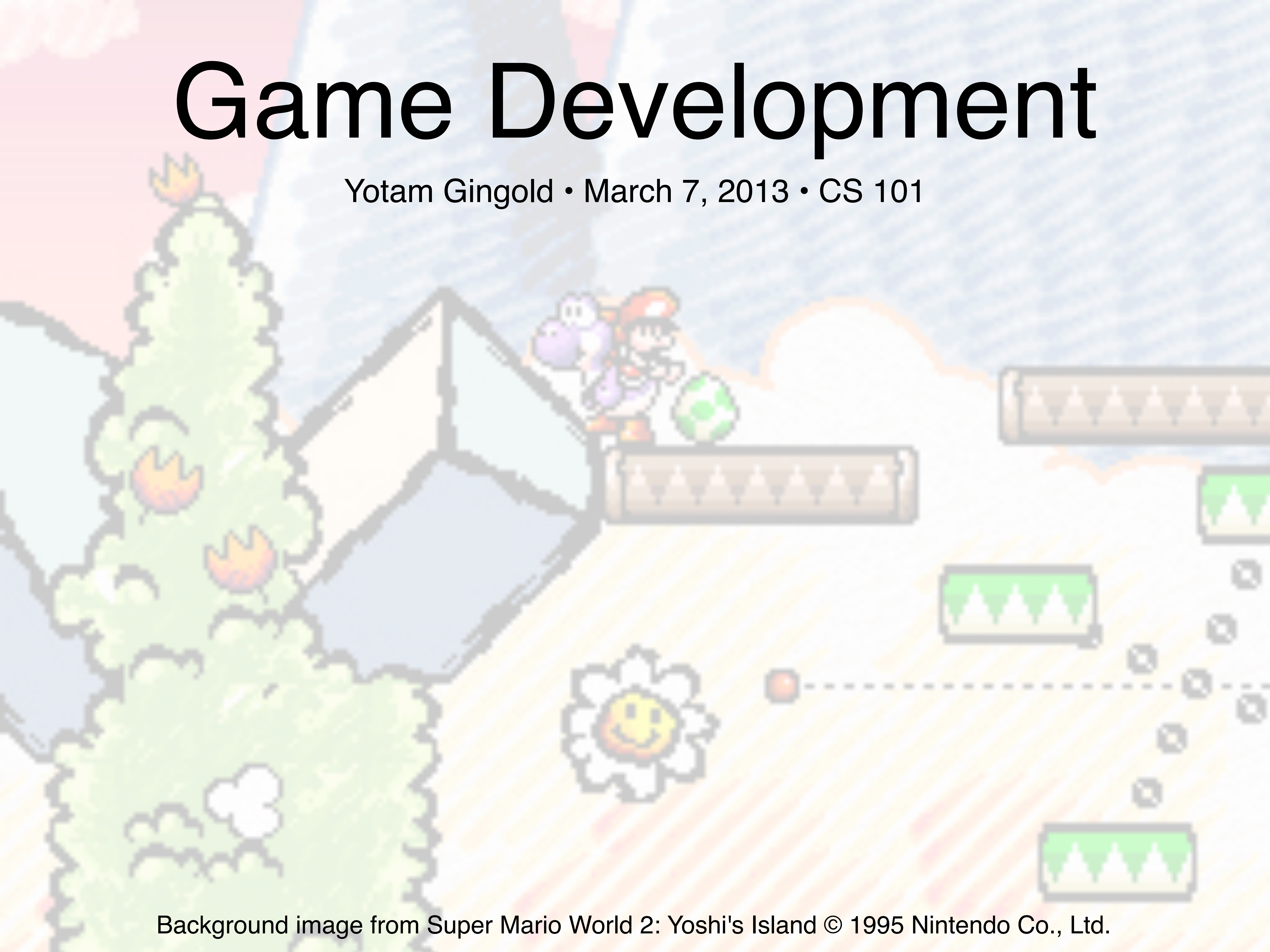
Game Development

Yotam Gingold • March 7, 2013 • CS 101

Game Development

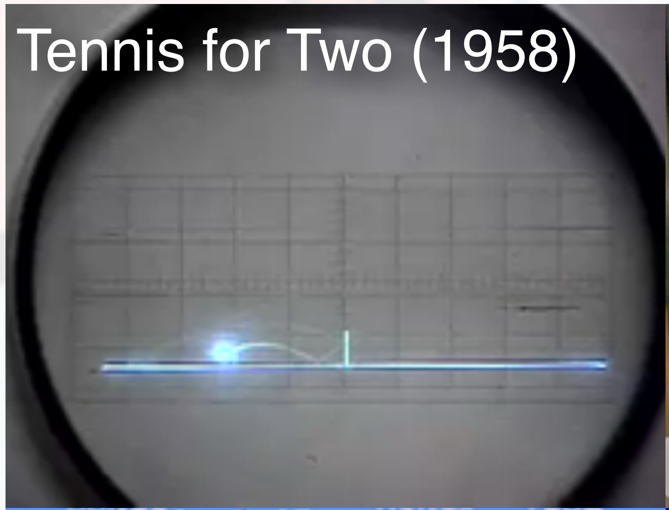
Yotam Gingold • March 7, 2013 • CS 101

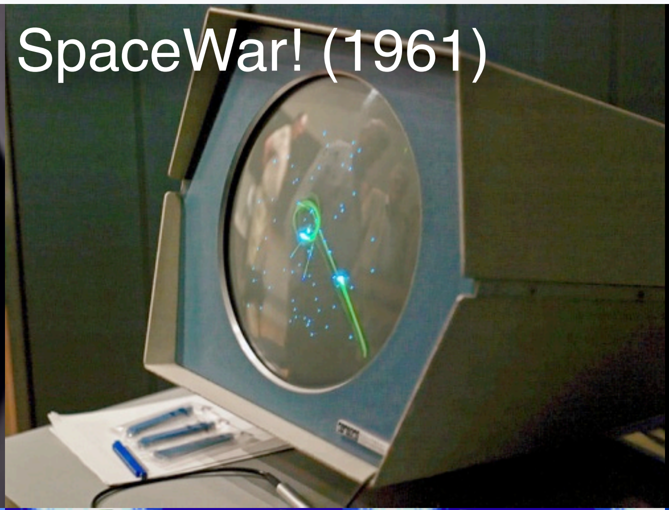


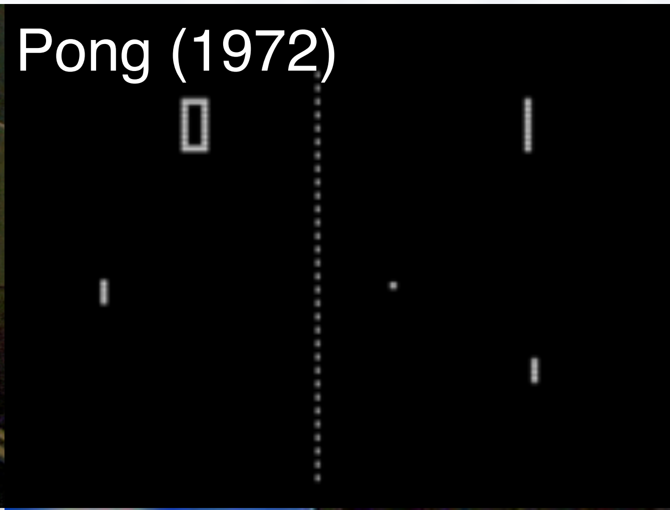
Background image from Super Mario World 2: Yoshi's Island © 1995 Nintendo Co., Ltd.

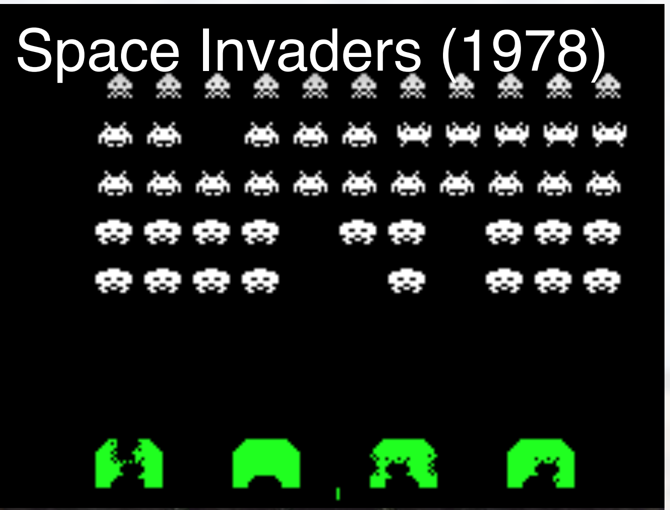
What games are you playing?

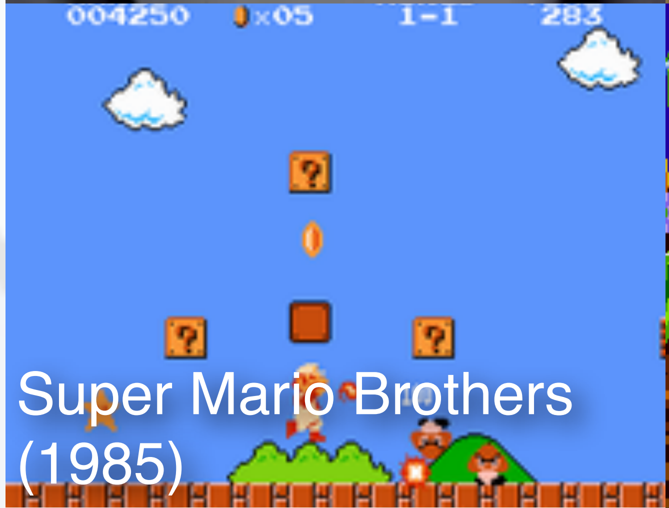
Video Games


Tennis for Two (1958)

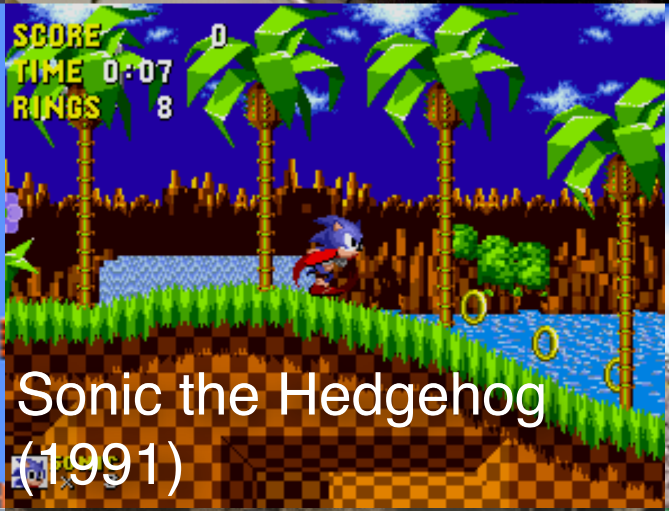

SpaceWar! (1961)


Pong (1972)

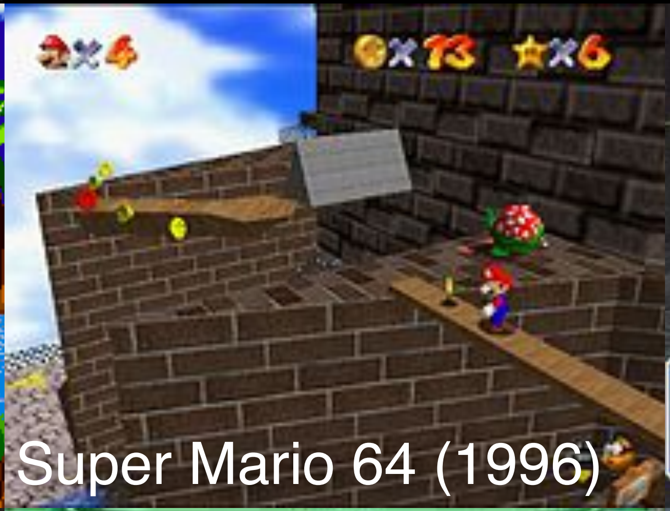

Space Invaders (1978)



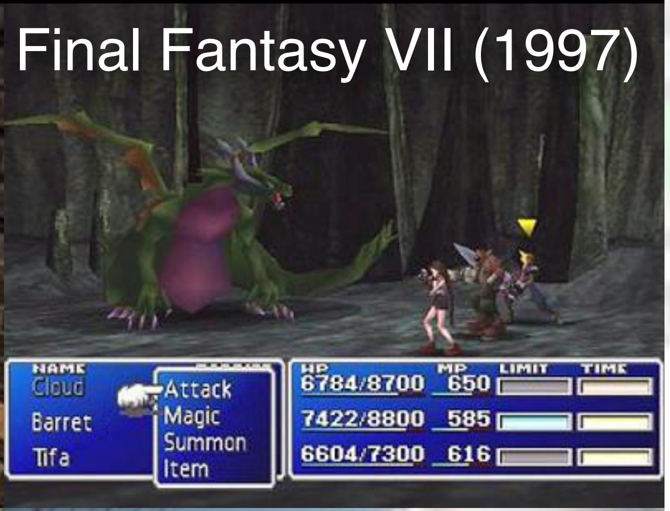
Super Mario Brothers (1985)



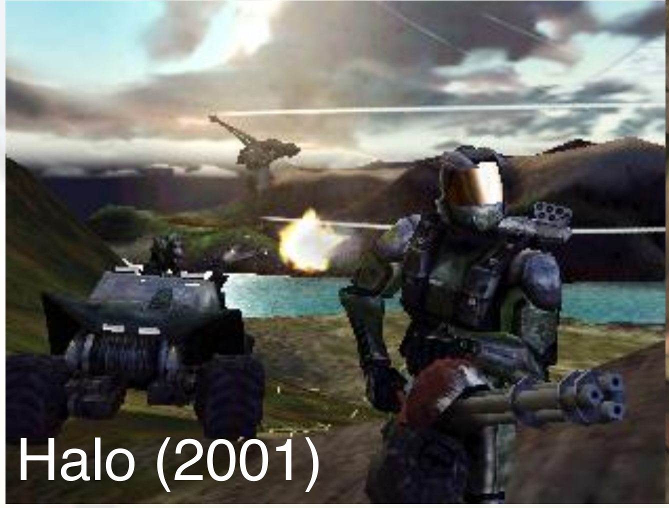
Sonic the Hedgehog (1991)

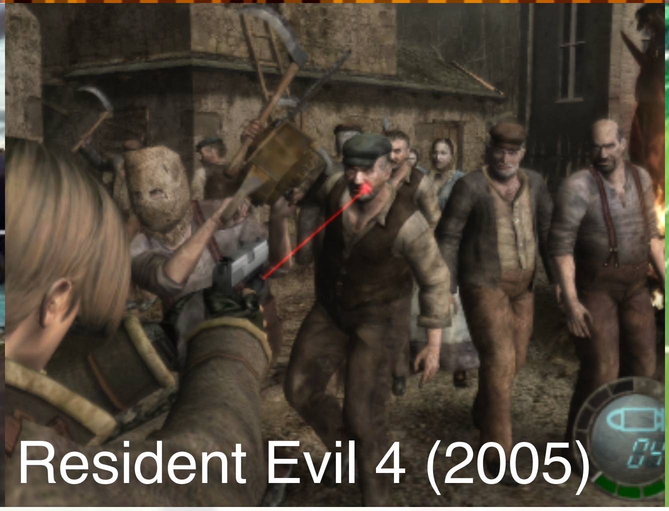


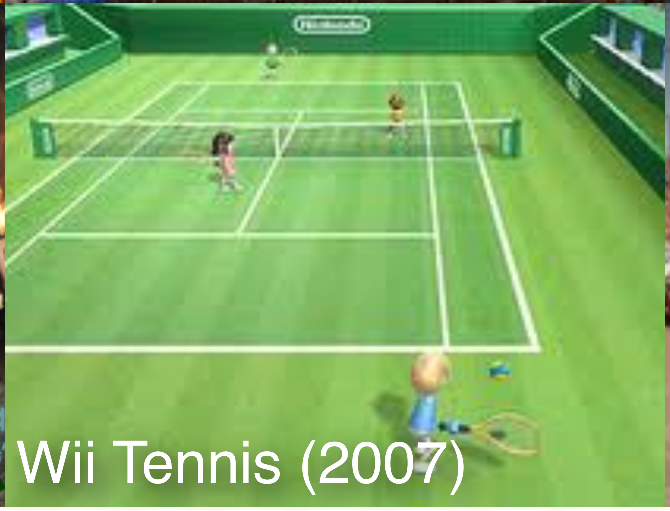
Super Mario 64 (1996)

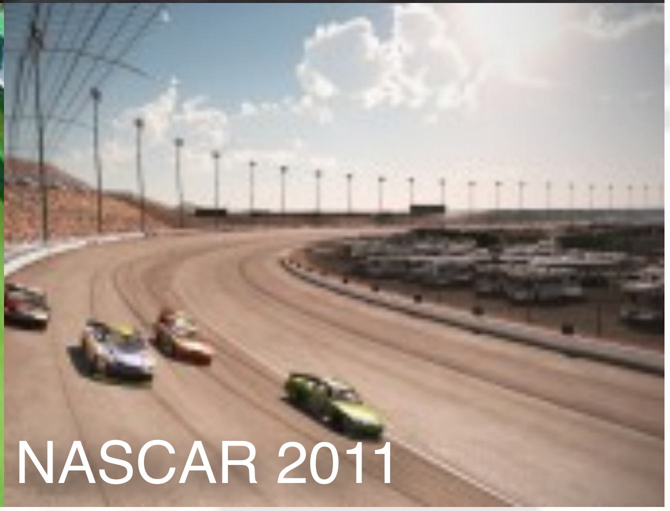


Final Fantasy VII (1997)


Halo (2001)


Resident Evil 4 (2005)


Wii Tennis (2007)


NASCAR 2011

Video Game Hardware

Generation: 1st 2nd 3rd

Video Game Hardware

Generation: 2nd 3rd 4th 5th

Video Game Hardware

3rd

4th

5th

6th

7th

What do we want from our games?

What do we want from our games?

Fun?

Pretty?

What do we want from our game technology?

What do we want from our game technology?

Simulate and display an interesting, complex world

What do we want from our game technology?

Simulate and display an interesting, complex world

Responsive

What do we want from our game technology?

- Simulate and display an interesting, complex world
- Responsive
- Networked play

What do we want from our game technology?

Simulate and display an interesting, complex world

Responsive

Networked play

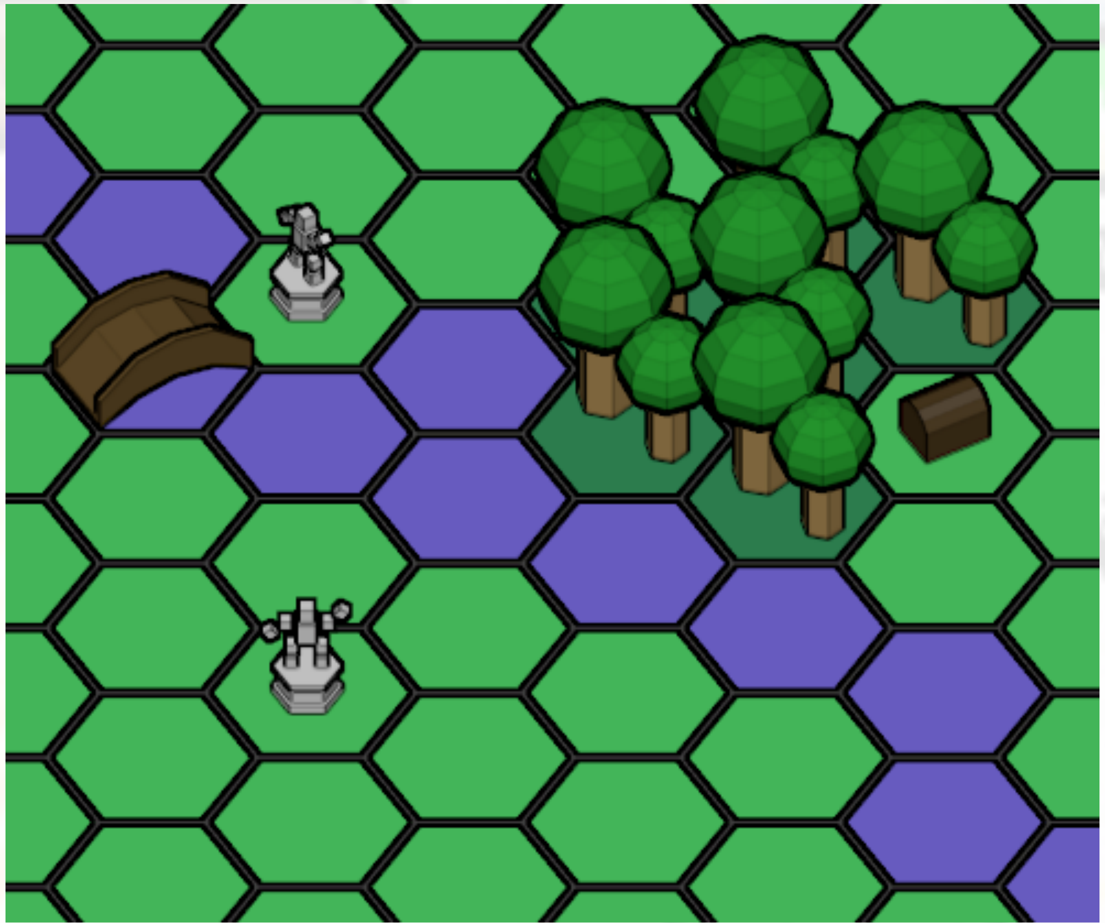
Not buggy

Programming languages

Run gameplay code in a language easy to program in

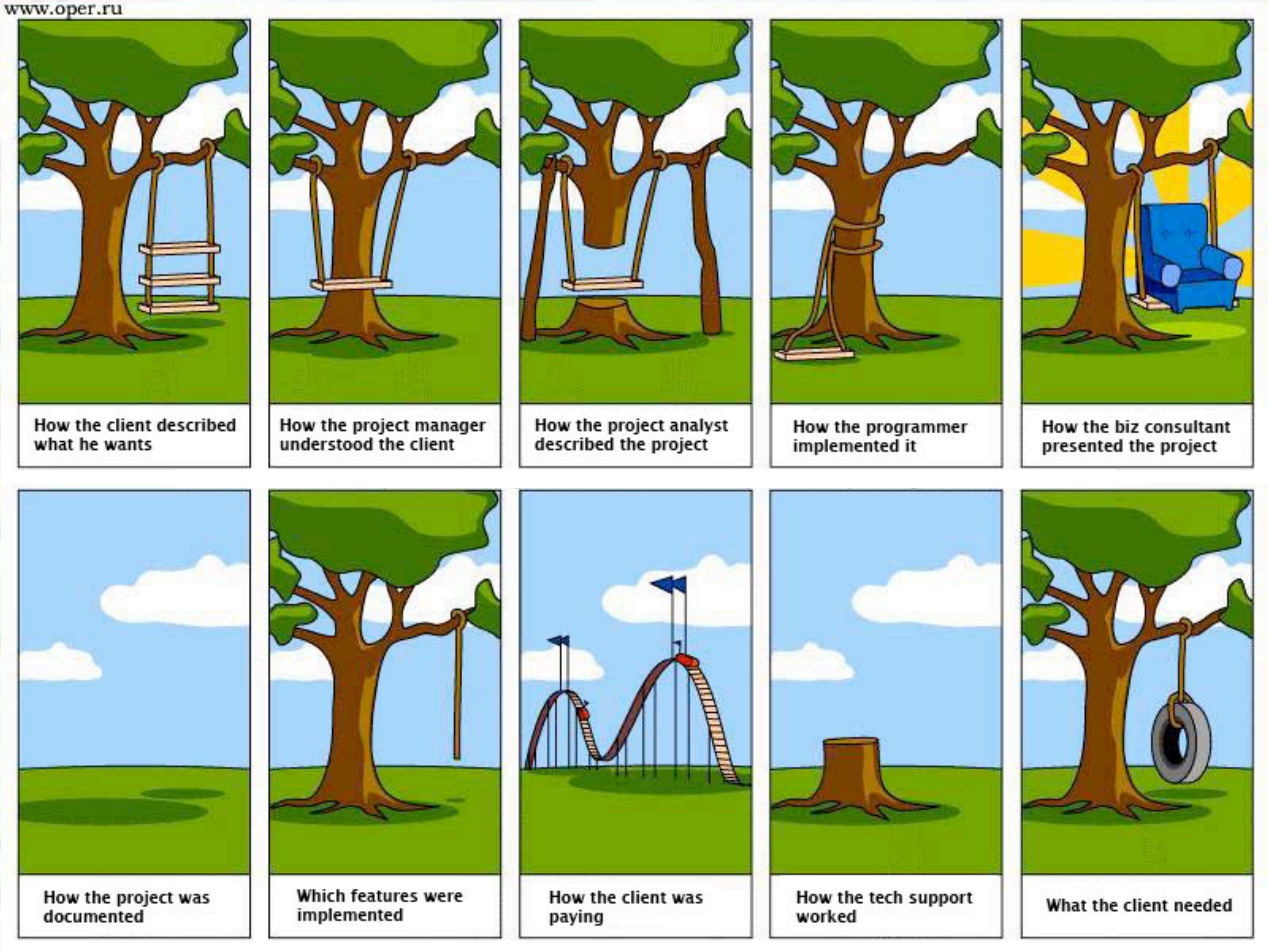
- allow designers to try ideas
- safely allow MODs

