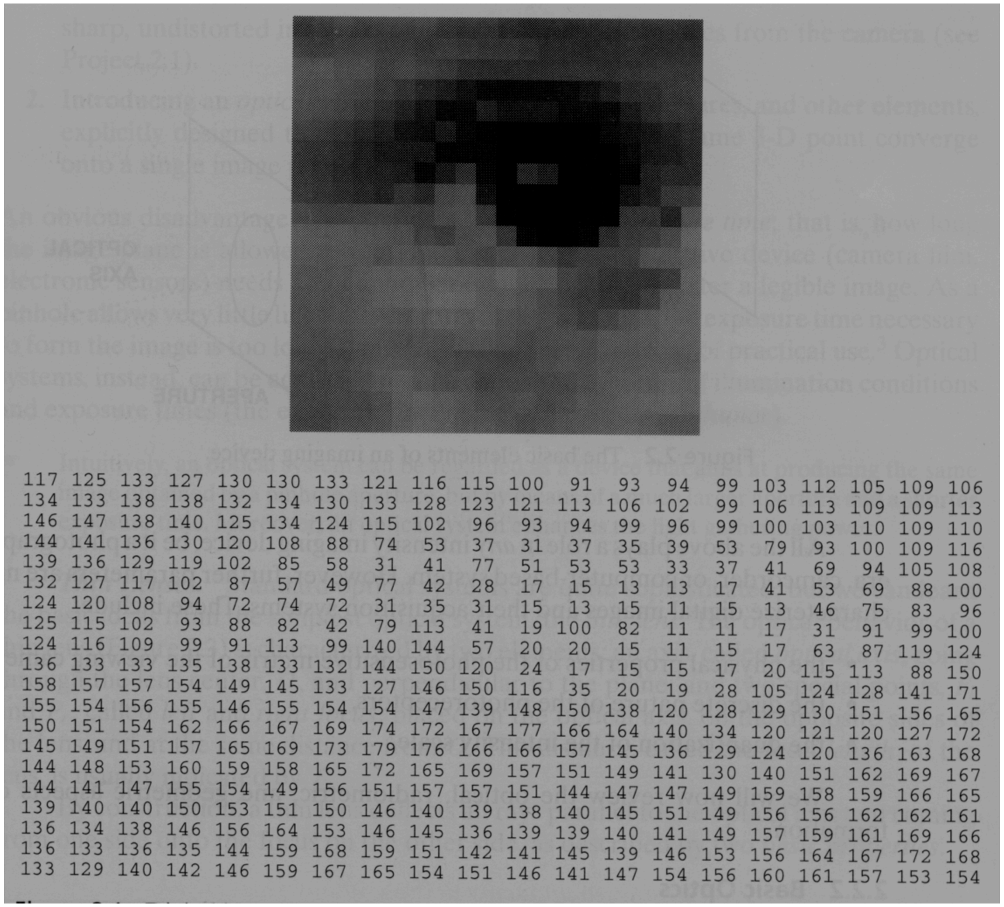
A Pixel Is *Not* A Little Square,
A Pixel Is *Not* A Little Square,
A Pixel Is *Not* A Little Square!
(And a Voxel is *Not* a Little Cube)[1]

Technical Memo 6

Alvy Ray Smith
July 17, 1995

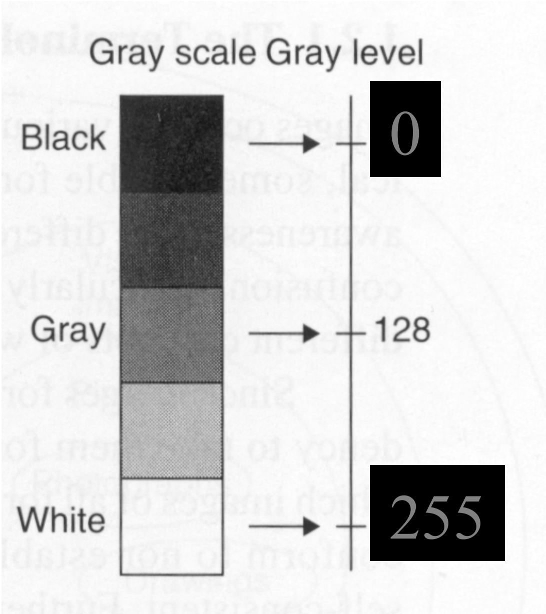
http://alvyray.com/Memos/CG/Microsoft/6_pixel.pdf

An image ?