Parallel programming



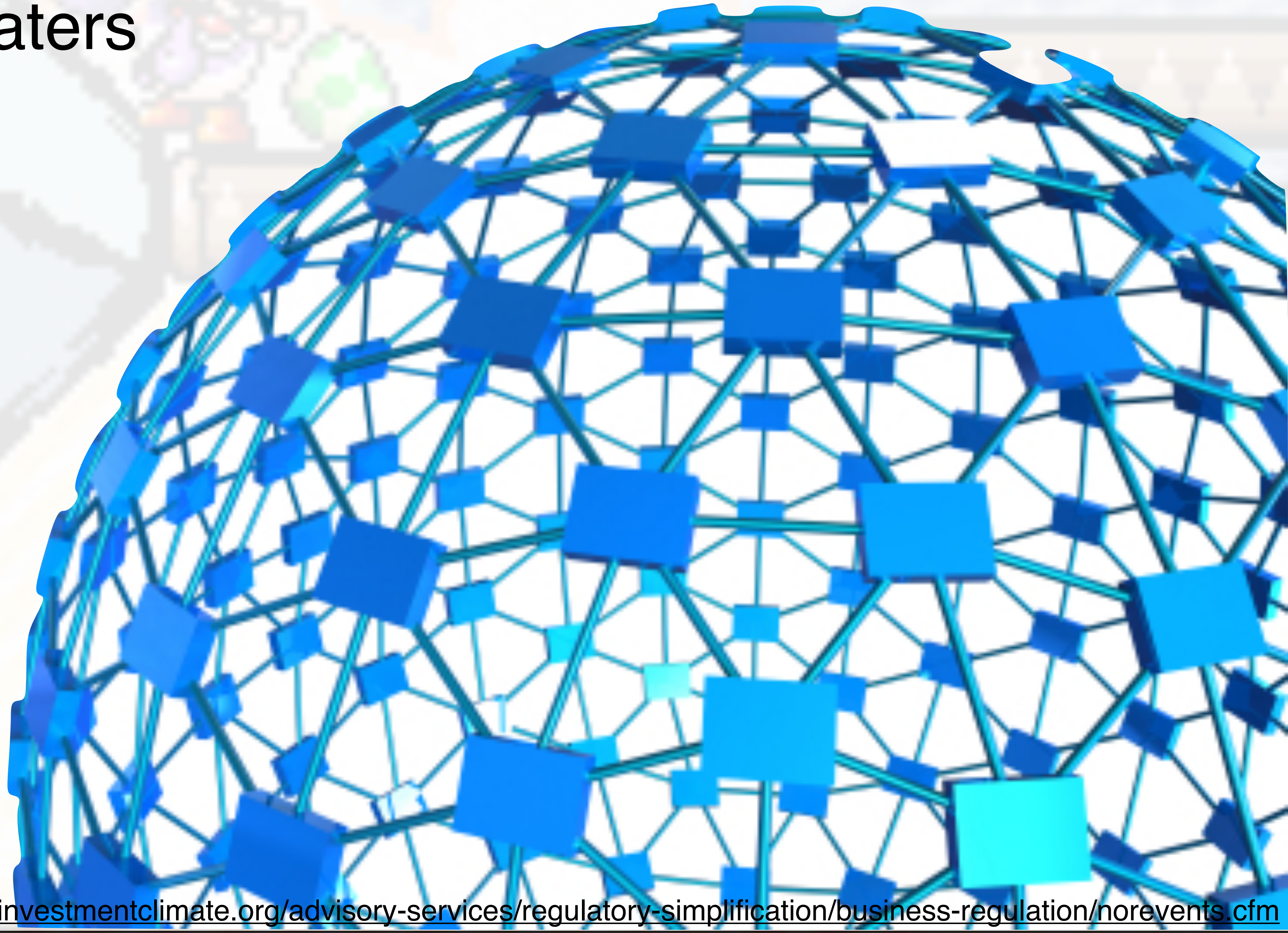
http://redclovergames.com/blog/?p=205

Software engineering


www.oper.ru
How the client described what he wants
How the project manager understood the client
How the project analyst described the project
How the programmer implemented it
How the biz consultant presented the project
How the project was documented
Which features were implemented
How the client was paying
How the tech support worked
What the client needed

Networking & Security

- Efficiently share state of the game with other players
- Detect "bots" or cheaters
- Copy protection



https://www.wbginvestmentclimate.org/advisory-services/regulatory-simplification/business-regulation/norevents.cfm

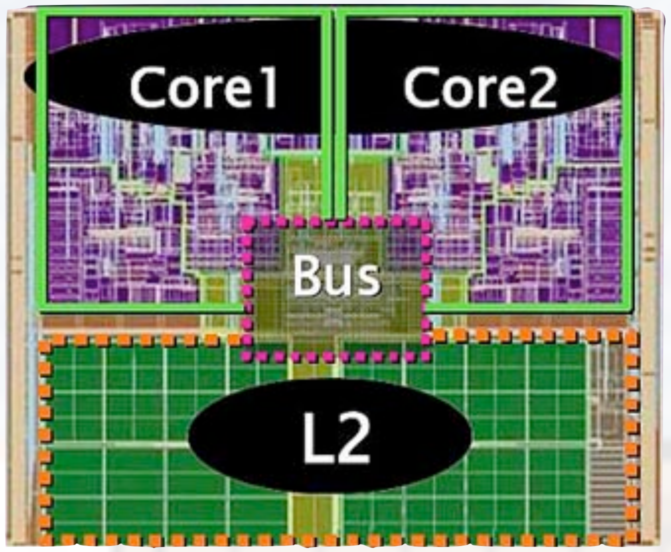
Operating Systems

Resource allocation

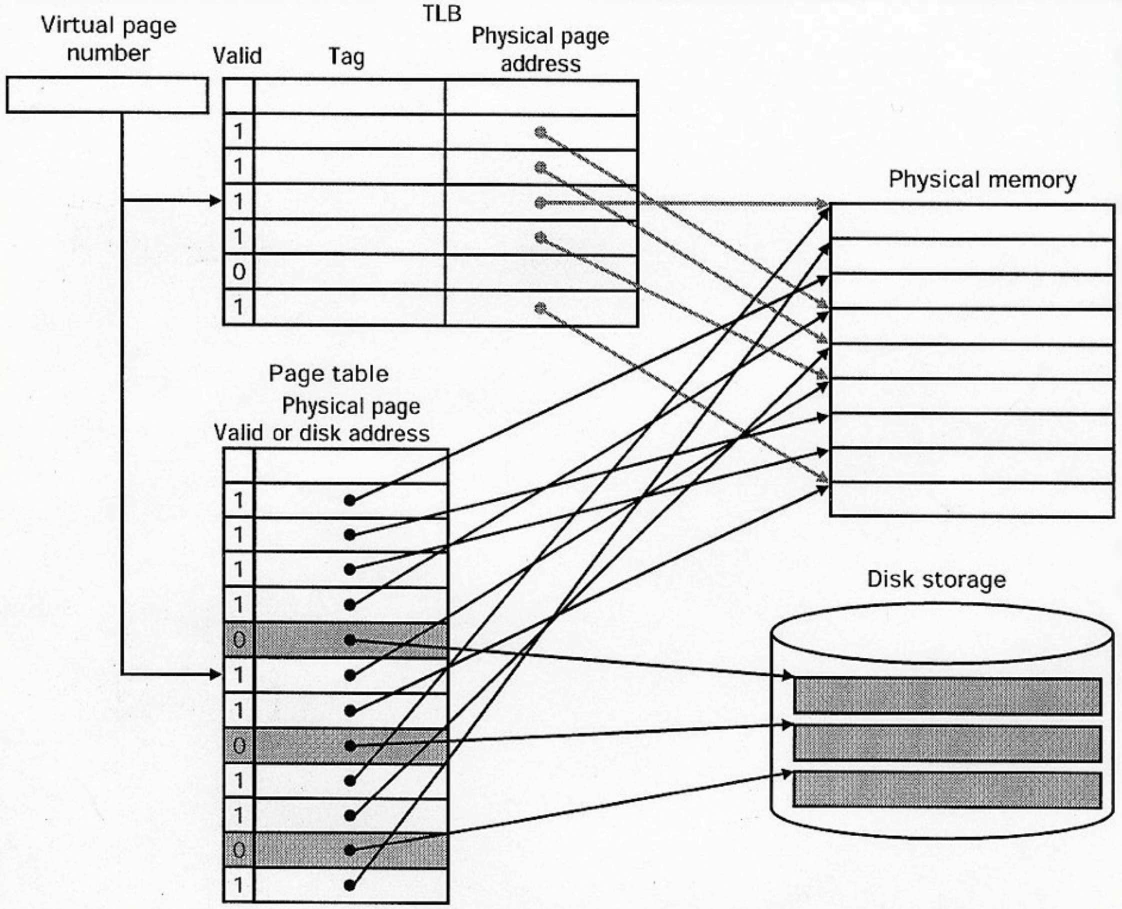
- RAM, caching

Scheduling CPU time for the subsystems

- graphics, physics, networking, AI, button presses



Intel Core Duo



http://users.ece.gatech.edu/~dblough/3055/

Game Design

The rules of the world and the game

- Thinking in Systems (Donella H. Meadows)
- game theory
- Playing to win (David Sirlin)

Game Design

Game Design

CHN WANG X.L/YU Y. 0 1
KOR JUNG K.E./KIM H.N. 0 2
Courtesy NBC Olympics
abc NEWS .com
#ABCWorldNews

Game Design

The rules of the world and the game

- Thinking in Systems (Donella H. Meadows)
- game theory
- Playing to win (David Sirlin)

Interaction design (Human-Computer Interaction)

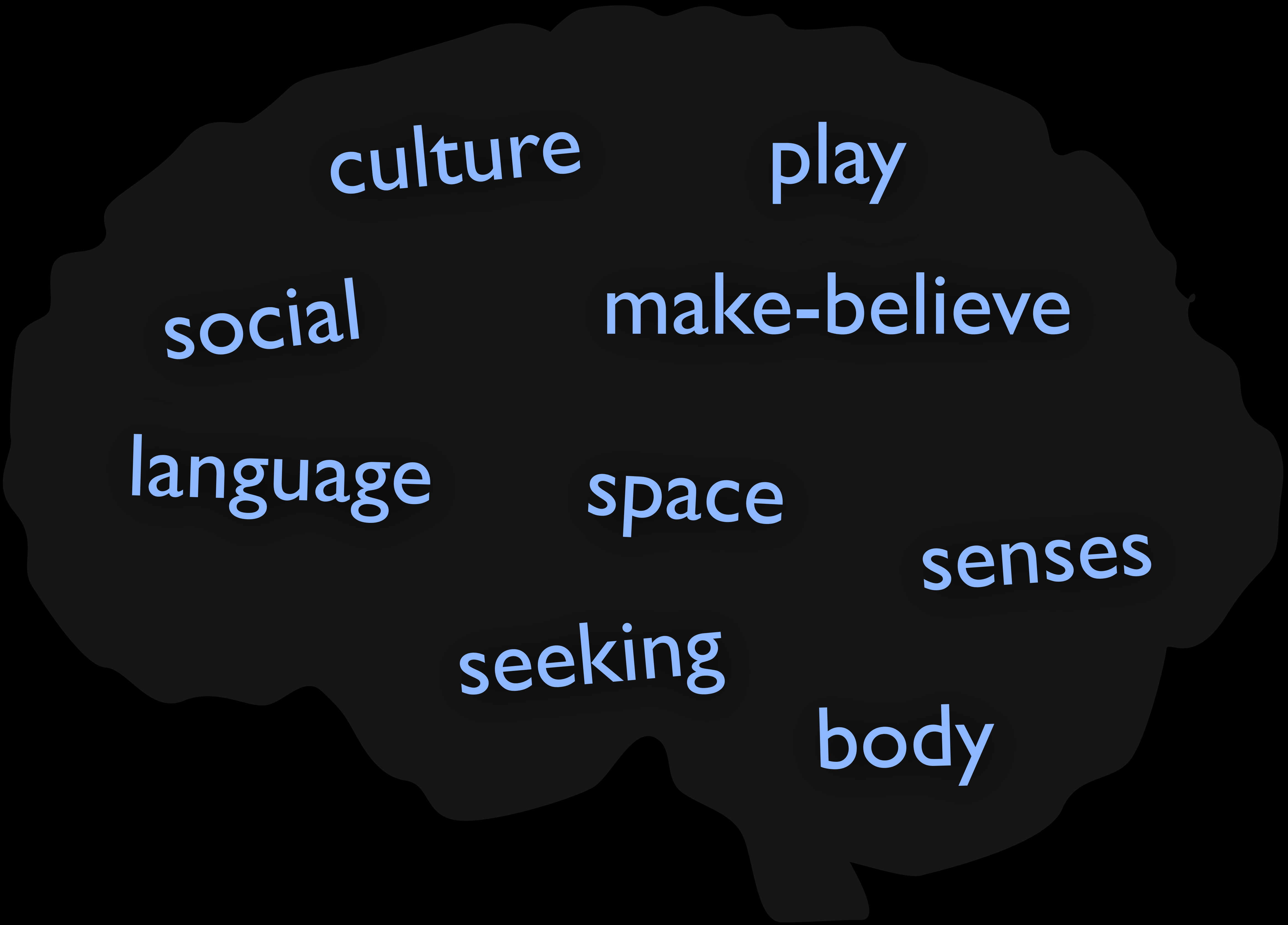
- Game Feel (Steve Swink)

What is the “fun”?

- psychology (e.g. Csikszentmihalyi’s “flow”)
- The Human Play Machine (Chaim Gingold)

The Human Play Machine

Chaim Gingold
cog@slackworks.com



culture

play

social

make-believe

language

space

senses

seeking

body

fun
exciting

joy

creativity
generativity
flexibility

all behavior

integration

learning

Play

hard to define
easy to identify

serious
silly

attitude

not real

a state of mind

real

fleeting

Seeking

easter egg hunt

exploring

adventure games

unwrapping presents

television

orienting

twitch games

shooters

unknown outcome

suspense

stories

games

deciding

economics

haggling

trading

chess

improving skills

learning

creating

making

inventing

PacMan

addiction

random rewards

gambling

slot machines

e-mail

Body

real body

dance

sex

climbing

sports

soccer

wii

ddr

virtual body

mario

game characters

environment

levels

virtual

building

destroying

real

sandcastles

pillow forts

tools

drawing

instruments

game controllers

tennis racquet

skateboarding

appropriation

playground

graffiti

meta

Katamari Damacy

Portal

power

Go

hide and seek

Space

hiking

SimCity

Mario

Disneyland

architecture

aesthetics

gardens

Japanese gardens

maze

levels

puzzle

cognitive

Image of the City, Kevin Lynch

stories
games
reality synthesis
fantasy
escape
puppets
Star Wars

Peter Pan
meta
improv
Italo Calvino
warp zones

Mario
control
aspiration
empathy
heroes
role play
dress up
prom
halloween
D&D
masks
LARP
Guitar Hero

Make believe

Spore
SimCity
The Sims
creativity augmentation
Guitar Hero
Little Big Planet

playing house
model making
fabrication
texting
doodling
pillow fort
e-mail
joking

Senses

touch

Rez

Go stones

World of Goo

Tactile Dome

fragrance

taste & smell

food

cooking

dinner party

Iron Chef

Cooking Moma

touch & feel

synesthesia

Kandinsky

Debussy

observing

art

drawing

vision

Where's Waldo?

iPod

sound

music

listening

playing

Escher

meta

Echochrome

Paper Mario Wii

Mario Galaxy

teams

affiliation

guilds

fans
Facebook

battle rap
pity match
potlatch

boasting

Pac Man
power

competition

sports
judo

Street Fighter II

folk dancing

coordination
mimicry

empathy

mirror neurons
wii

film

game identity

Social

complexity

virtual
Facade

The Sims

multiplayer

Hello Kitty

nurturing

pets

Tamagotchi
Yoshi's Island
The Sims

lust

dating
online

flirting

romance novels
strip clubs

DNA

signs

emergence

interactive fiction

poetry

literature

word play

Taboo

Scrabble

Language

music

song

gibberish

appropriation

Burning Man

hip-hop

skateboarding

graffiti

Katamari Damacy

levity

nonsense

Lewis Carroll

Culture

Wario Ware

improv everywhere

sacred cows

breaking rules

cultural codes

social conventions

satire

clowning

Charlie Chaplin

Grand Theft Auto

The Sims

Borat

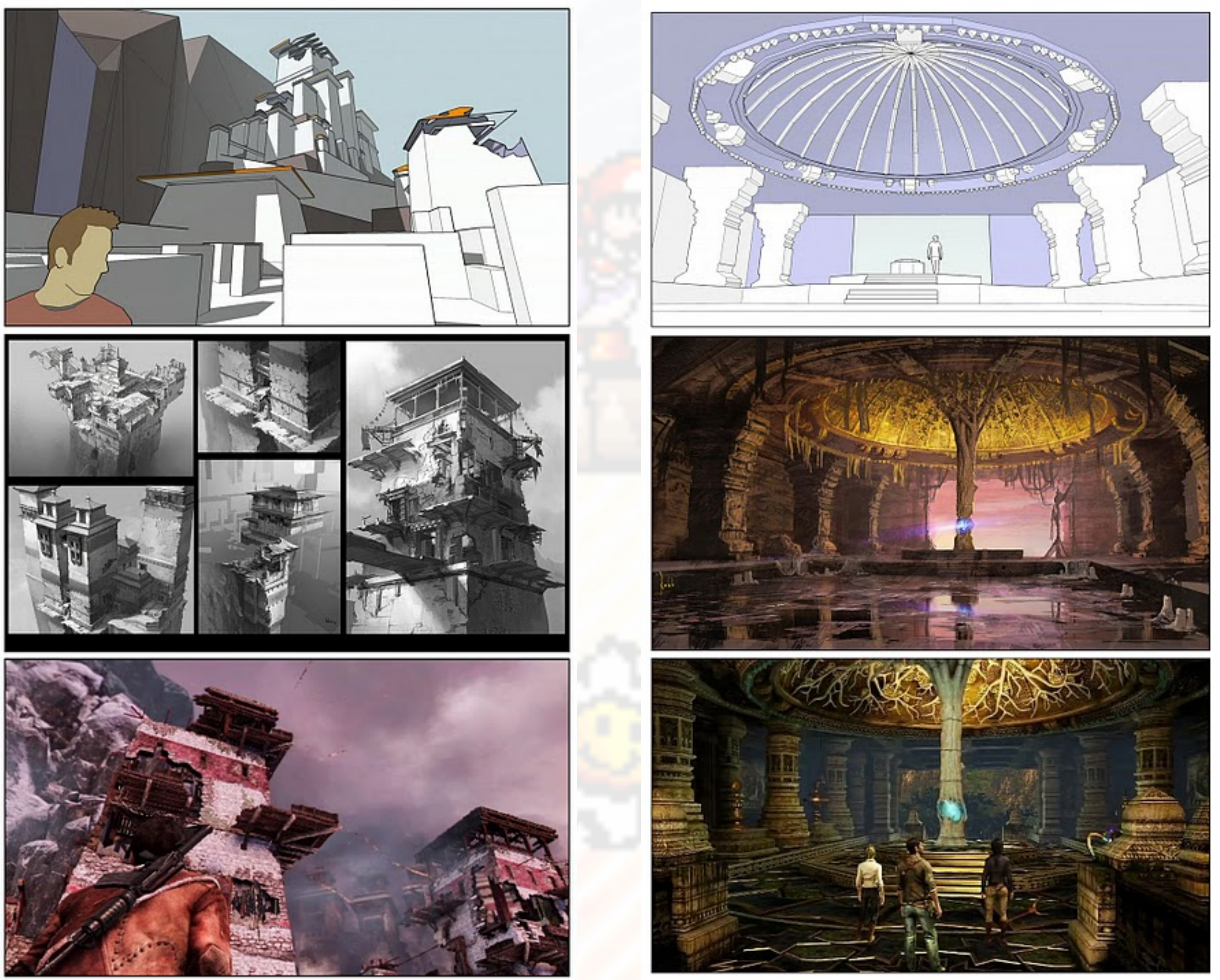
griefing

trolling

The Human Play Machine Chaim Gingold © 2009

Art

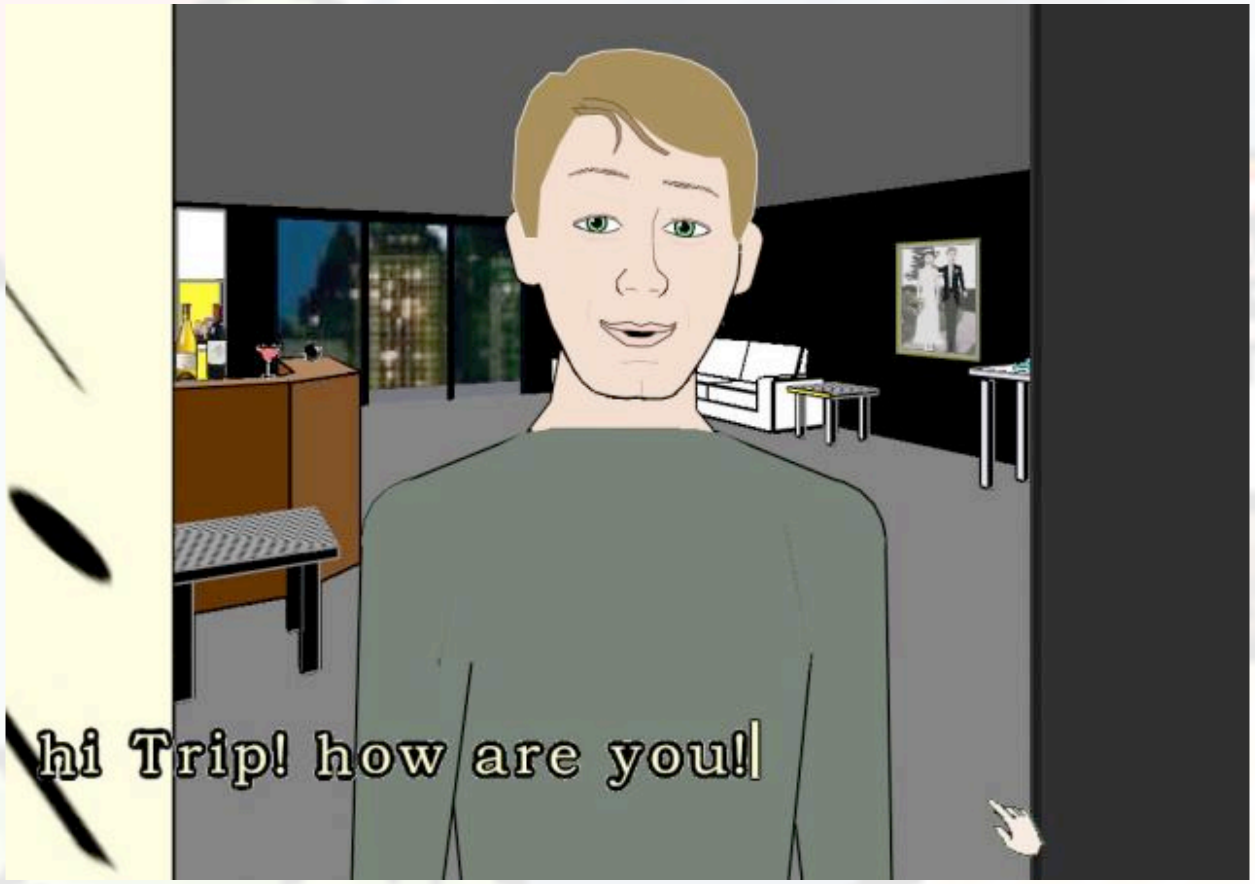
Concept



Final

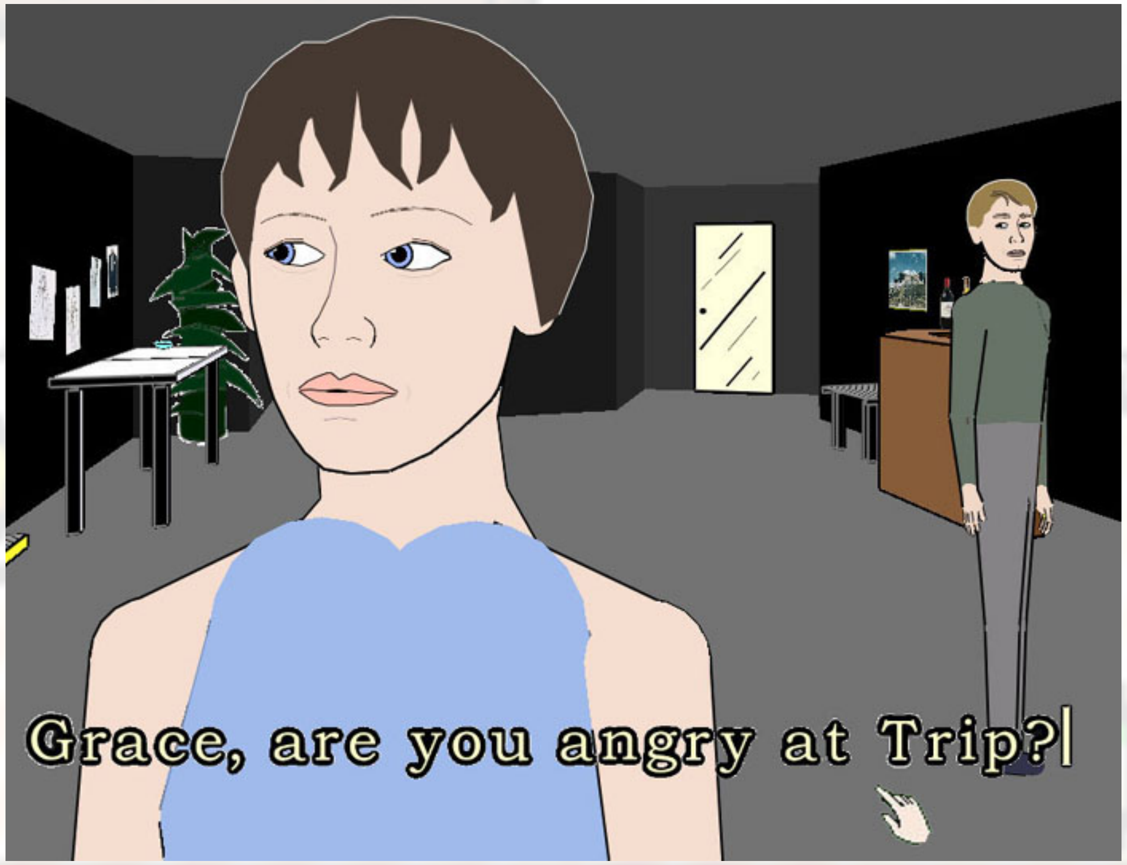
Story

Façade



hi Trip! how are you!

You ha



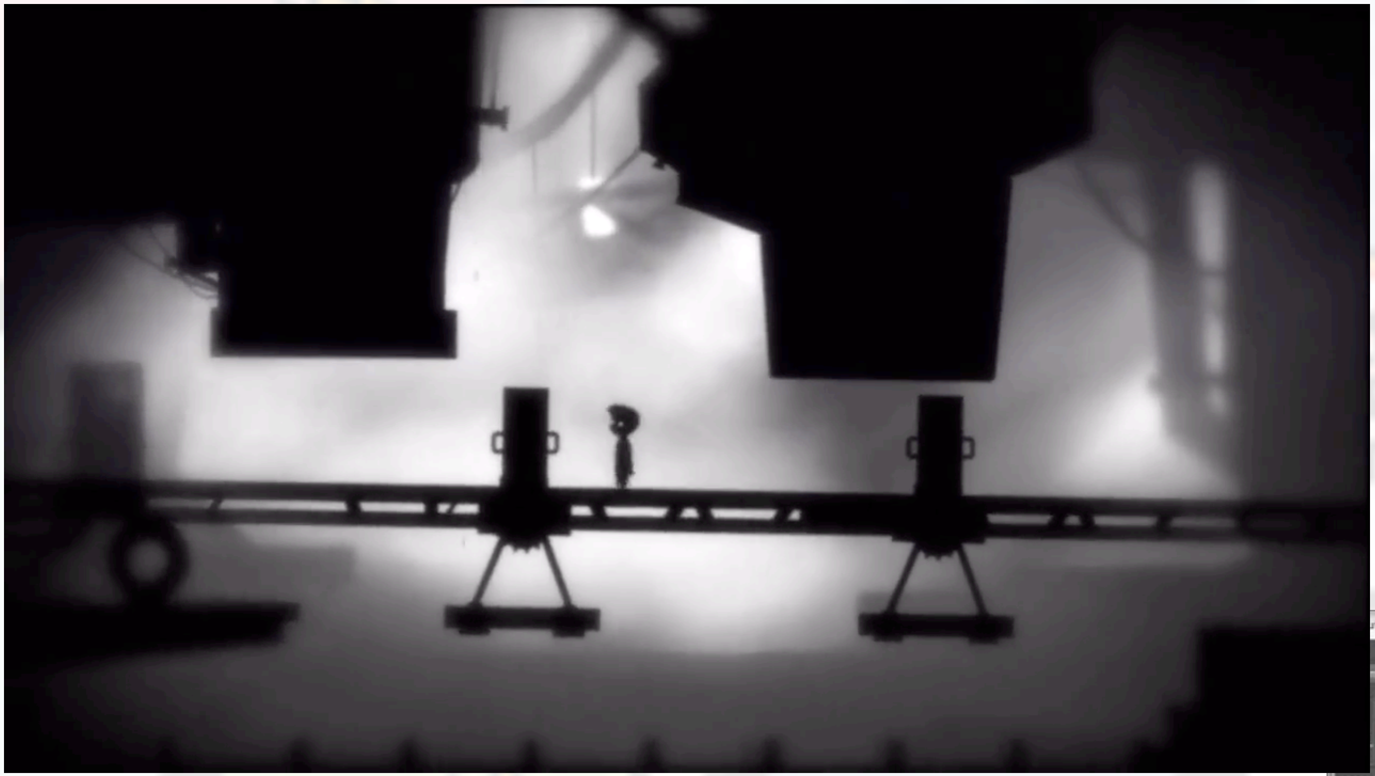
Grace, are you angry at Trip?

Artificial Intelligence

An intelligent world

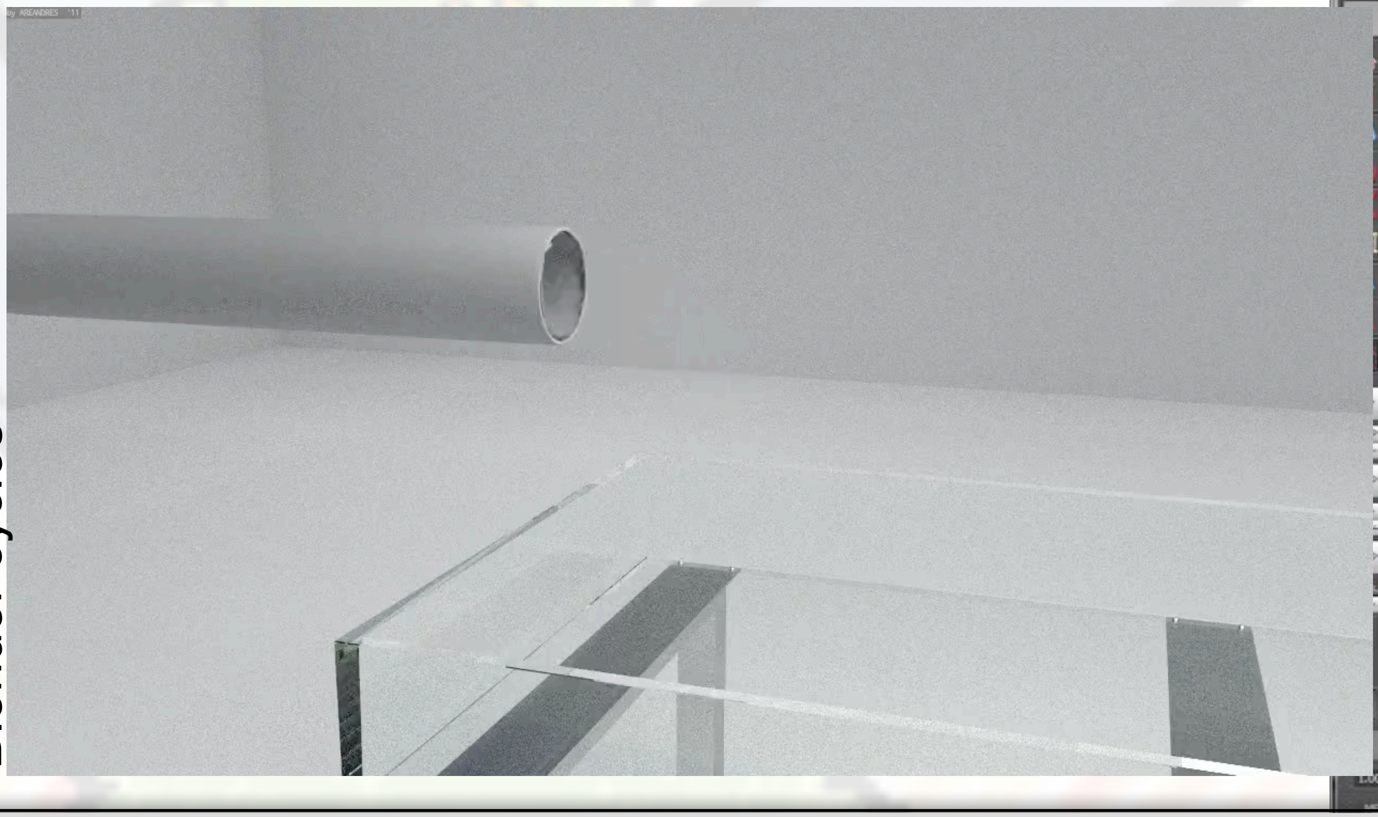
- Characters
- Interactive story

Physics & Animation



Limbo

Angry Birds



Blender cycles

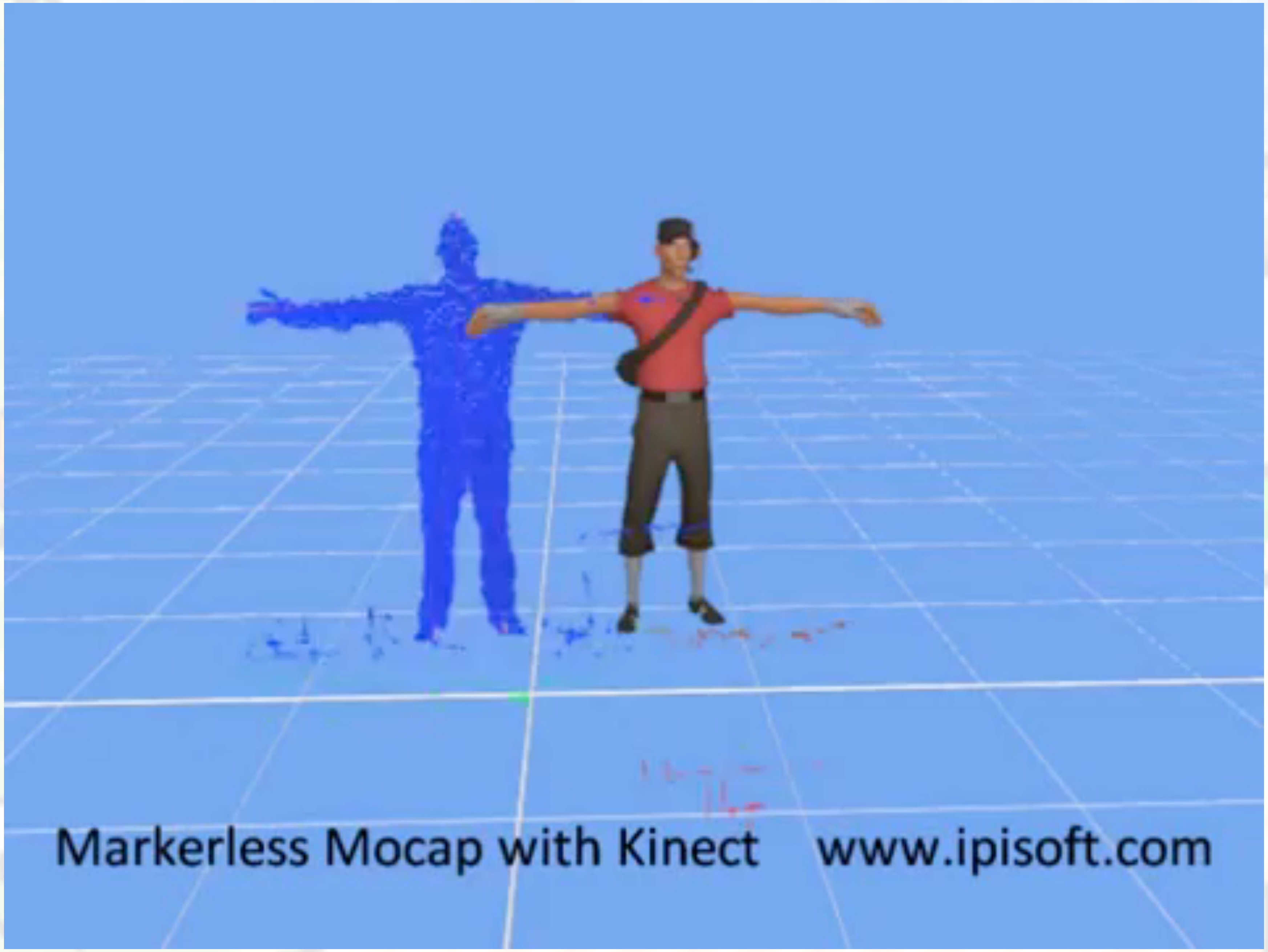
IKinema

IKinema

Input

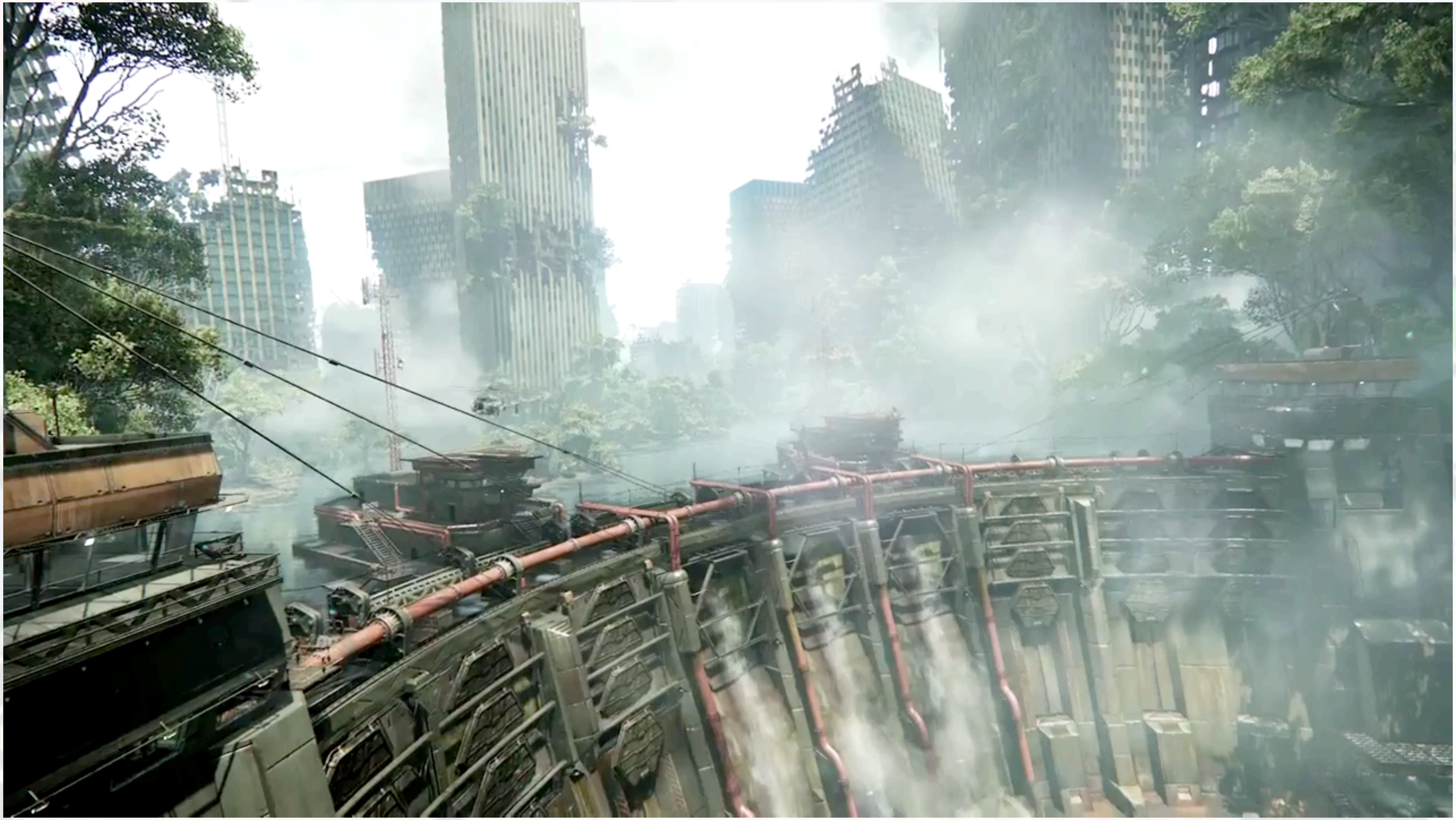
Computer vision

Pattern recognition

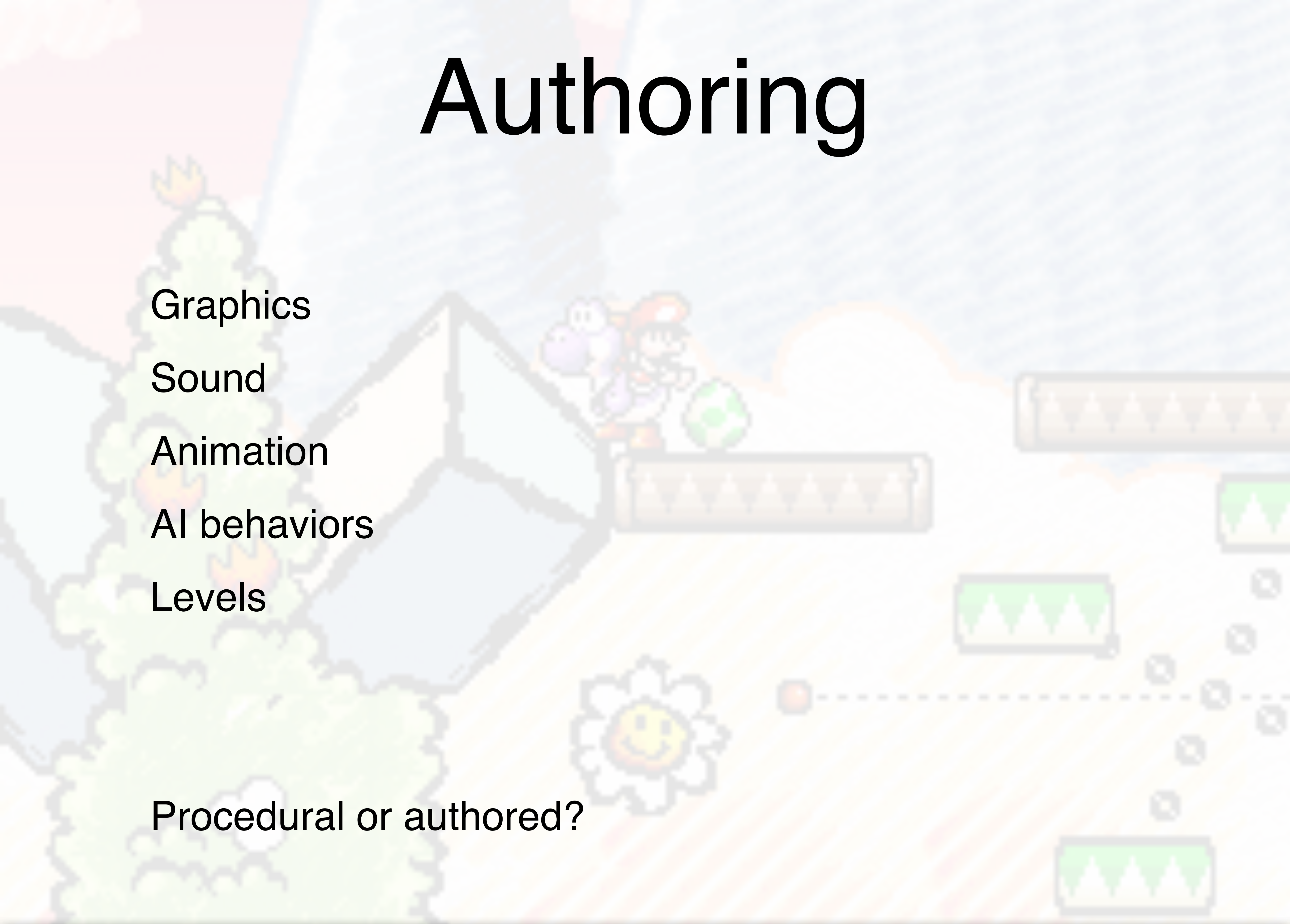


Rendering

Crysis 3



Authoring



Graphics

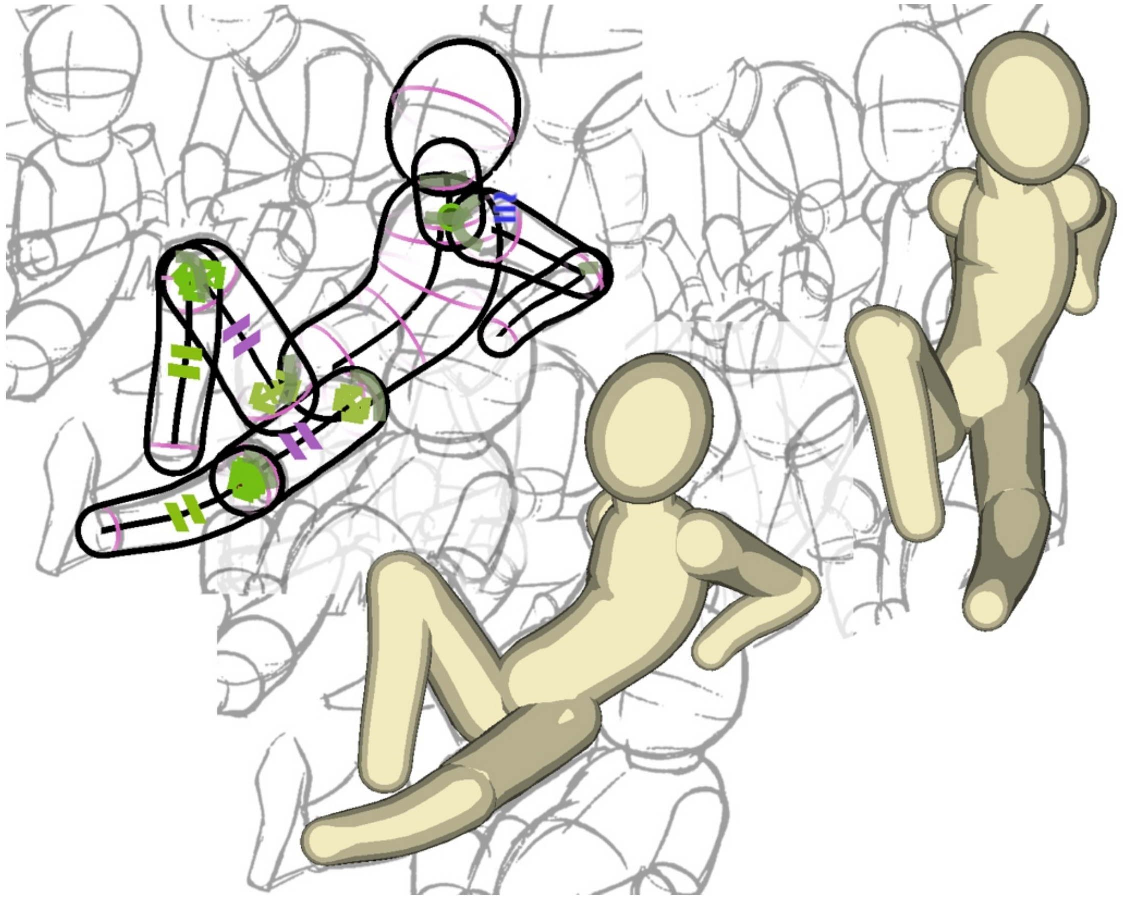
Sound

Animation

AI behaviors

Levels

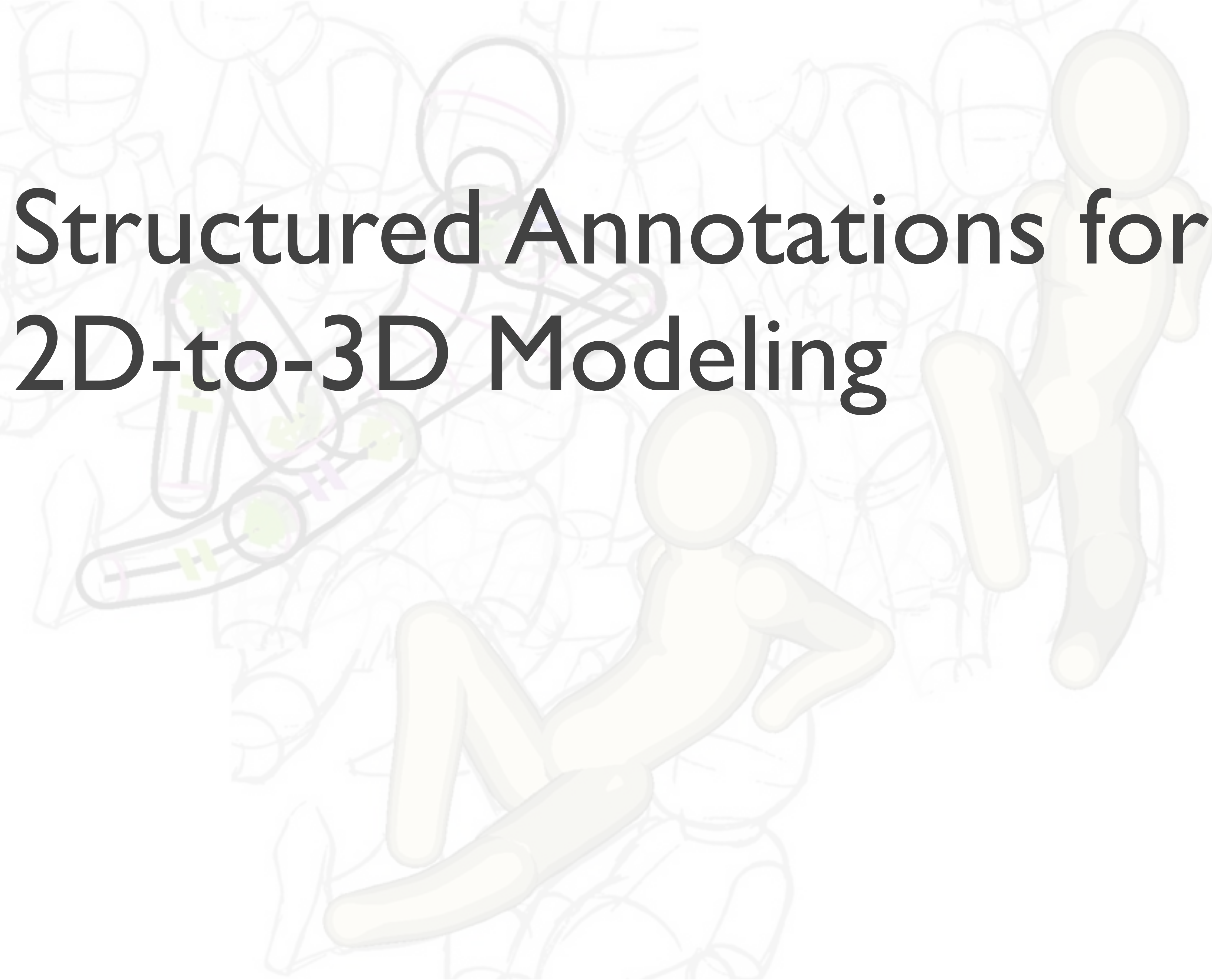
Procedural or authored?