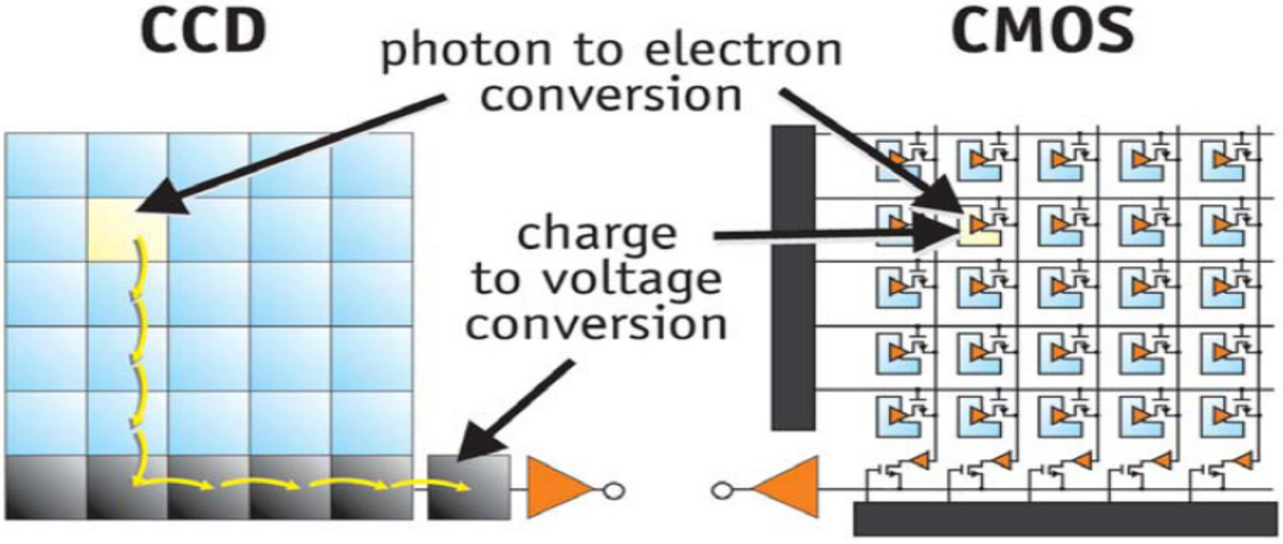


117	125	133	127	130	130	133	121	116	115	100	91	93	94	99	103	112	105	109	106
134	133	138	138	132	134	130	133	128	123	121	113	106	102	99	106	113	109	109	113
146	147	138	140	125	134	124	115	102	96	93	94	99	96	99	100	103	110	109	110
144	141	136	130	120	108	88	74	53	37	31	37	35	39	53	79	93	100	109	116
139	136	129	119	102	85	58	31	41	77	51	53	53	33	37	41	69	94	105	108
132	127	117	102	87	57	49	77	42	28	17	15	13	13	17	41	53	69	88	100
124	120	108	94	72	74	72	31	35	31	15	13	15	11	15	13	46	75	83	96
125	115	102	93	88	82	42	79	113	41	19	100	82	11	11	17	31	91	99	100
124	116	109	99	91	113	99	140	144	57	20	20	15	11	15	17	63	87	119	124
136	133	133	135	138	133	132	144	150	120	24	17	15	15	17	20	115	113	88	150
158	157	157	154	149	145	133	127	146	150	116	35	20	19	28	105	124	128	141	171
155	154	156	155	146	155	154	154	147	139	148	150	138	120	128	129	130	151	156	165
150	151	154	162	166	167	169	174	172	167	177	166	164	140	134	120	121	120	127	172
145	149	151	157	165	169	173	179	176	166	166	157	145	136	129	124	120	136	163	168
144	148	153	160	159	158	165	172	165	169	157	151	149	141	130	140	151	162	169	167
144	141	147	155	154	149	156	151	157	157	151	144	147	147	149	159	158	159	166	165
139	140	140	150	153	151	150	146	140	139	138	140	145	151	149	156	156	162	162	161
136	134	138	146	156	164	153	146	145	136	139	139	140	141	149	157	159	161	169	166
136	133	136	135	144	159	168	159	151	142	141	145	139	146	153	156	164	167	172	168
133	129	140	142	146	159	167	165	154	151	146	141	147	154	156	160	161	157	153	154