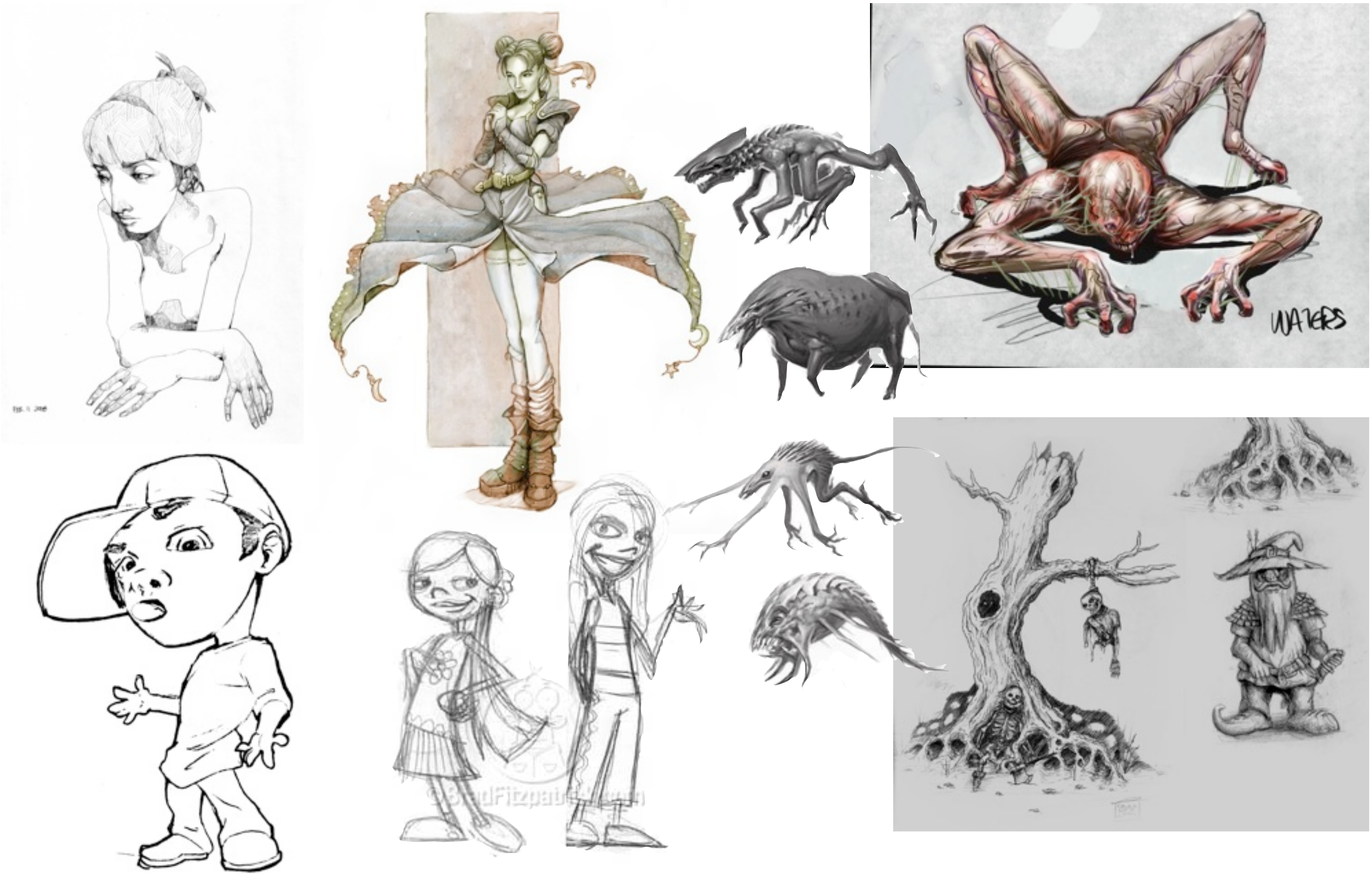
Structured Annotations for 2D-to-3D Modeling

A Direct Texture Placement and Editing Interface

Structured Annotations for 2D-to-3D Modeling

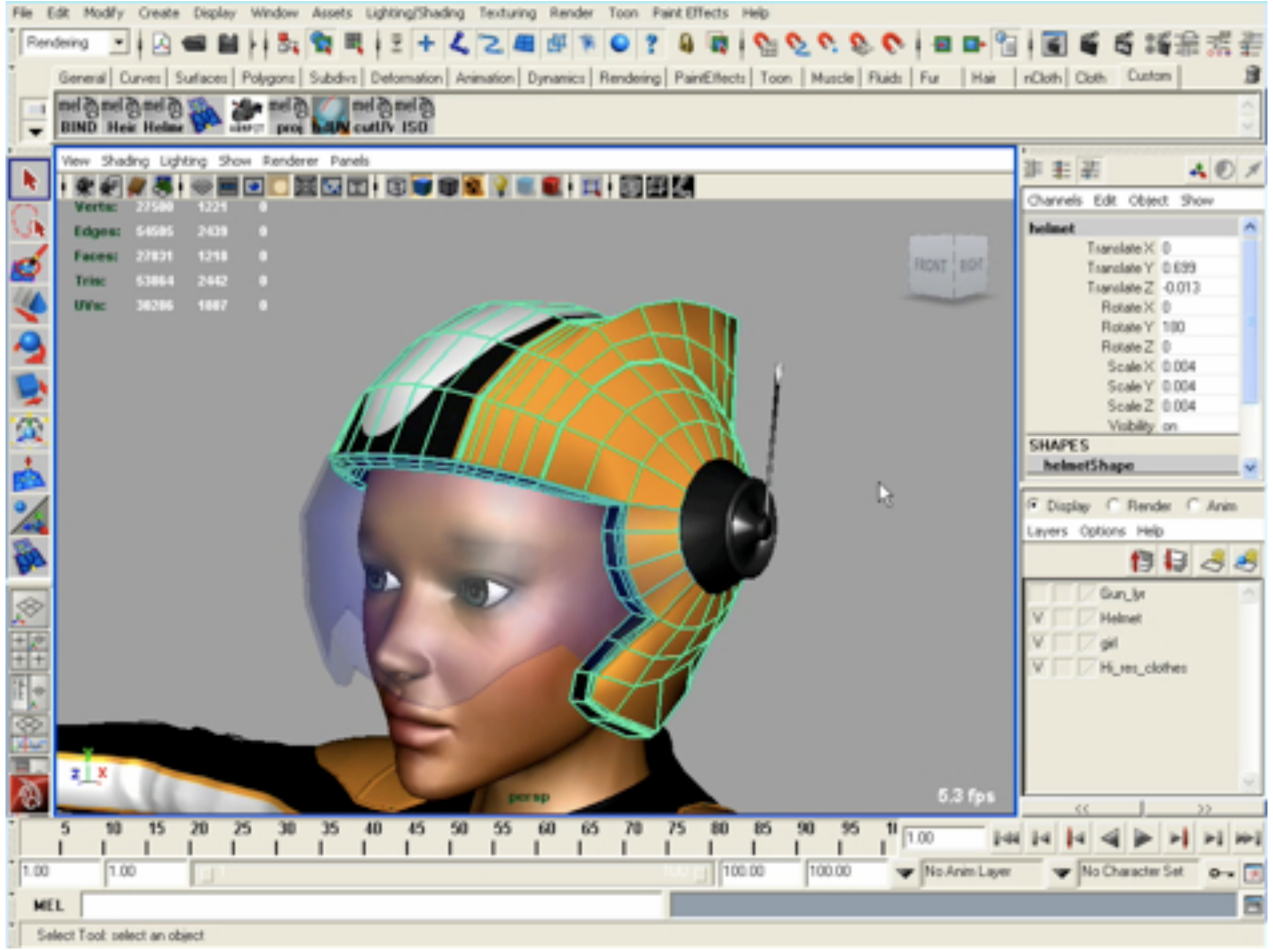


Pencil and Paper Sketches

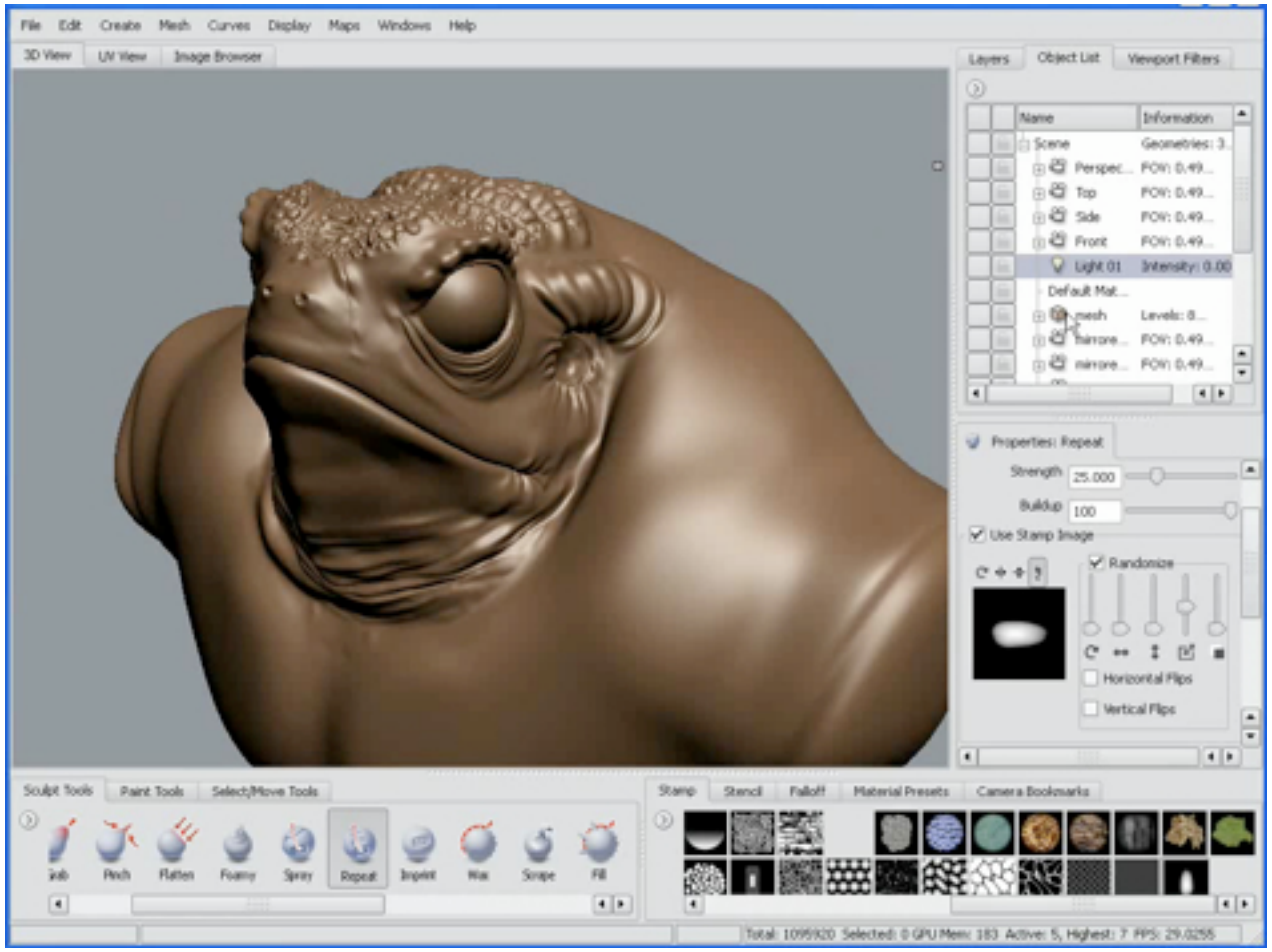


Traditional Expert Systems

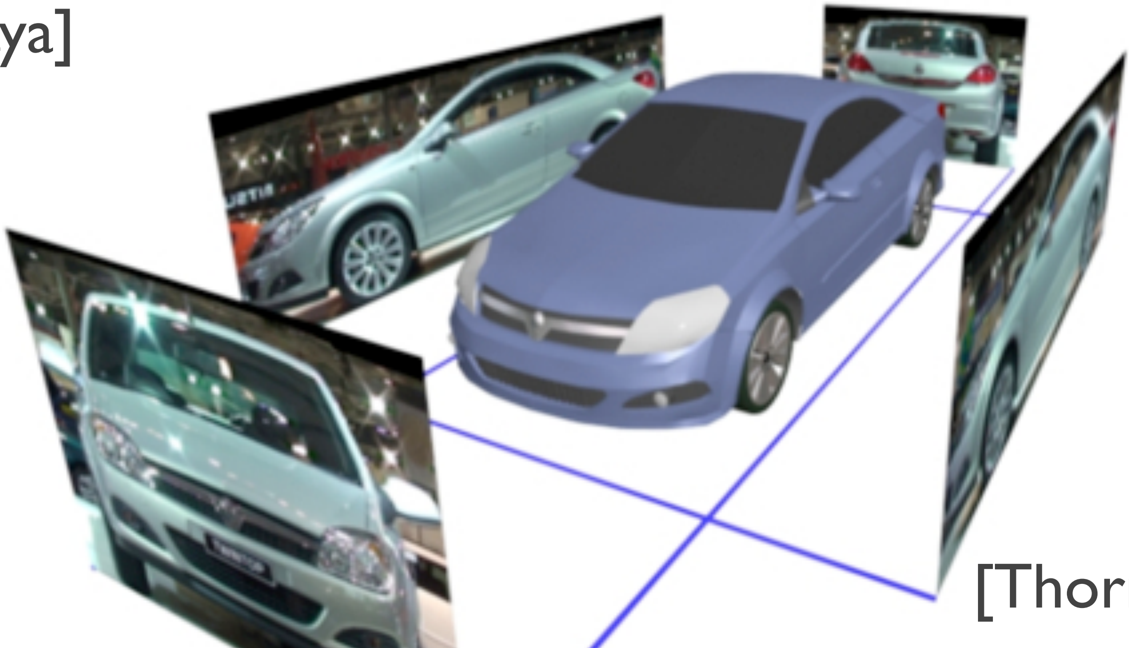
Concept artwork plays no direct role



[Maya]


File Edit Create Mesh Curves Display Maps Windows Help
3D View UV View Image Browser
Layers Object List Viewport Filters
Name Information
Scene Geometries: 3
Perspec... FOV: 0.49
Top FOV: 0.49
Side FOV: 0.49
Front FOV: 0.49
Light 01 Intensity: 0.00
Default Mat...
mesh Levels: 0
mirrore... FOV: 0.49
mirrore... FOV: 0.49
Properties: Repeat
Strength 25.000
Buildup 100
Use Stamp Image
Randomize
Horizontal Flips
Vertical Flips
Sculpt Tools Paint Tools Select/Move Tools
Sculpt Pinch Flatten Foamy Spray Repeat Imprint Wax Scrape Fill
Stamp Stencil Falloff Material Presets Camera Bookmarks
Total: 1095920 Selected: 0 GPU Mem: 183 Active: 5, Highest: 7 FPS: 29.0256

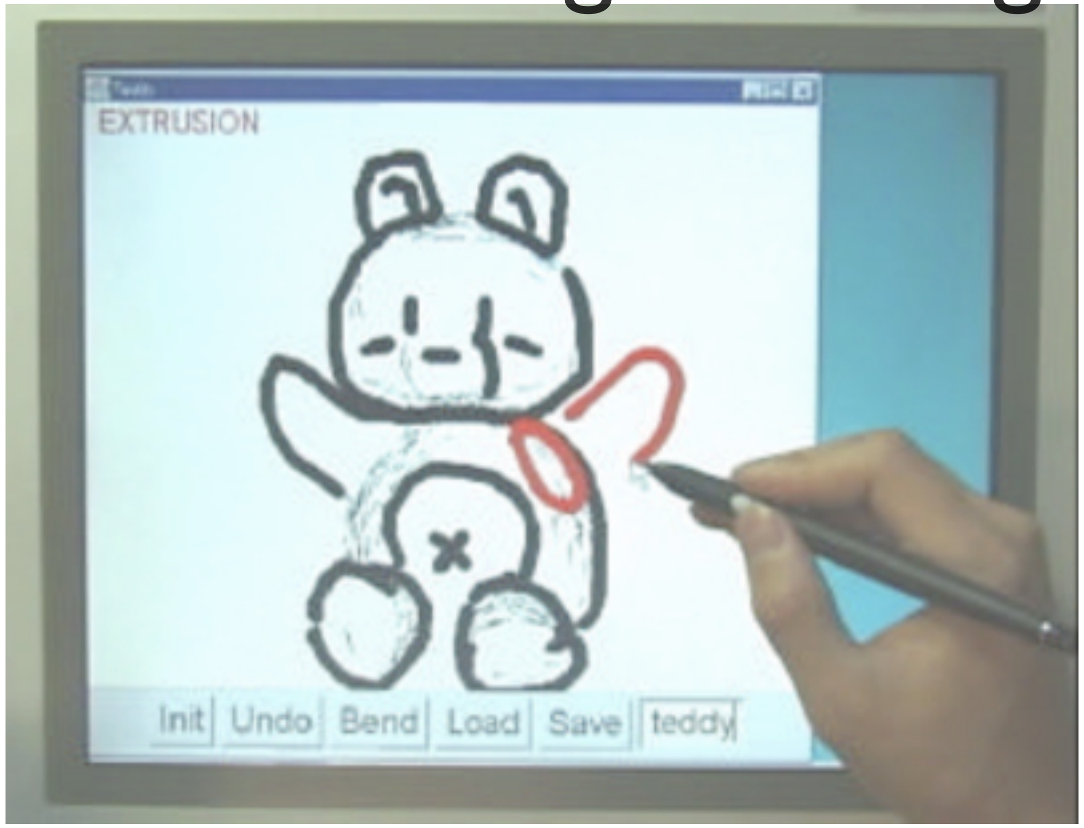
[Mudbox]



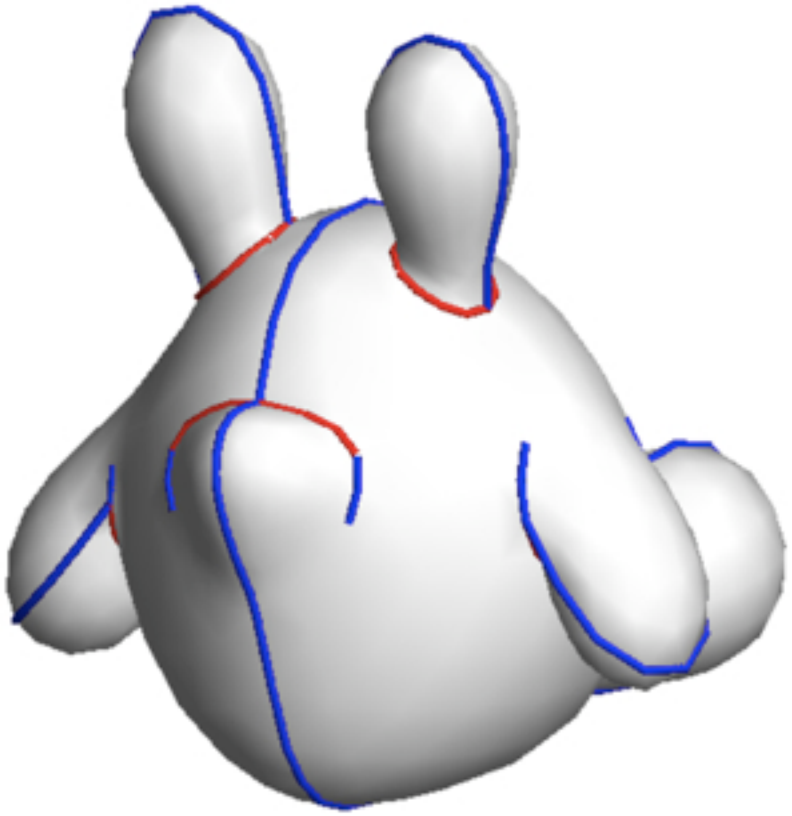
[Thormälen and Seidel 2008]

Novice Systems (See [Olsen et al. 2008] for a recent survey.)

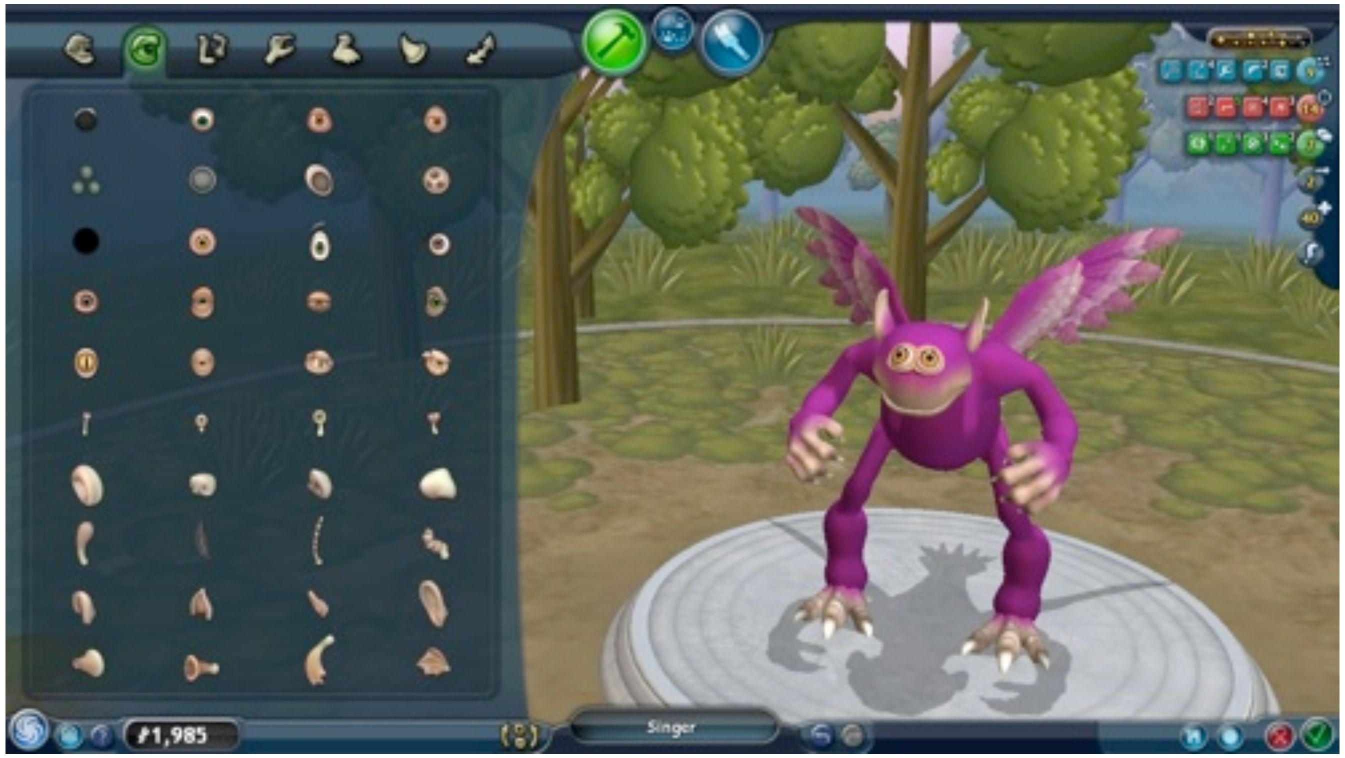
Can’t trace a guide image



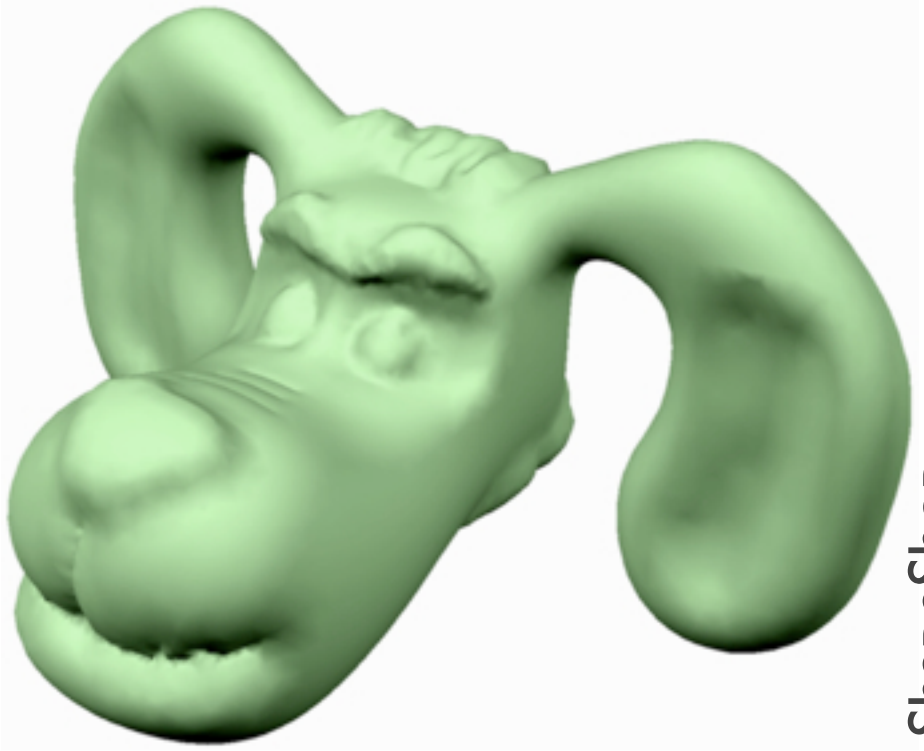
Teddy [Igarashi et al. 1999]



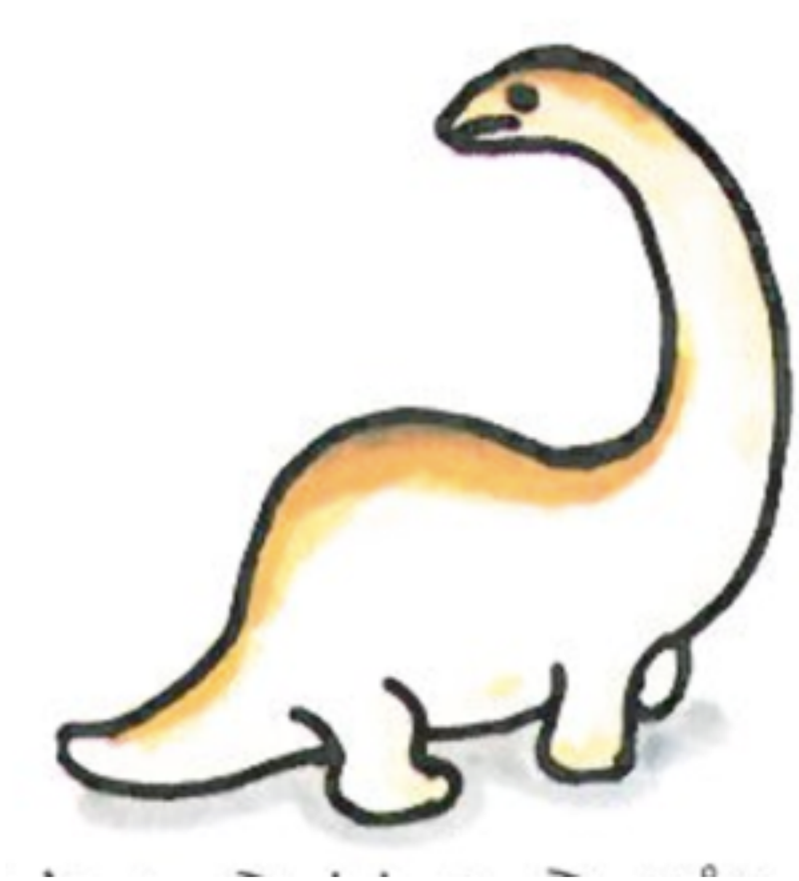
FiberMesh [Nealen et al. 2007]



Spore [Maxis 2008]

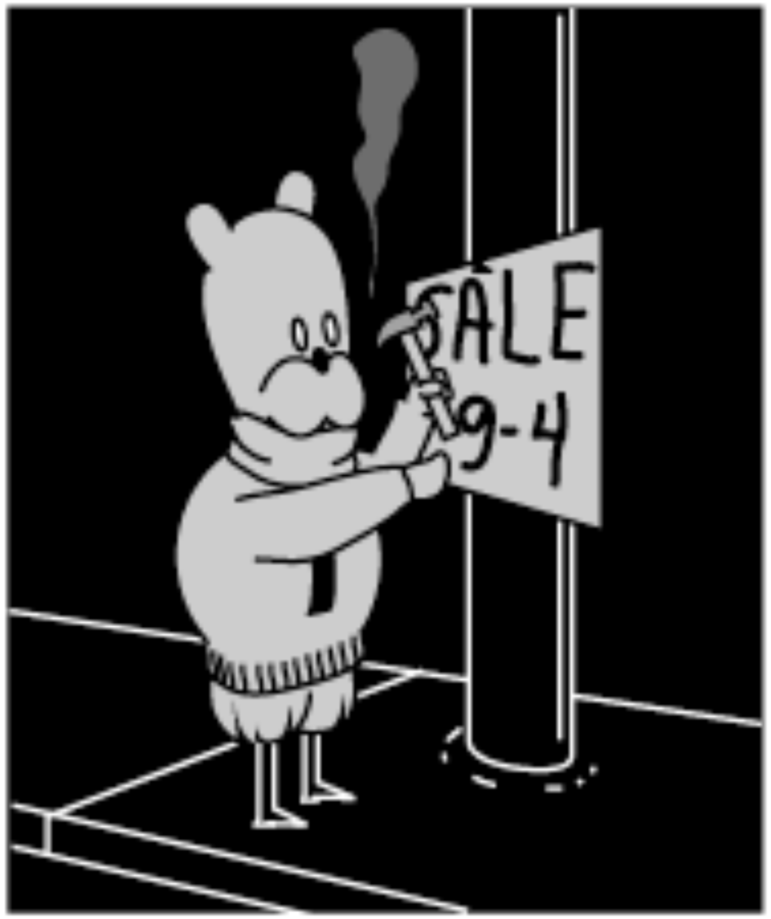


ShapeShop [Schmidt et al. 2005-8]



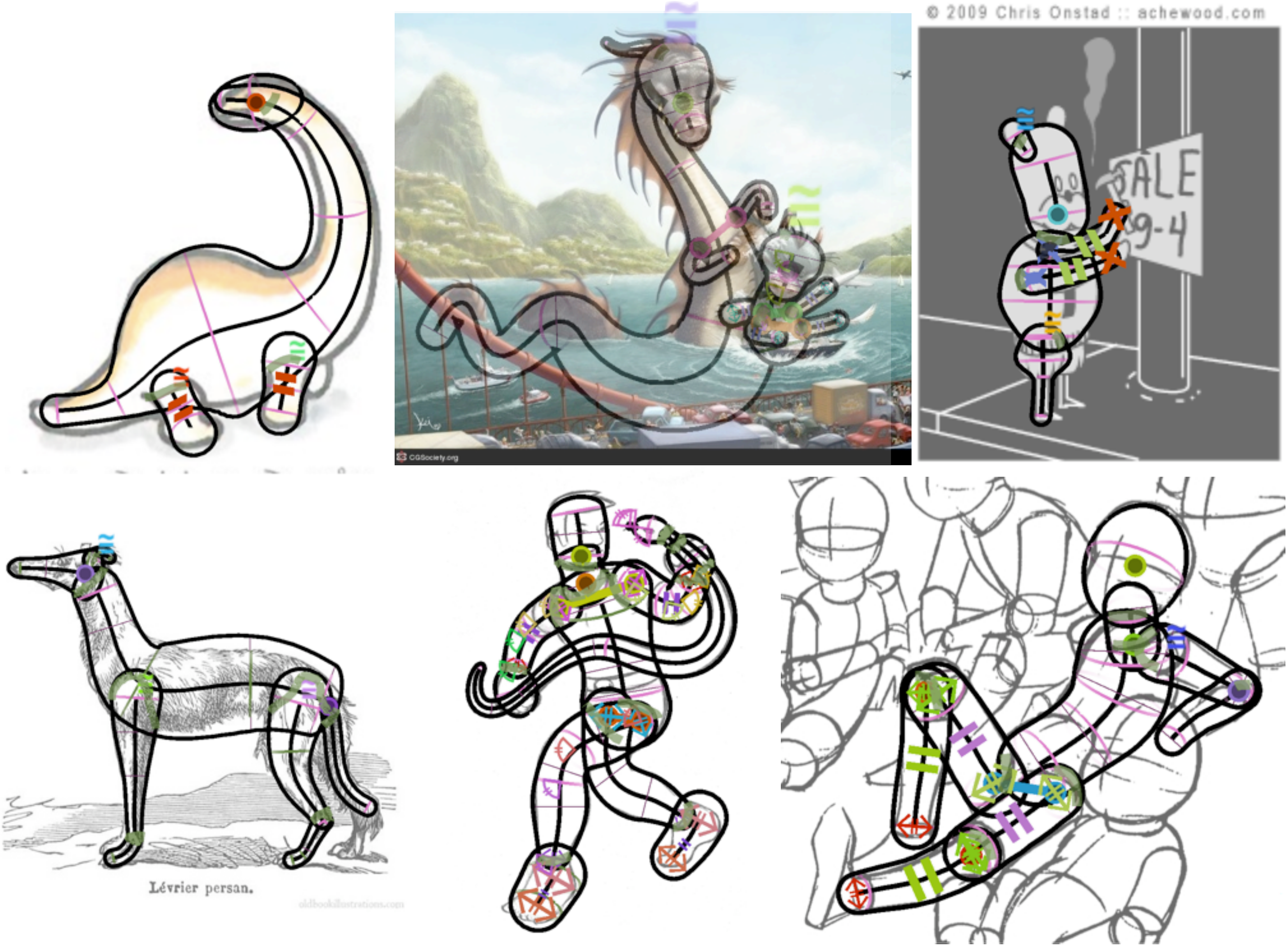
CGSociety.org

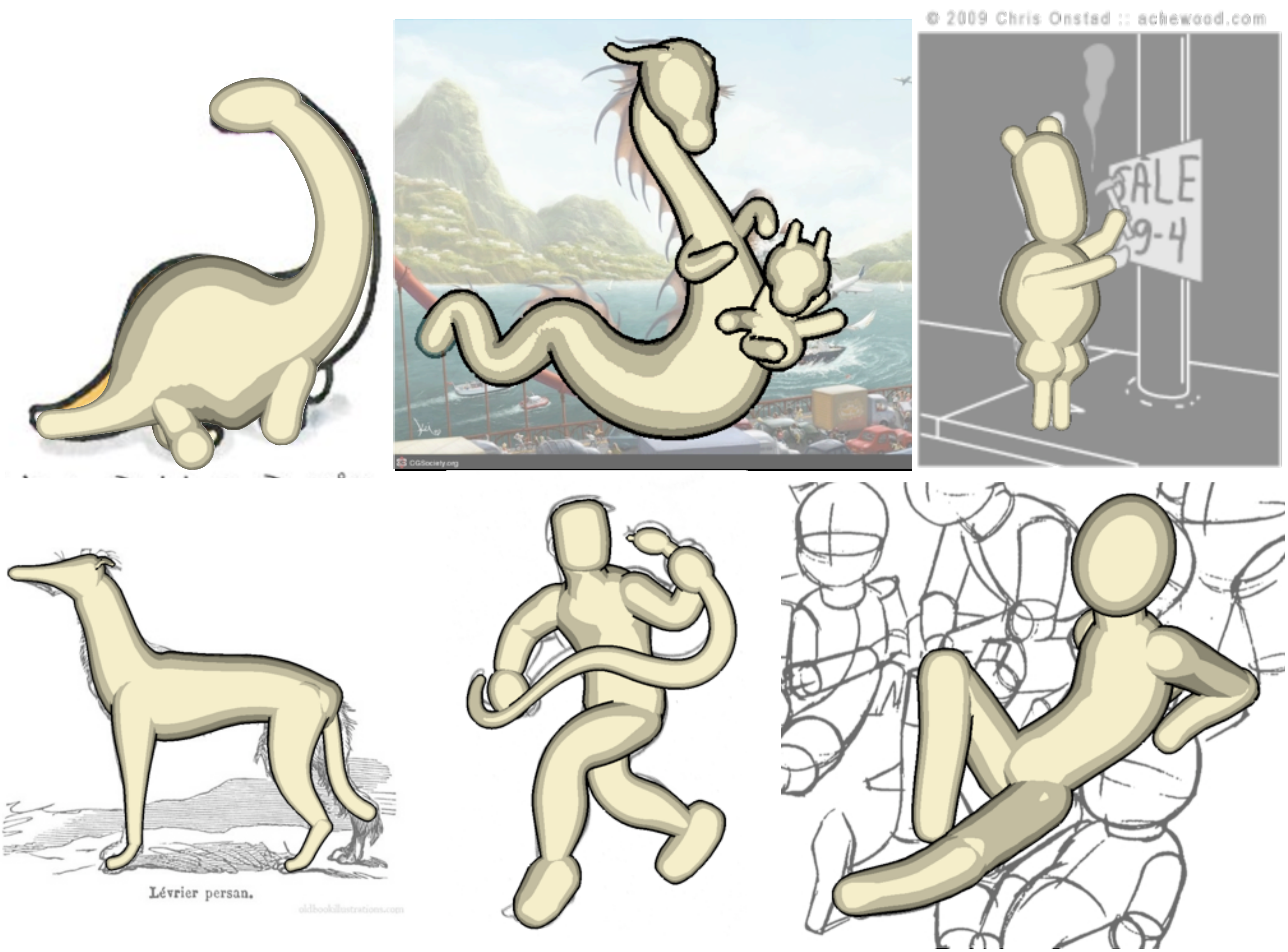
© 2009 Chris Onstad :: achewood.com



Lévrier persan.

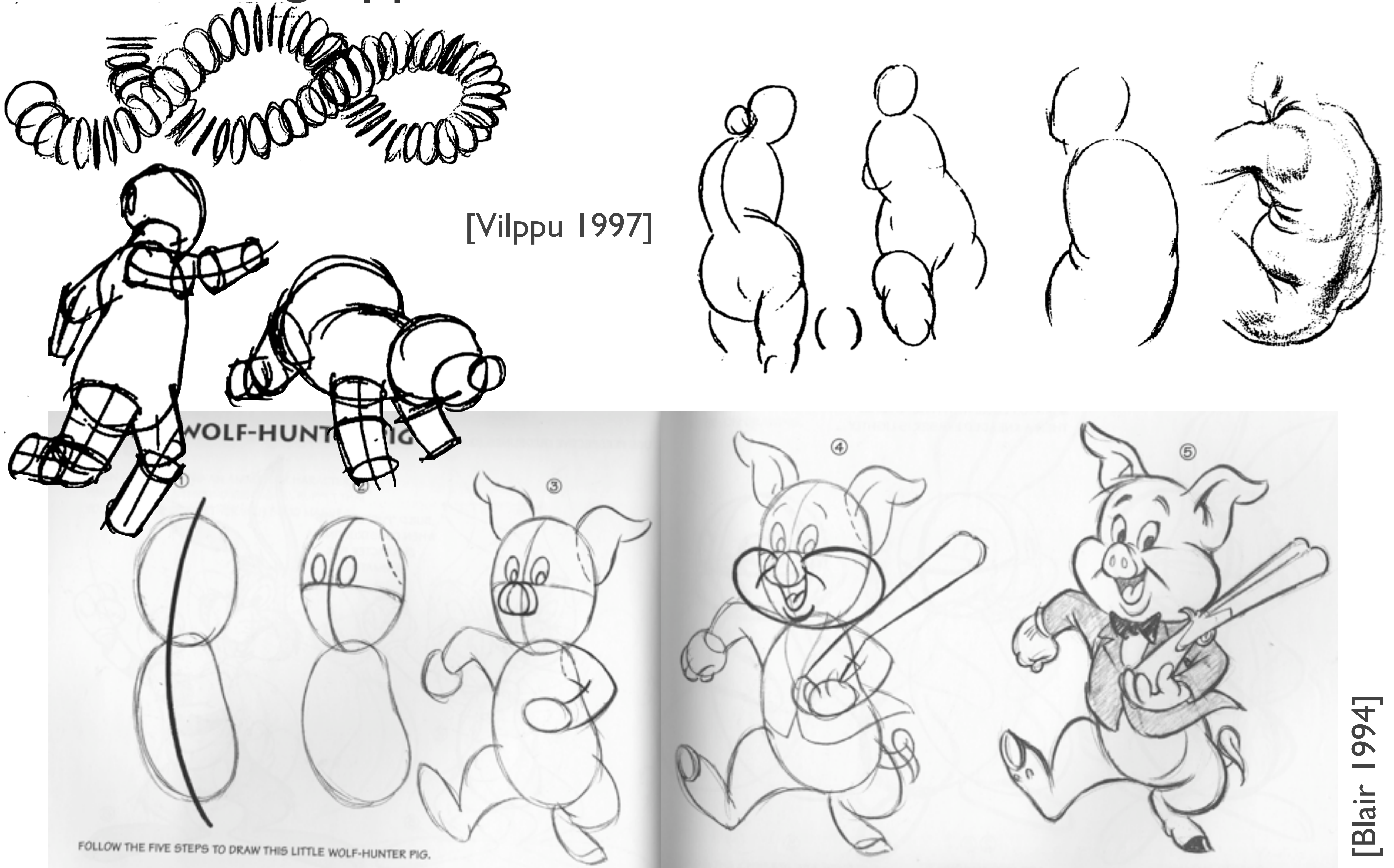
oldbookillustrations.com


© 2009 Chris Onstad :: achewood.com
SALE
9-4
CGSociety.org
Lévrier persan.
oldbookillustrations.com


CGSociety.org
© 2009 Chris Onstad :: achewood.com
SALE
9-4
Lévrier persan.
oldbookillustrations.com

Inspiration

2D Drawing Approaches

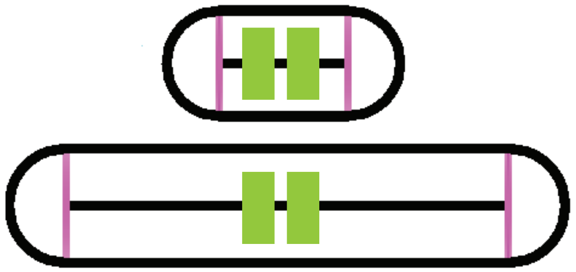


[Vilppu 1997]

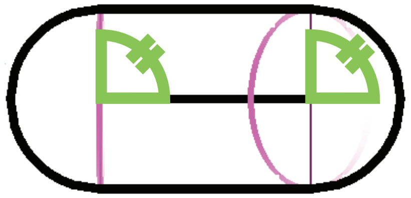
[Blair 1994]

Annotations

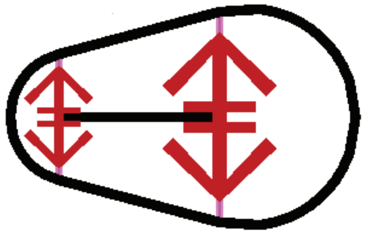
Same-length



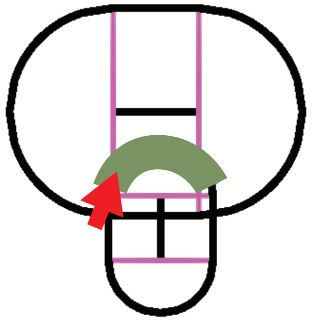
Same-tilt



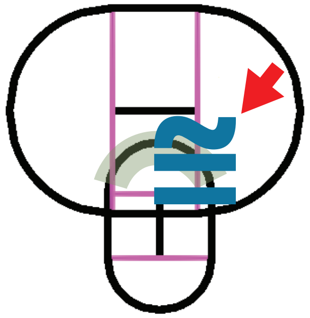
Same-scale



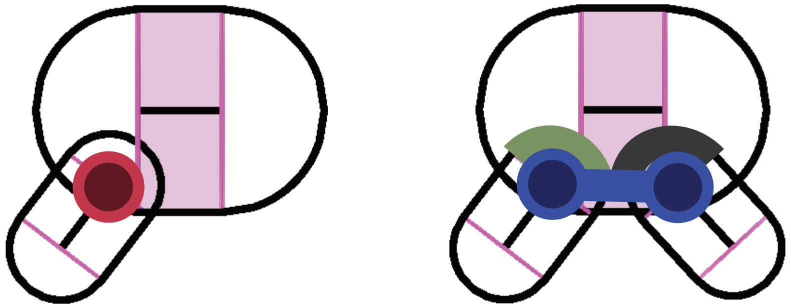
Connection curve



Mirror



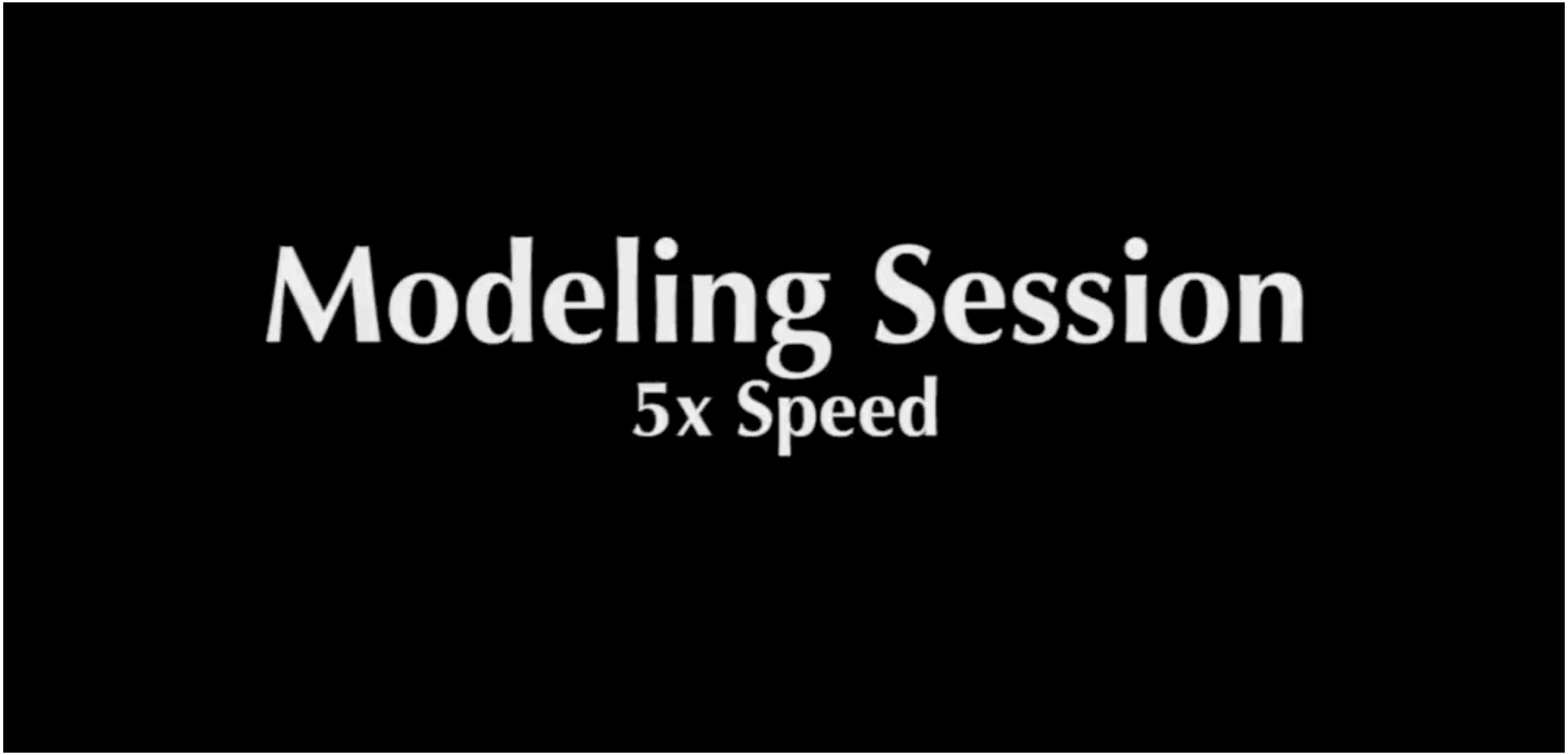
Alignment



Demo

Guide image [Vilppu 1997]