8 bits/pixel

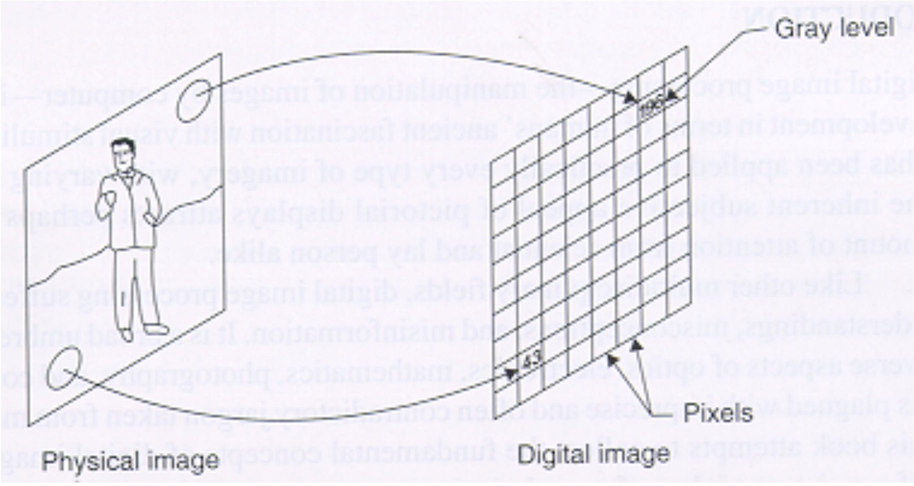


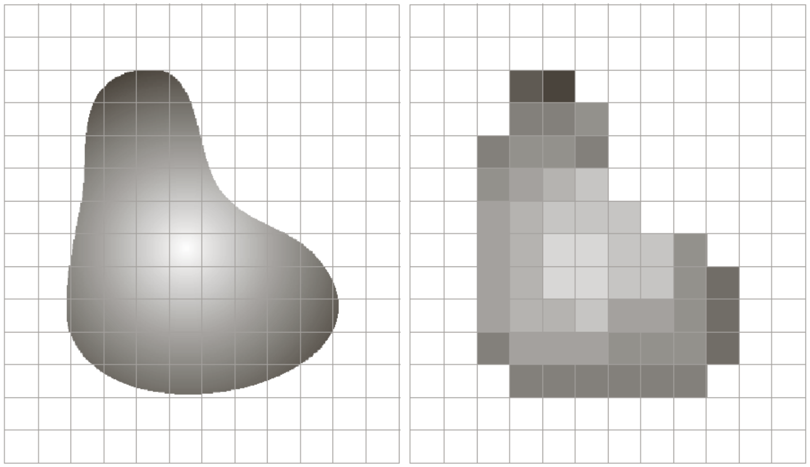
CCD Cameras - Resolution



http://www.dalsa.com/shared/content/pdfs/CCD_vs_CMOS_Litwiller_2005.pdf

Image Digitization

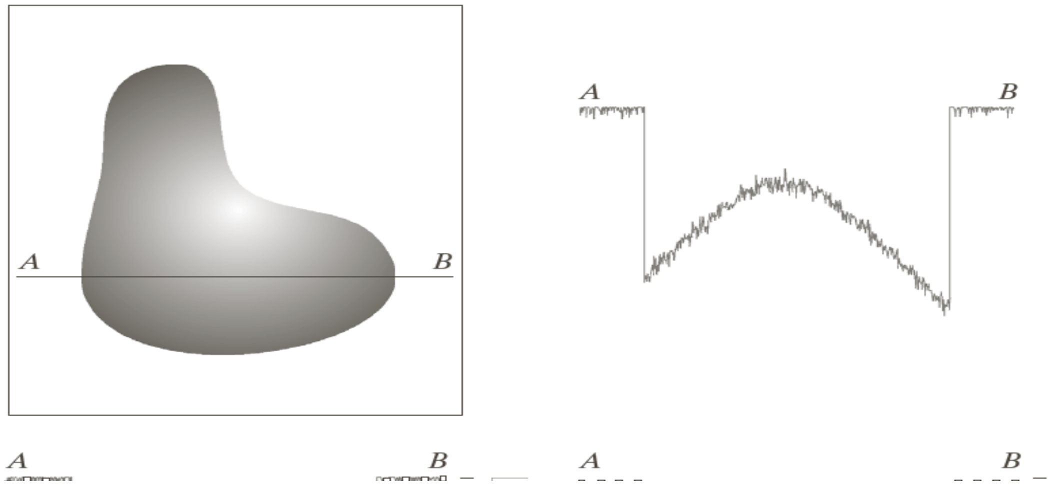


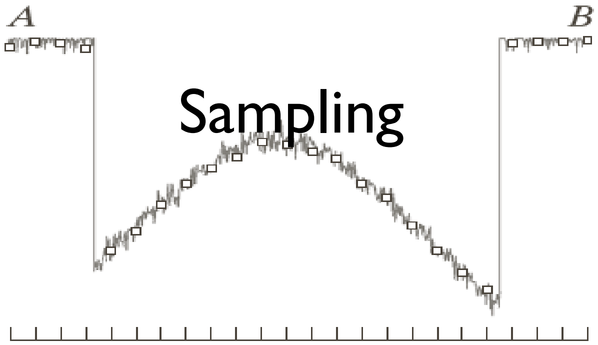


Sampling: Resolution

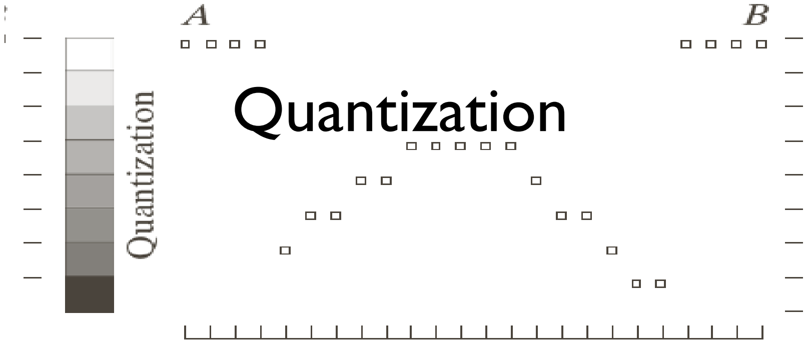
Quantization: Measured Value

Image Digitization





Sampling



Quantization

WebGL™

https://get.webgl.org/

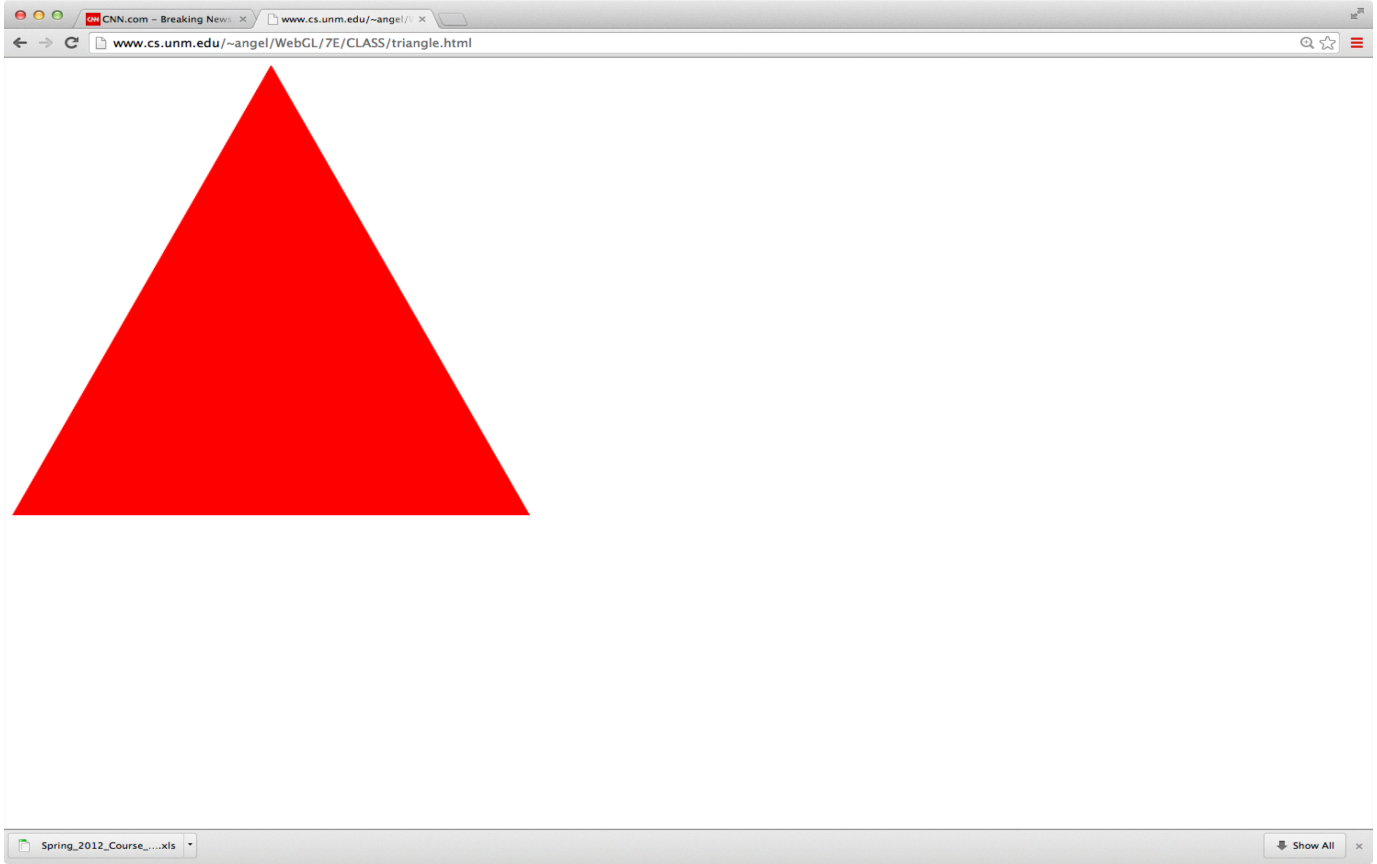
Kewl Sites

http://www.chromeexperiments.com/webgl/

Coding in WebGL

- Can run WebGL on any recent browser
    - Chrome
    - Firefox
    - Safari
    - IE
- Code written in JavaScript
- JS runs within browser
    - Use local resources

Example: triangle.html


CNN.com – Breaking News
www.cs.unm.edu/~angel/
www.cs.unm.edu/~angel/WebGL/7E/CLASS/triangle.html
Spring_2012_Course_....xls
Show All

Example Code

```
<!DOCTYPE html>
<html>
<head>
<script id="vertex-shader" type="x-shader/x-vertex">
attribute vec4 vPosition;
void main(){
  gl_Position = vPosition;
}
</script>
<script id="fragment-shader" type="x-shader/x-fragment">