Demo

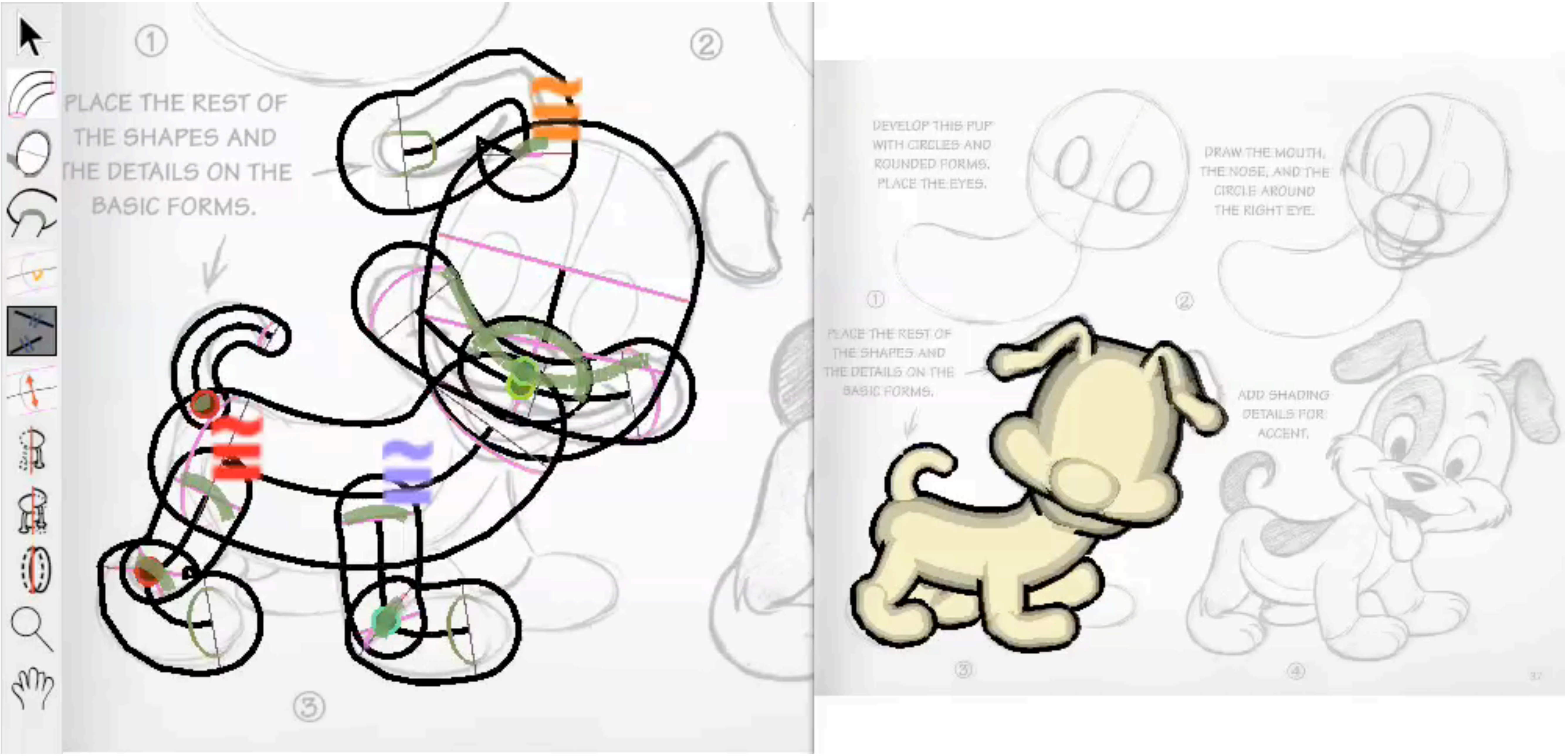


Guide image [Vilppu 1997]

Results

Guide images: [Blair 1994]; © Alex Rosmarin; © Kei Acedera, Imaginism Studios 2008; © Björn Hurri, www.bjornhurri.com; © Alex Rosmarin; © Alex Rosmarin; [Kako 1973]; [Kako 1973]

Results

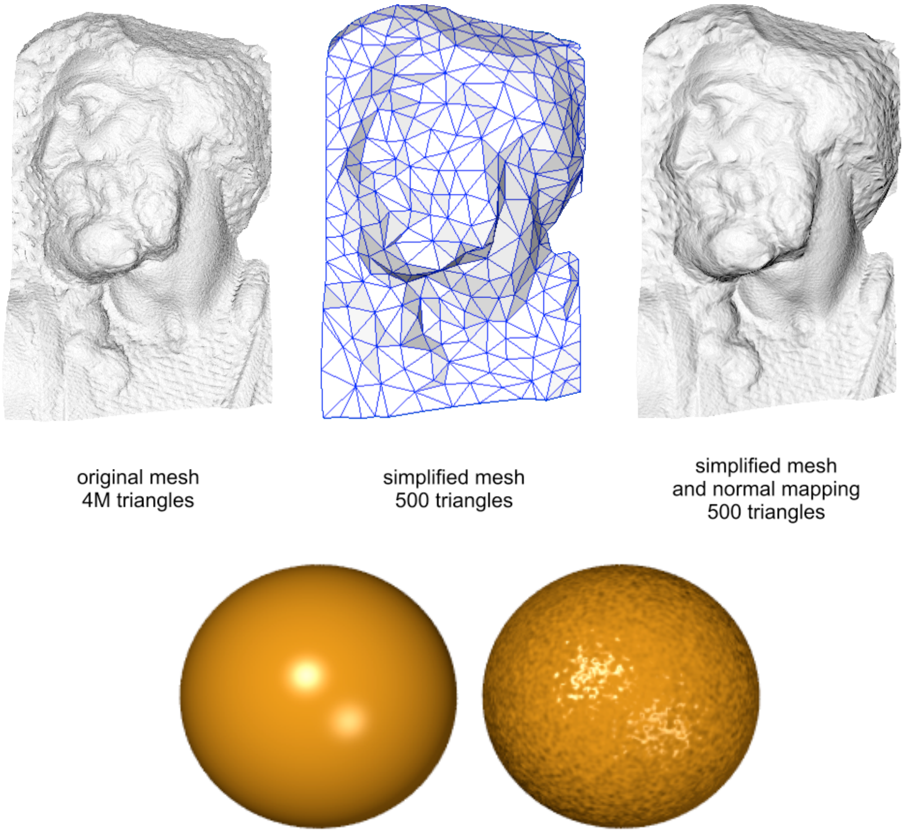

1
PLACE THE REST OF THE SHAPES AND THE DETAILS ON THE BASIC FORMS.
2
3
DEVELOP THIS PUP WITH CIRCLES AND ROUNDED FORMS. PLACE THE EYES.
DRAW THE MOUTH, THE NOSE, AND THE CIRCLE AROUND THE RIGHT EYE.
ADD SHADING DETAILS FOR ACCENT.
4

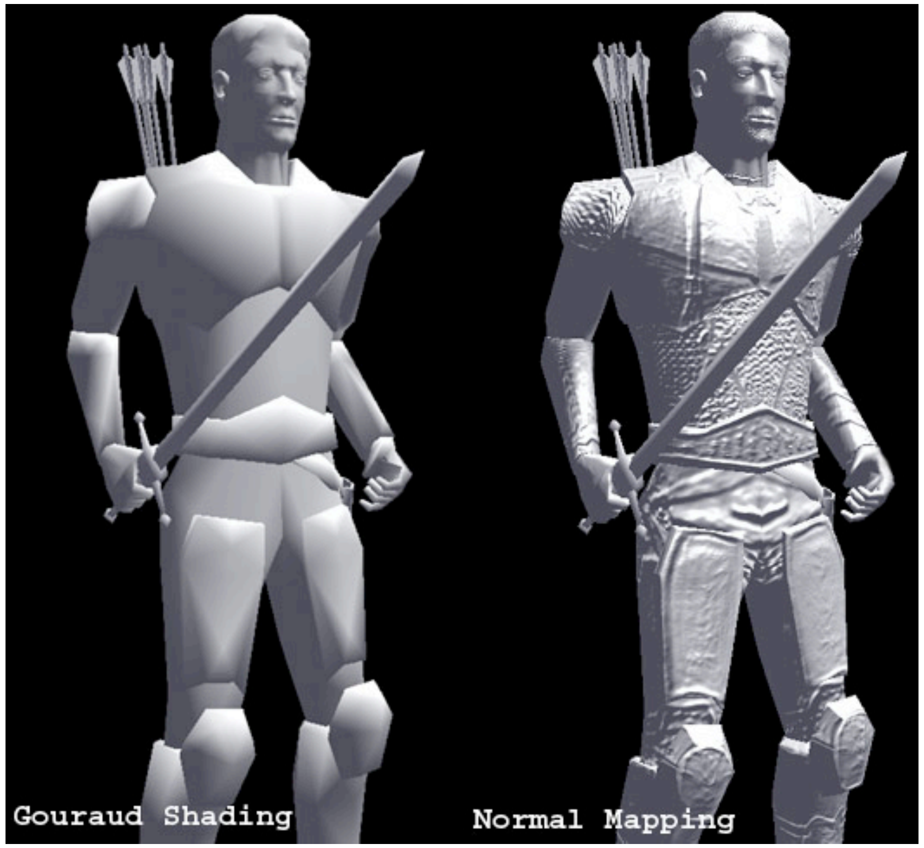
Guide images: [Blair 1994]; © Alex Rosmarin; © Kei Acedera, Imaginism Studios 2008; © Björn Hurri, www.bjornhurri.com; © Alex Rosmarin; © Alex Rosmarin; [Kako 1973]; [Kako 1973]

A Direct Texture Placement and Editing Interface

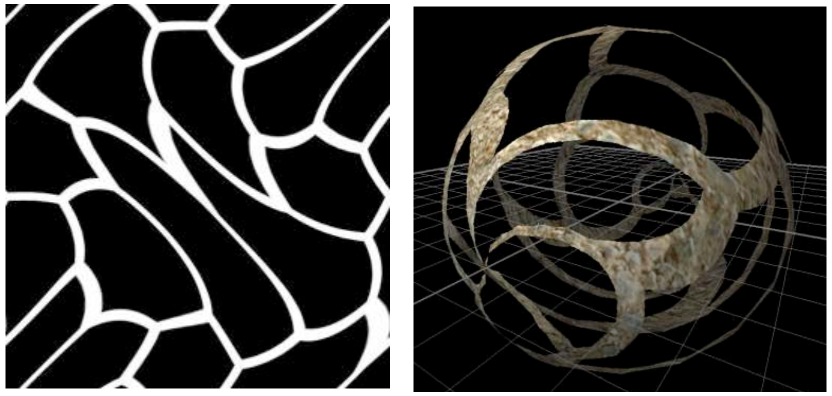
Textures

Color Map

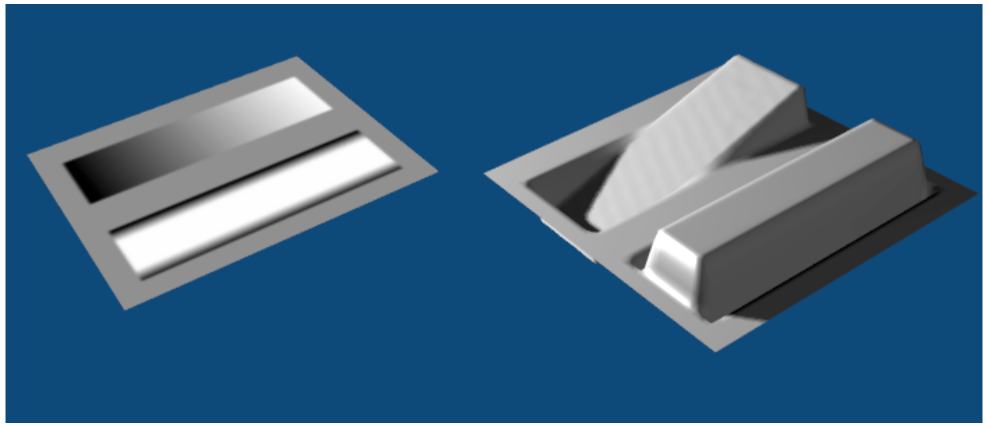




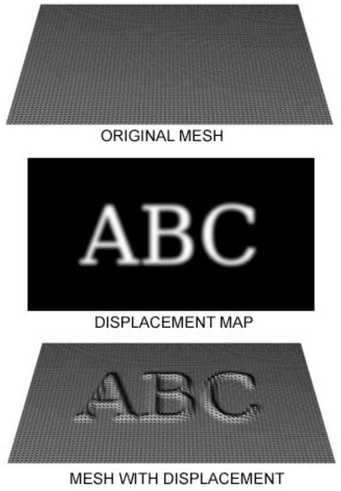
Normal Map



Alpha Map



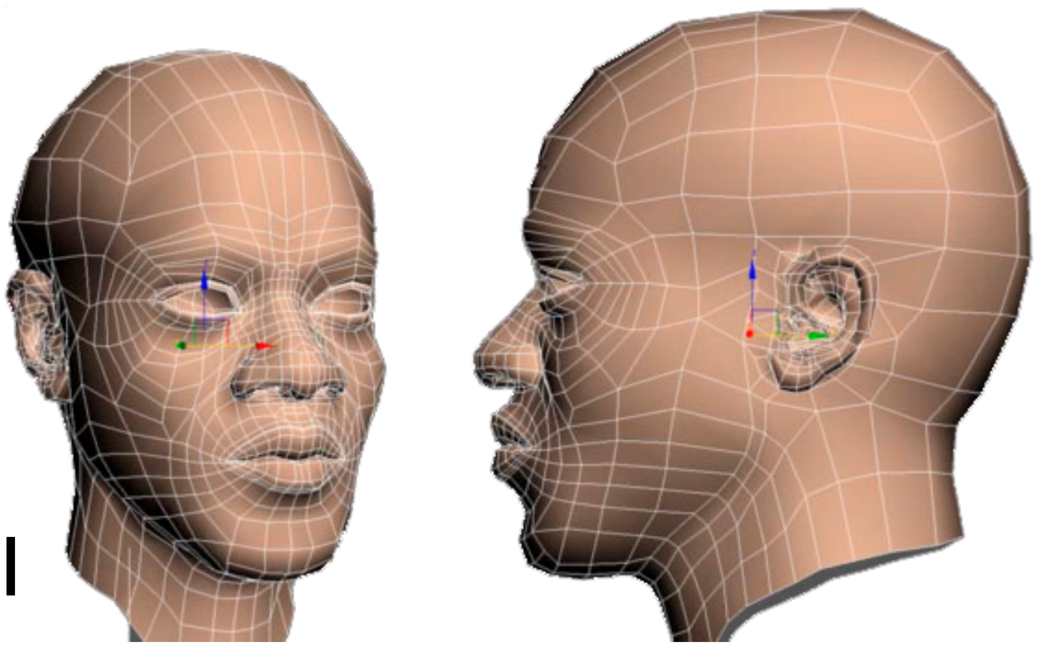
Displacement Map



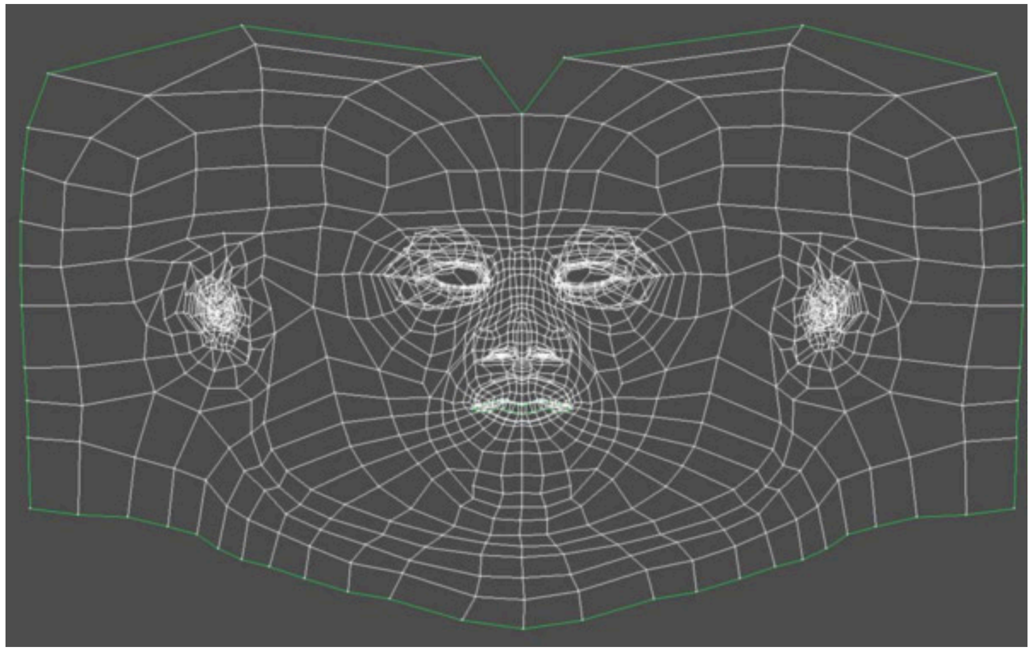
First Approach to Texturing

Jiri’s Texturing Tutorial

[Jiri Adamec]