precision mediump float;
void main(){
    gl_FragColor = vec4( 1.0, 0.0, 0.0, 1.0 );
}
</script>
```

HTML File (contd.)

```
<script type="text/javascript" src="../Common/webgl-utils.js"></script>
<script type="text/javascript" src="../Common/initShaders.js"></script>
<script type="text/javascript" src="../Common/MV.js"></script>
<script type="text/javascript" src="triangle.js"></script>
</head>
<body>
<canvas id="gl-canvas" width="512" height="512">
Oops ... your browser doesn't support the HTML5 canvas element
</canvas>
</body>
</html>
```

JS File

```
var gl;
var points;

window.onload = function init(){
    var canvas = document.getElementById( "gl-canvas" );
     gl = WebGLUtils.setupWebGL( canvas );
     if ( !gl ) { alert( "WebGL isn't available" );
}

// Three Vertices

var vertices = [
        vec2( -1, -1 ),
        vec2(  0,  1 ),
        vec2(  1, -1 )
];
```

JS File (contd.)

```
//  Configure WebGL
//
    gl.viewport( 0, 0, canvas.width, canvas.height );
    gl.clearColor( 1.0, 1.0, 1.0, 1.0 );

//  Load shaders and initialize attribute buffers

    var program = initShaders( gl, "vertex-shader", "fragment-shader" );
    gl.useProgram( program );

// Load the data into the GPU

    var bufferId = gl.createBuffer();
    gl.bindBuffer( gl.ARRAY_BUFFER, bufferId );
    gl.bufferData( gl.ARRAY_BUFFER, flatten(vertices), gl.STATIC_DRAW );
```

JS File (contd.)

```
// Associate out shader variables with our data buffer

      var vPosition = gl.getAttribLocation( program, "vPosition" );
      gl.vertexAttribPointer( vPosition, 2, gl.FLOAT, false, 0, 0 );
      gl.enableVertexAttribArray( vPosition );
      render();
};

function render() {
    gl.clear( gl.COLOR_BUFFER_BIT );
   gl.drawArrays( gl.TRIANGLES, 0, 3 );
}
```

JavaScript Notes

- JavaScript (JS) is the language of the Web
  - All browsers will execute JS code
  - JavaScript is an interpreted object-oriented language

JS Notes

- Is JS slow?
  - JS engines in browsers are getting much faster
  - Not a key issues for graphics since once we get the data to the GPU it doesn't matter how we got the data there
- JS is a (too) big language
  - We don't need to use it all
  - Choose parts we want to use
  - Don't try to make your code look like C or Java

That's all Folks!