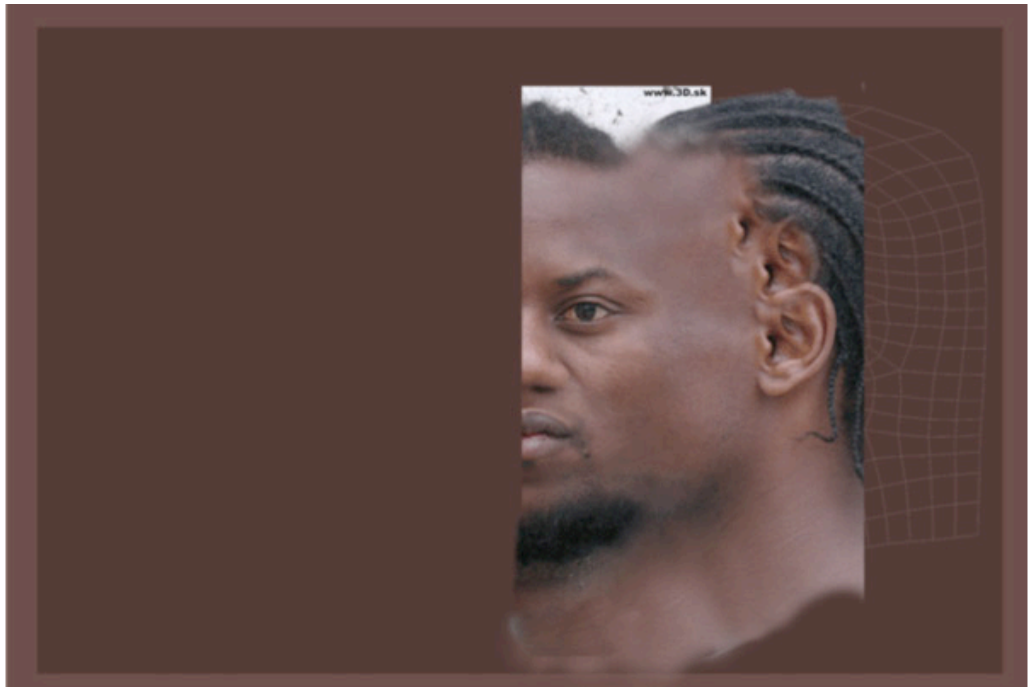


1



2

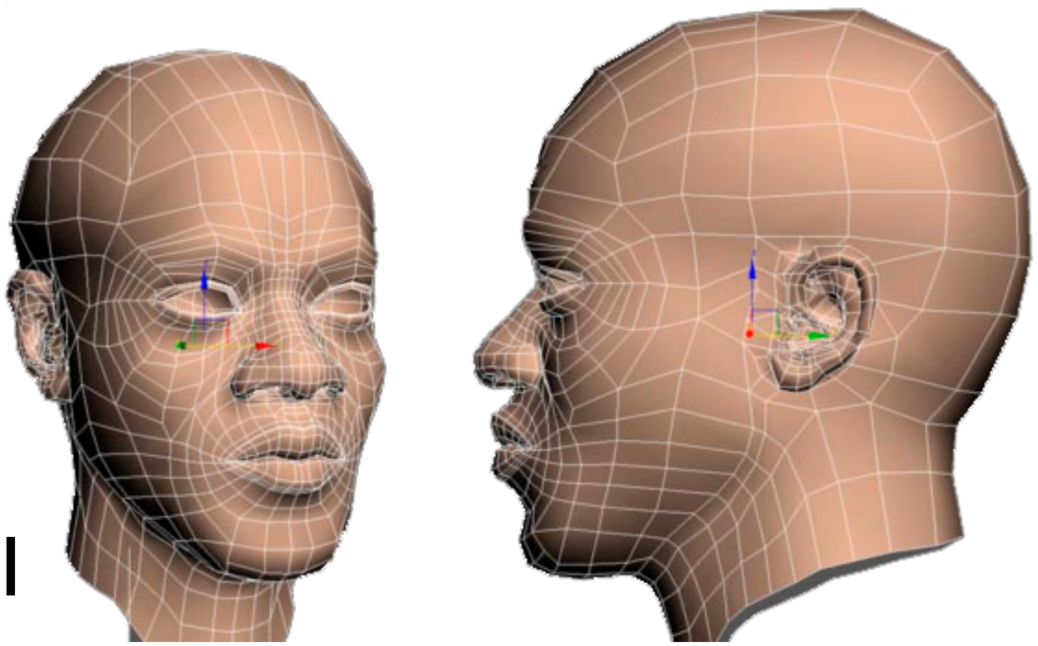
3



4

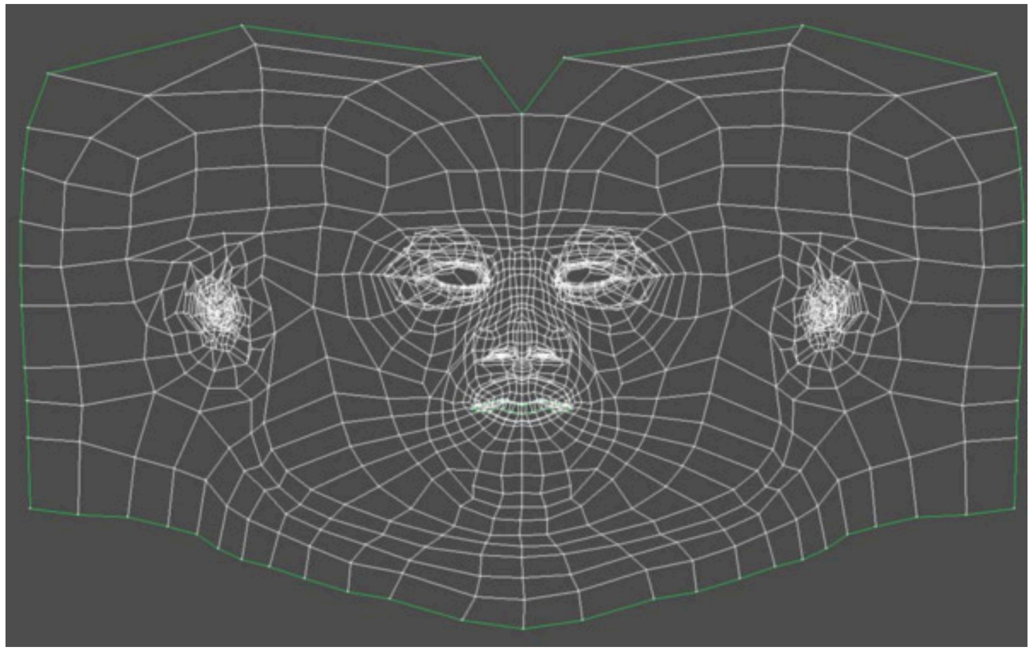
Technical Digression

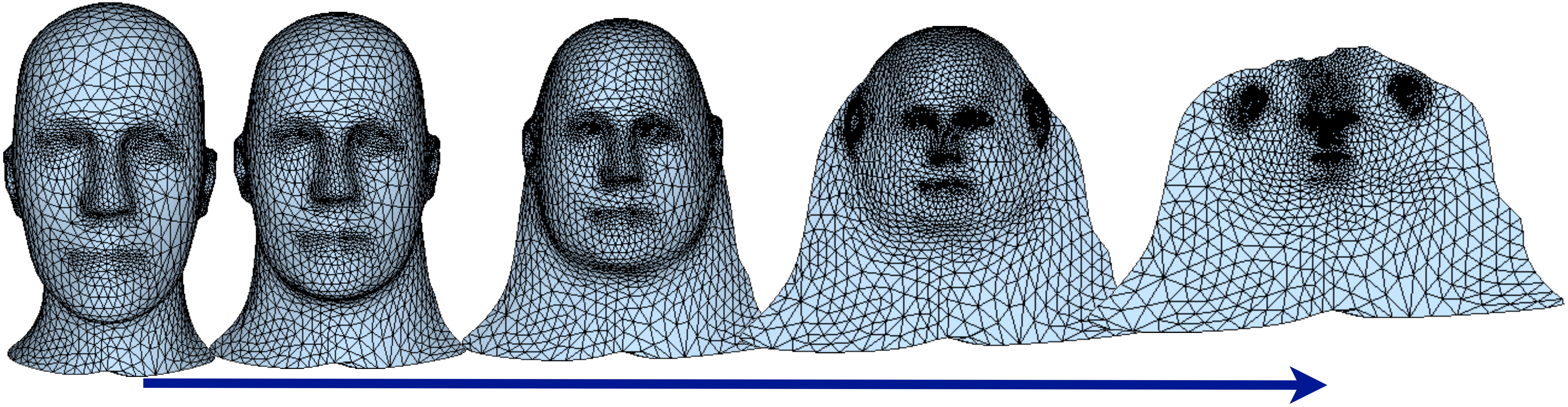
Flattening



1

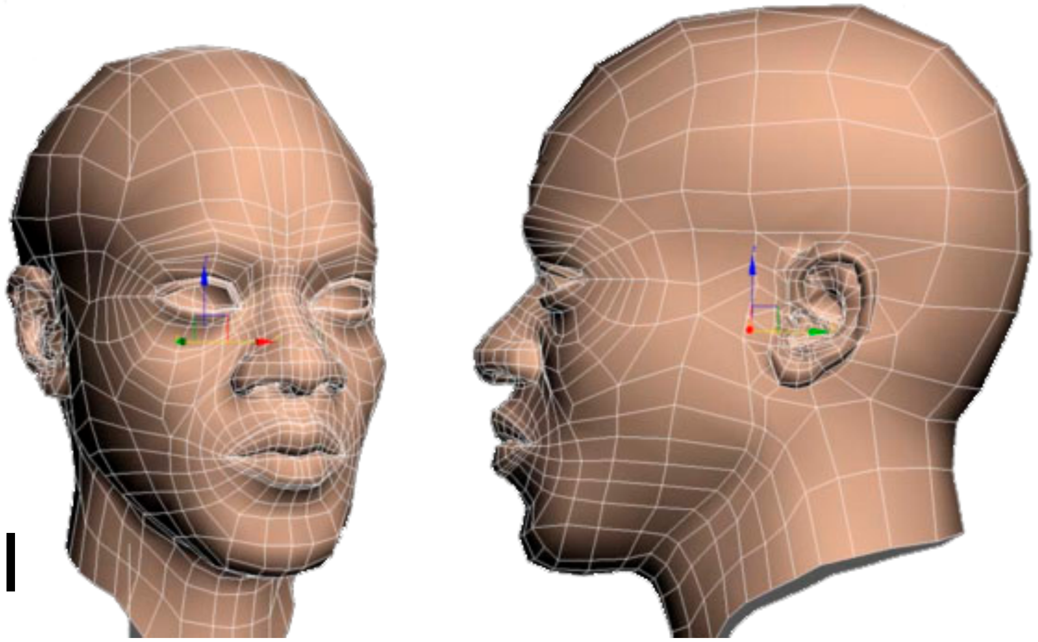
2



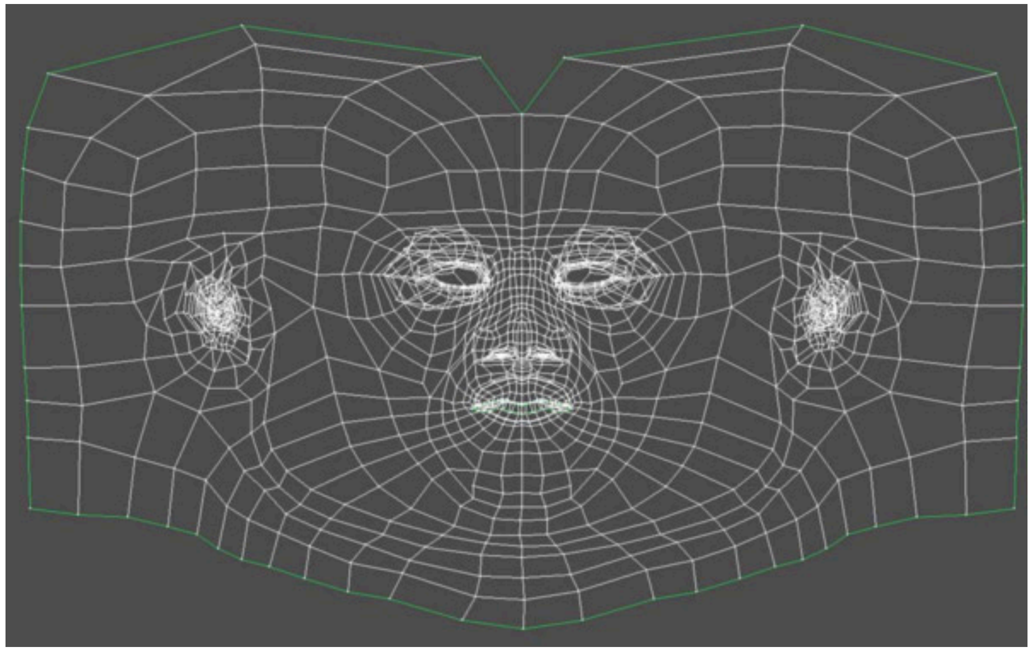


Jiri's Texturing Tutorial

[Jiri Adamec]

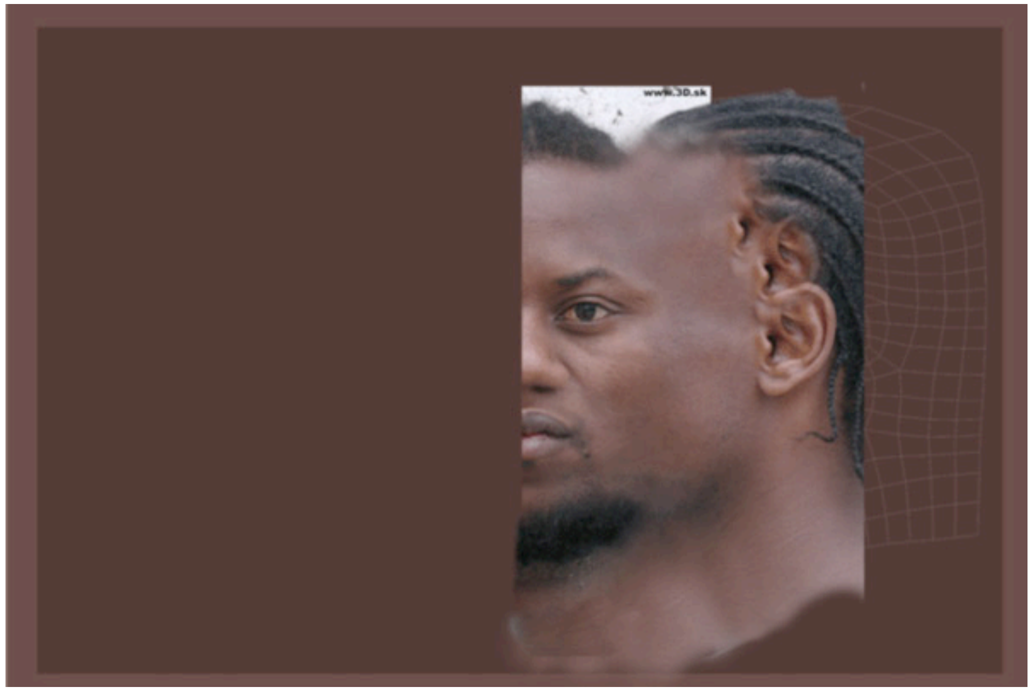


1



2

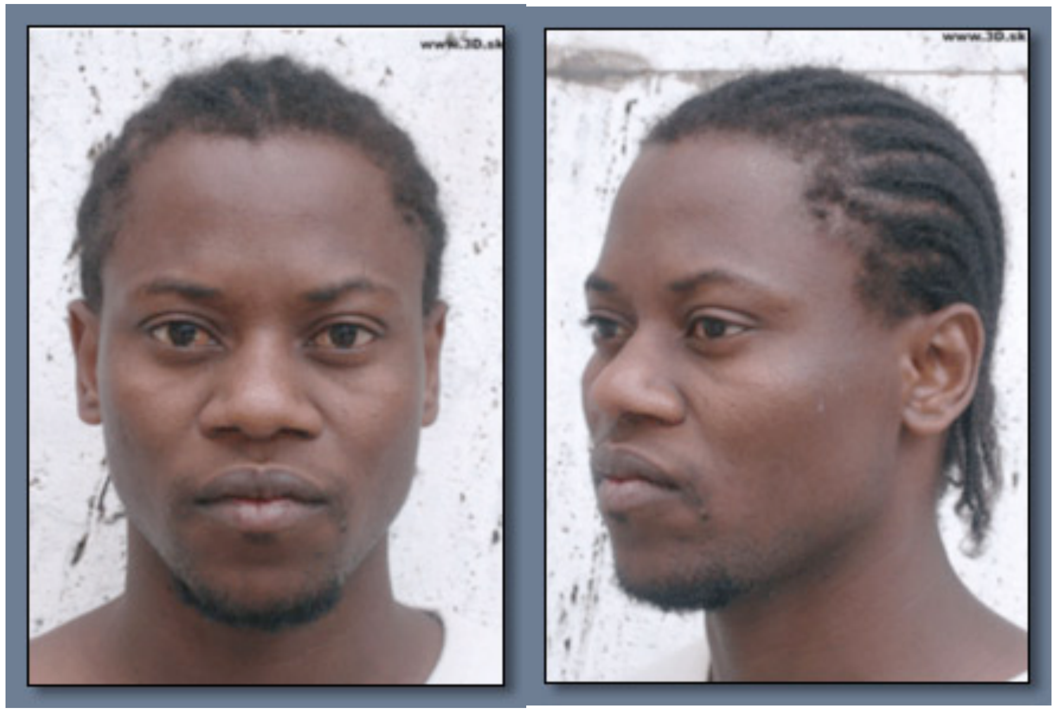
3



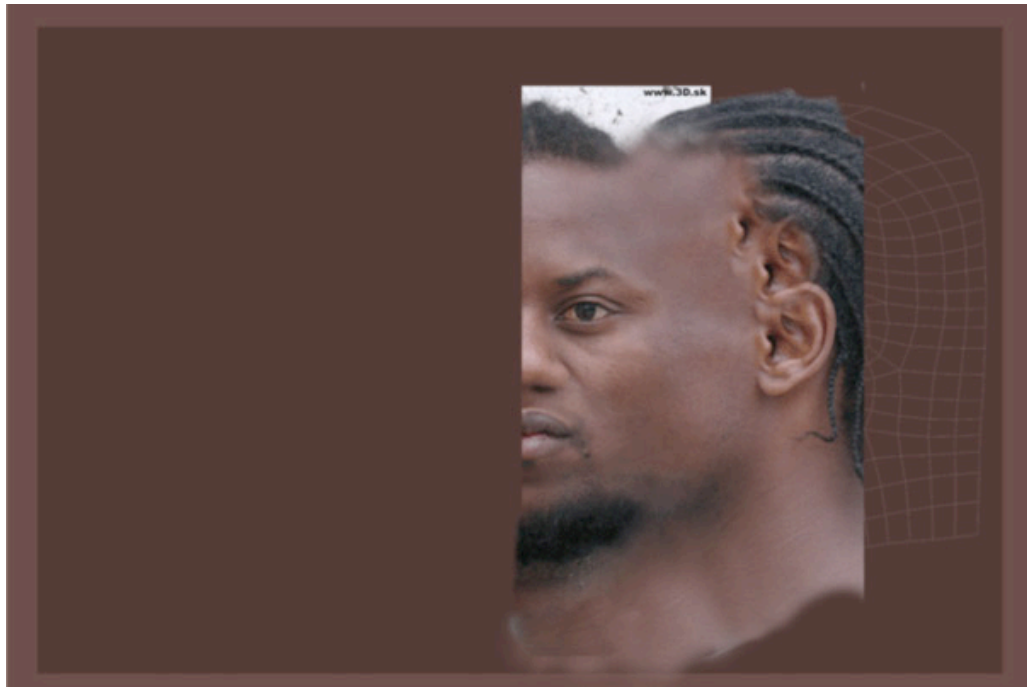
4

Jiri's Texturing Tutorial

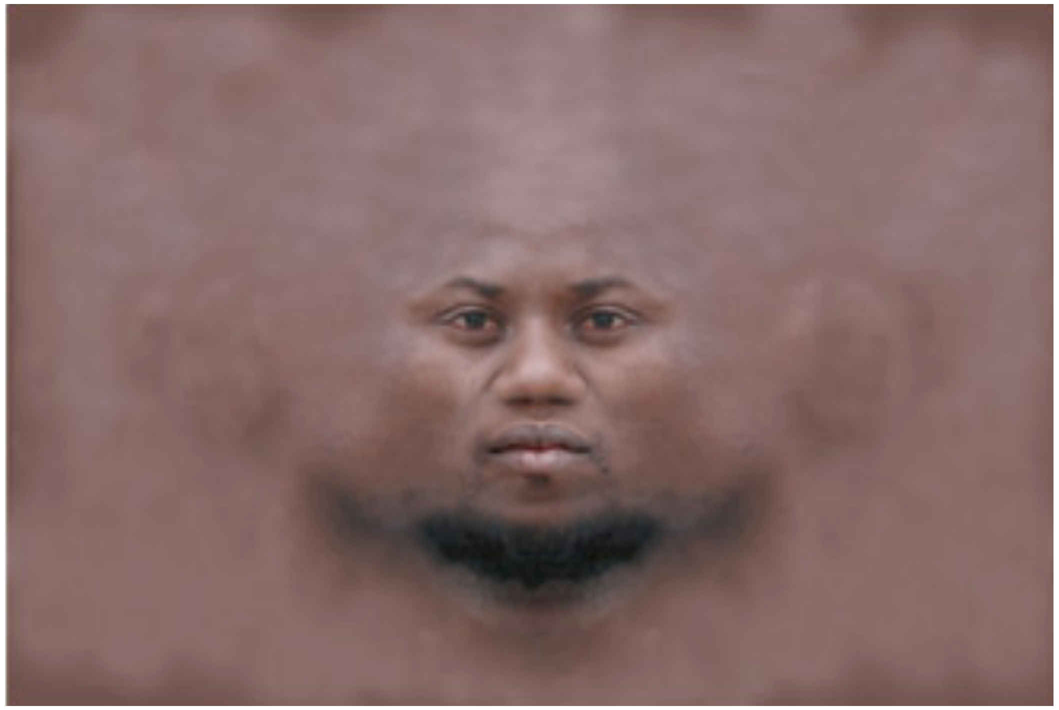
[Jiri Adamec]



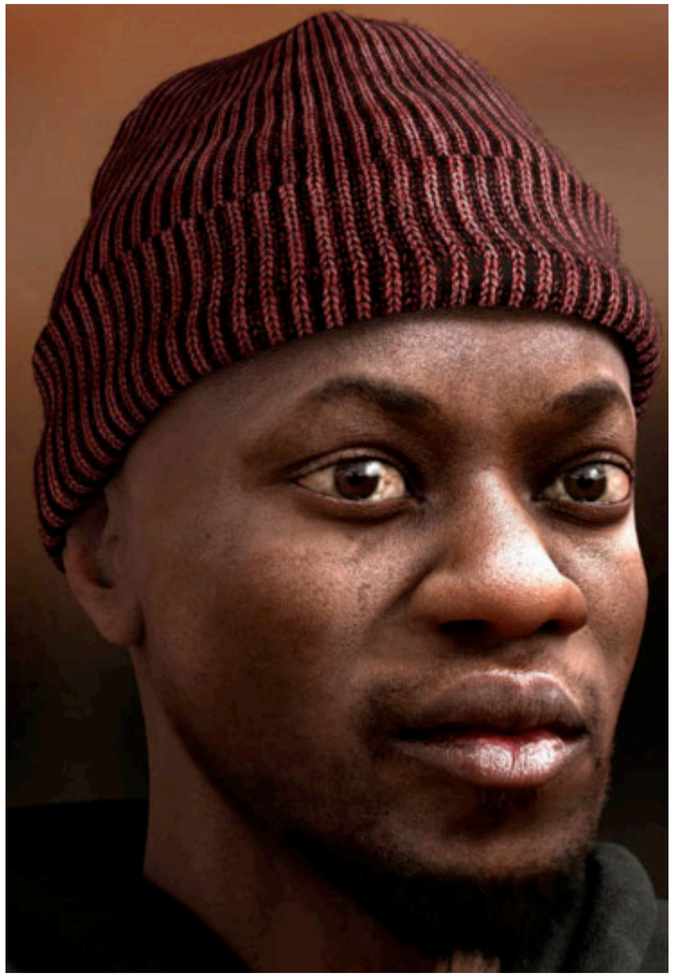
3



4



5

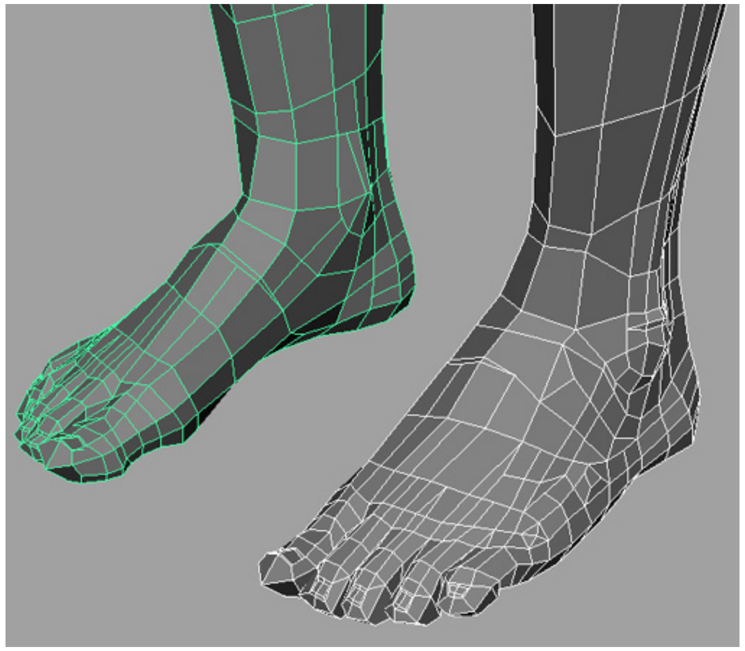


6

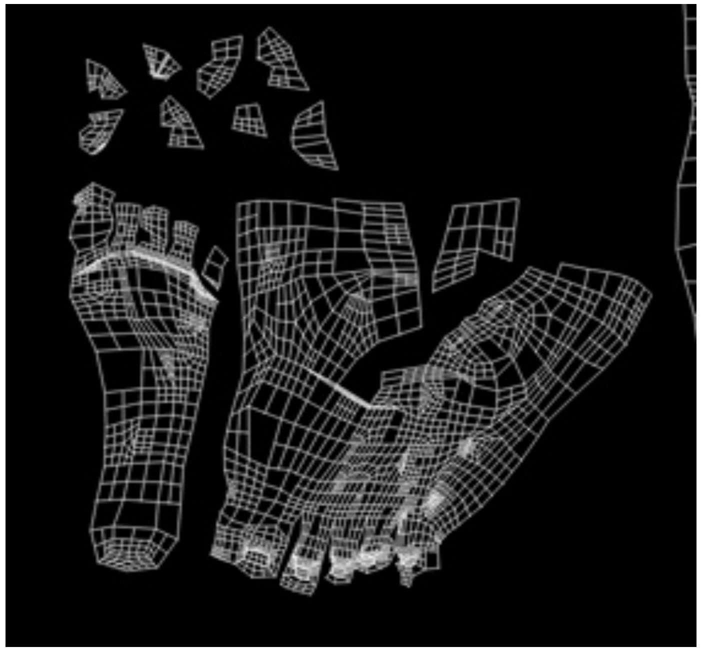
Feet Texturing Tutorial

[Steven Stahlberg]

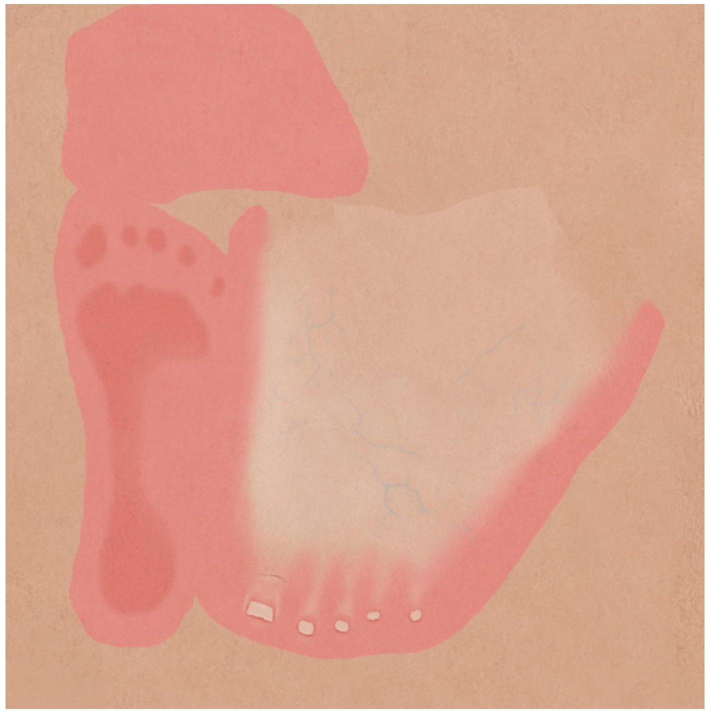
1



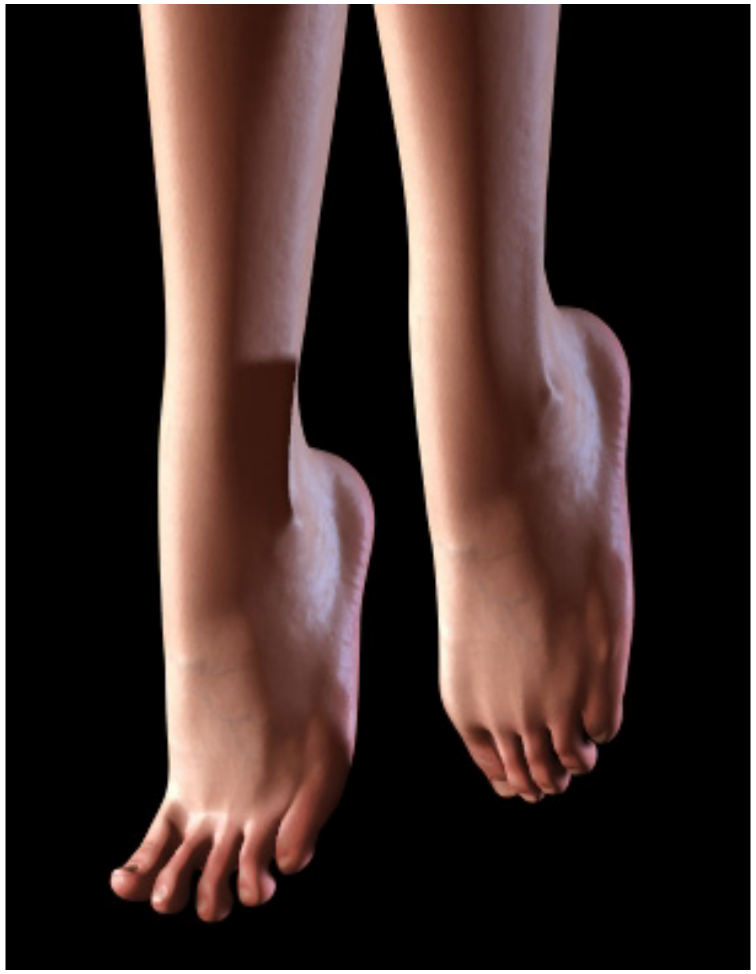
2



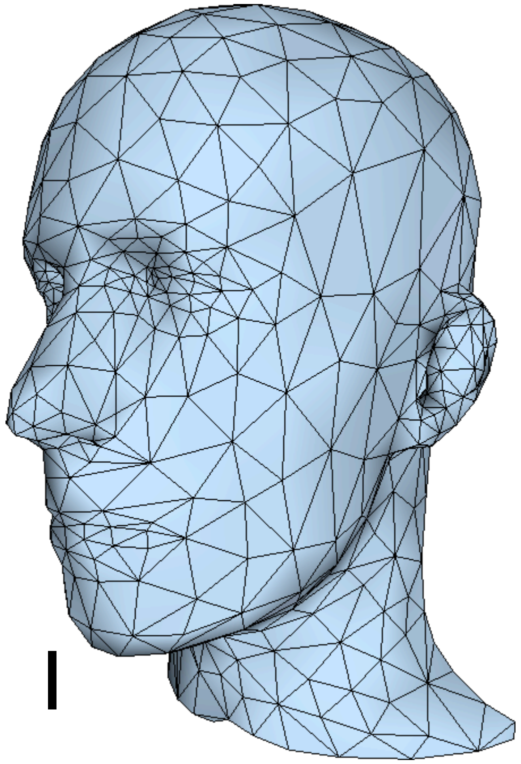
3



4

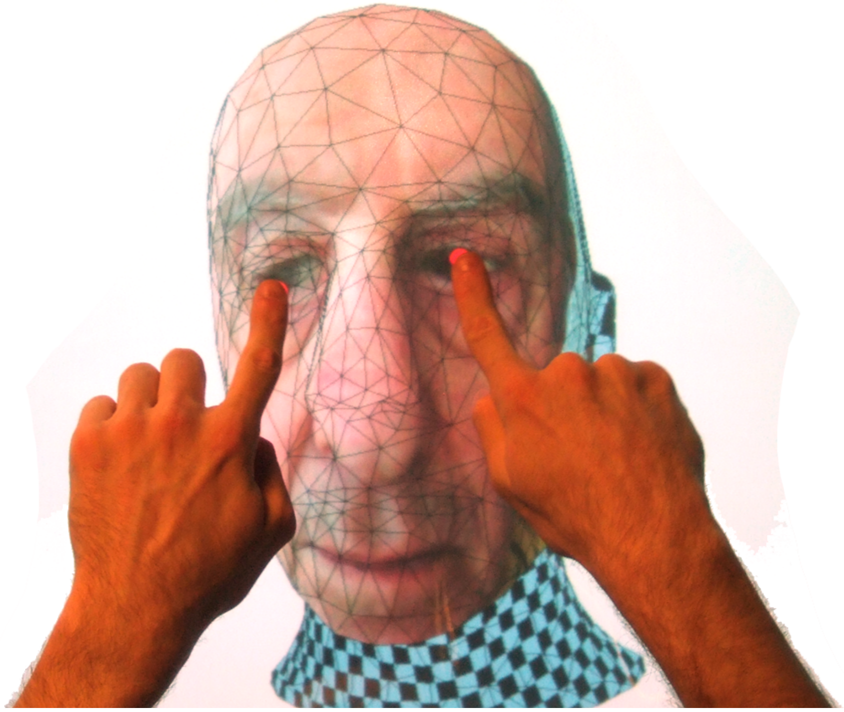


Our Approach to Texturing

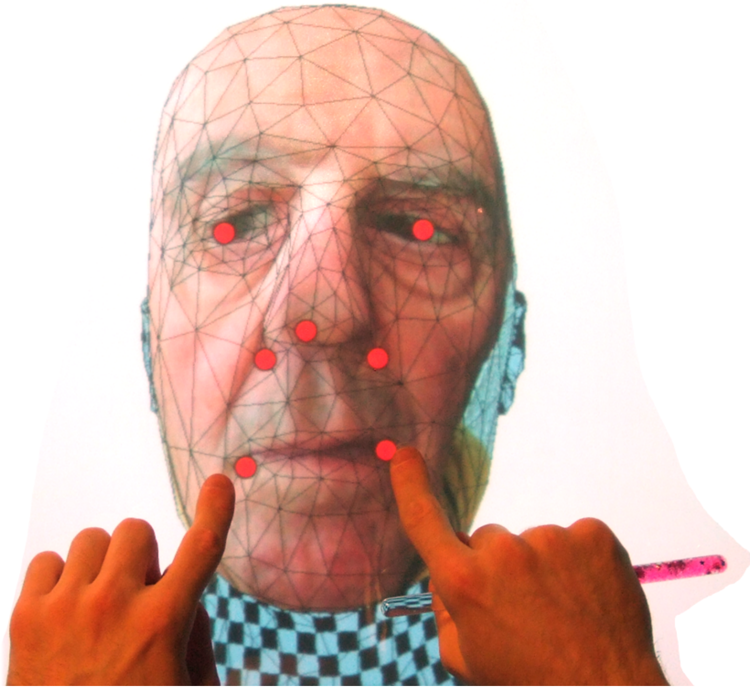


1

2



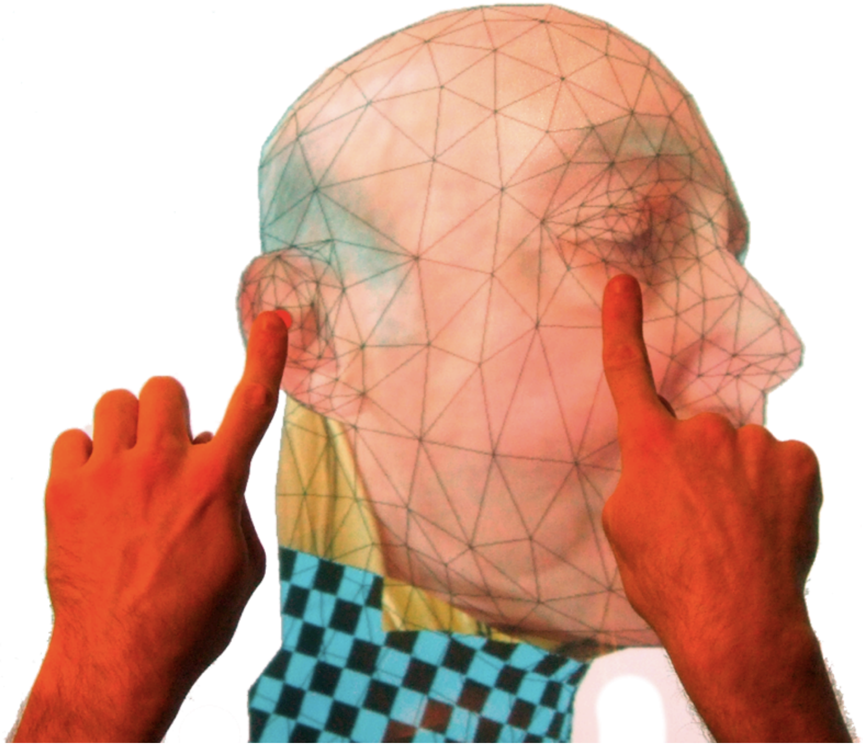
3



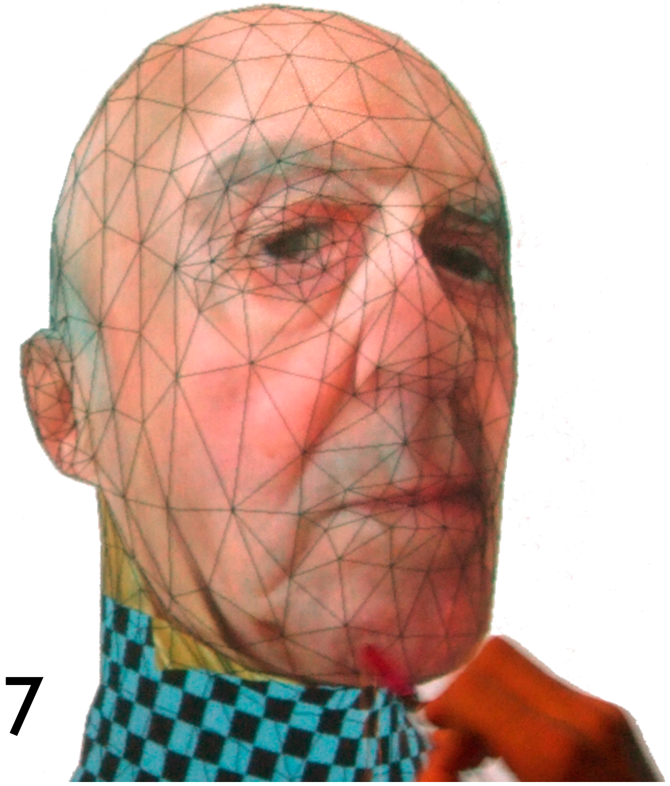
4

5

6



7

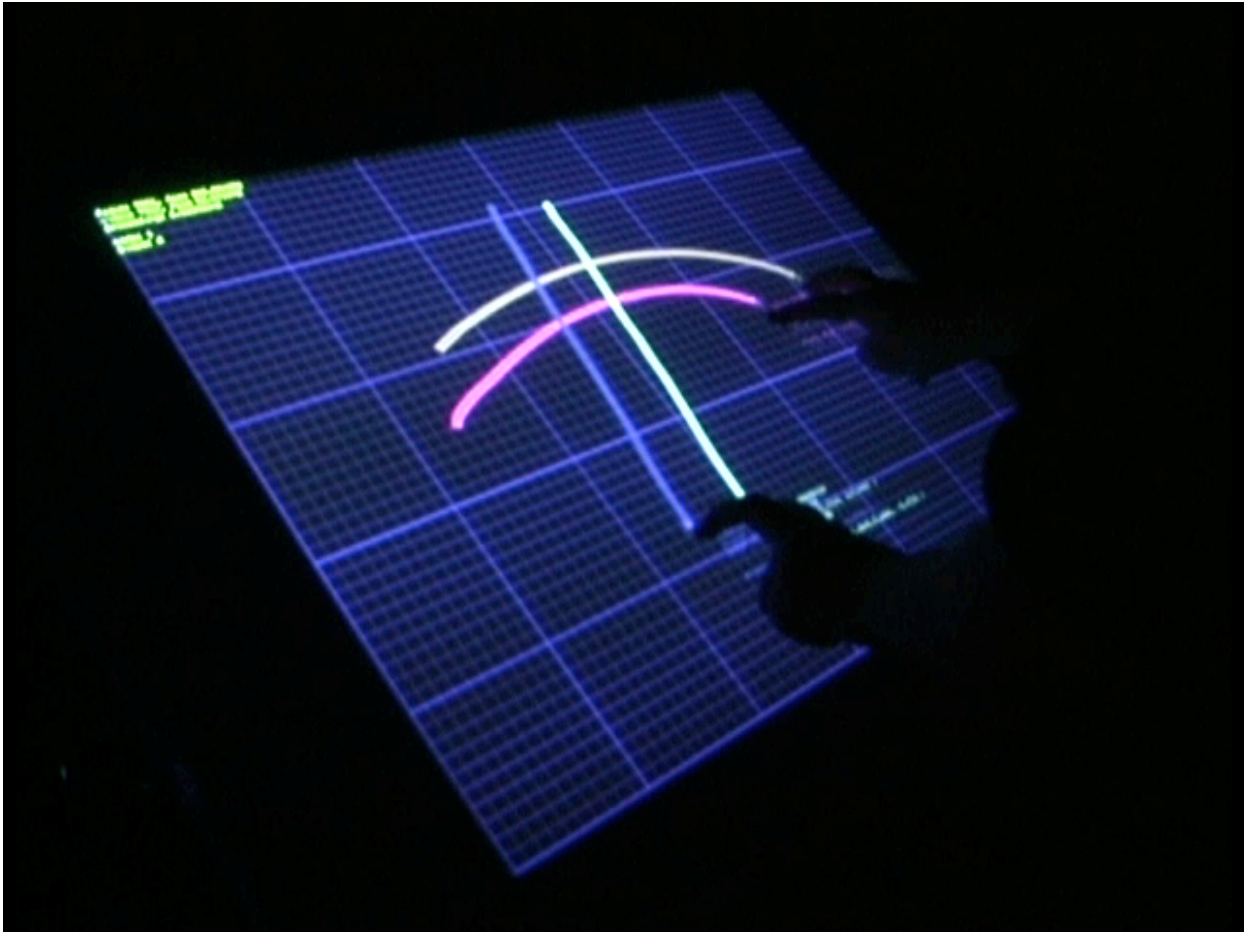


...

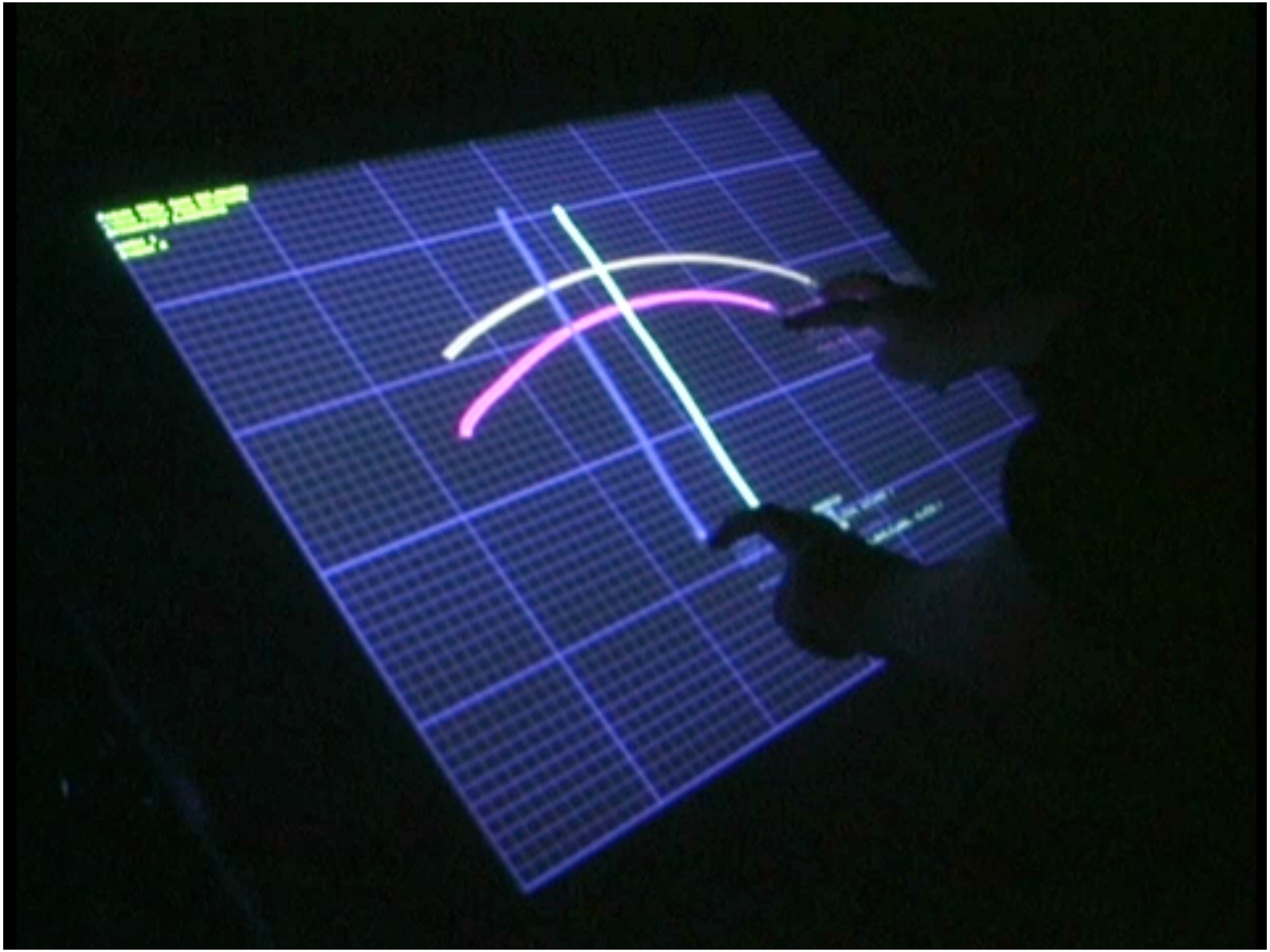
Repeat

7 Operations

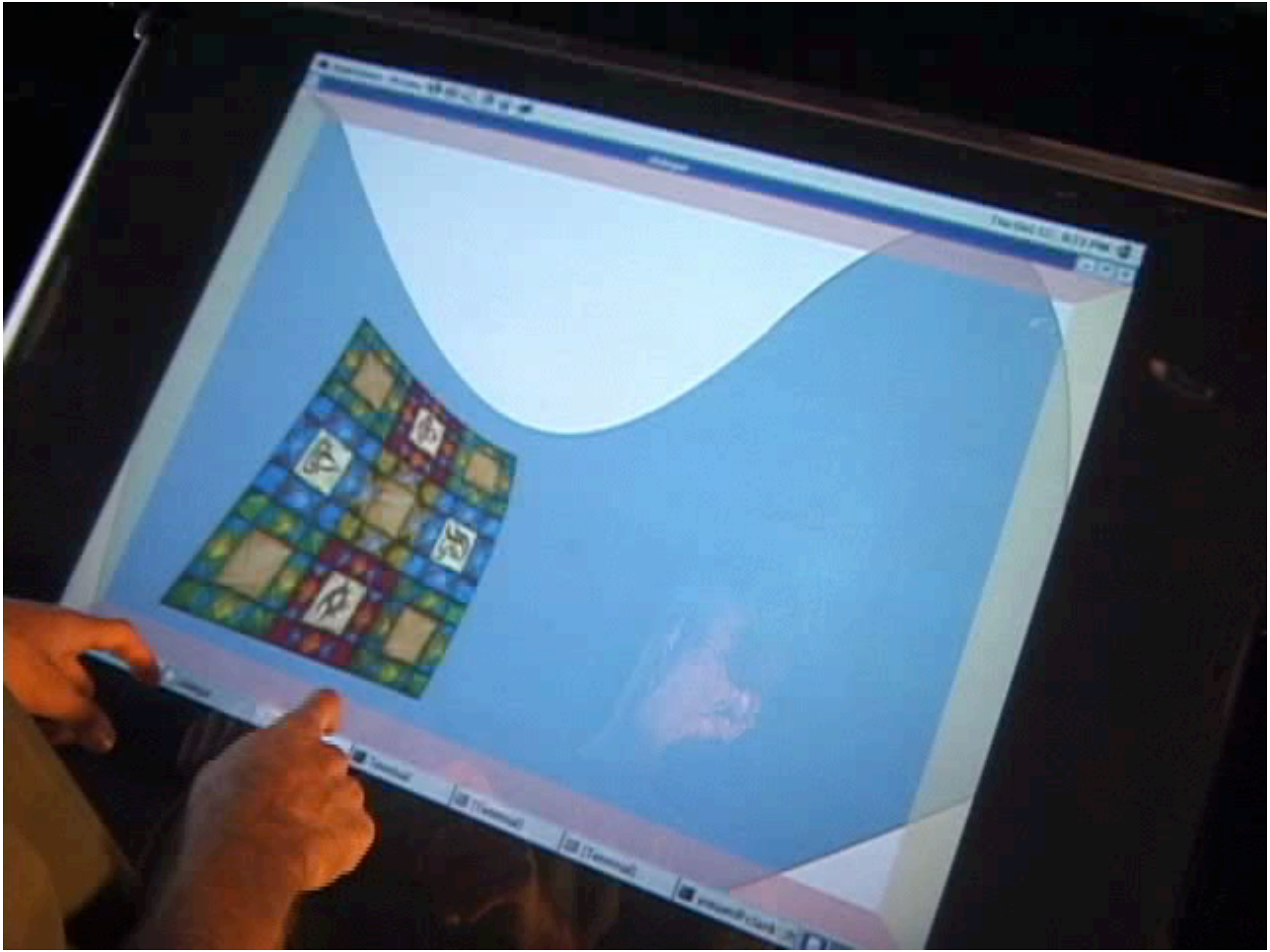
Multi-touch



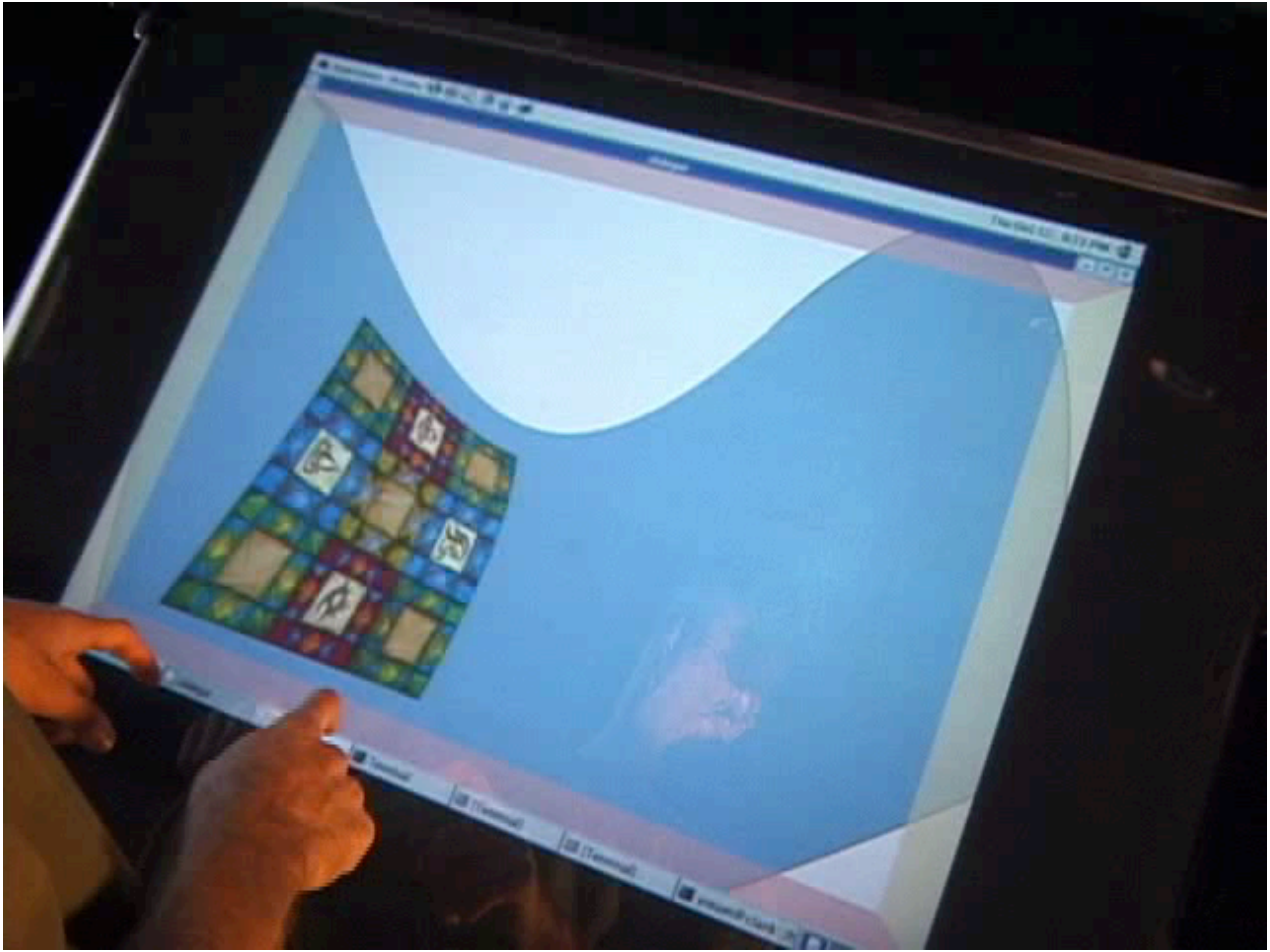
Multi-touch



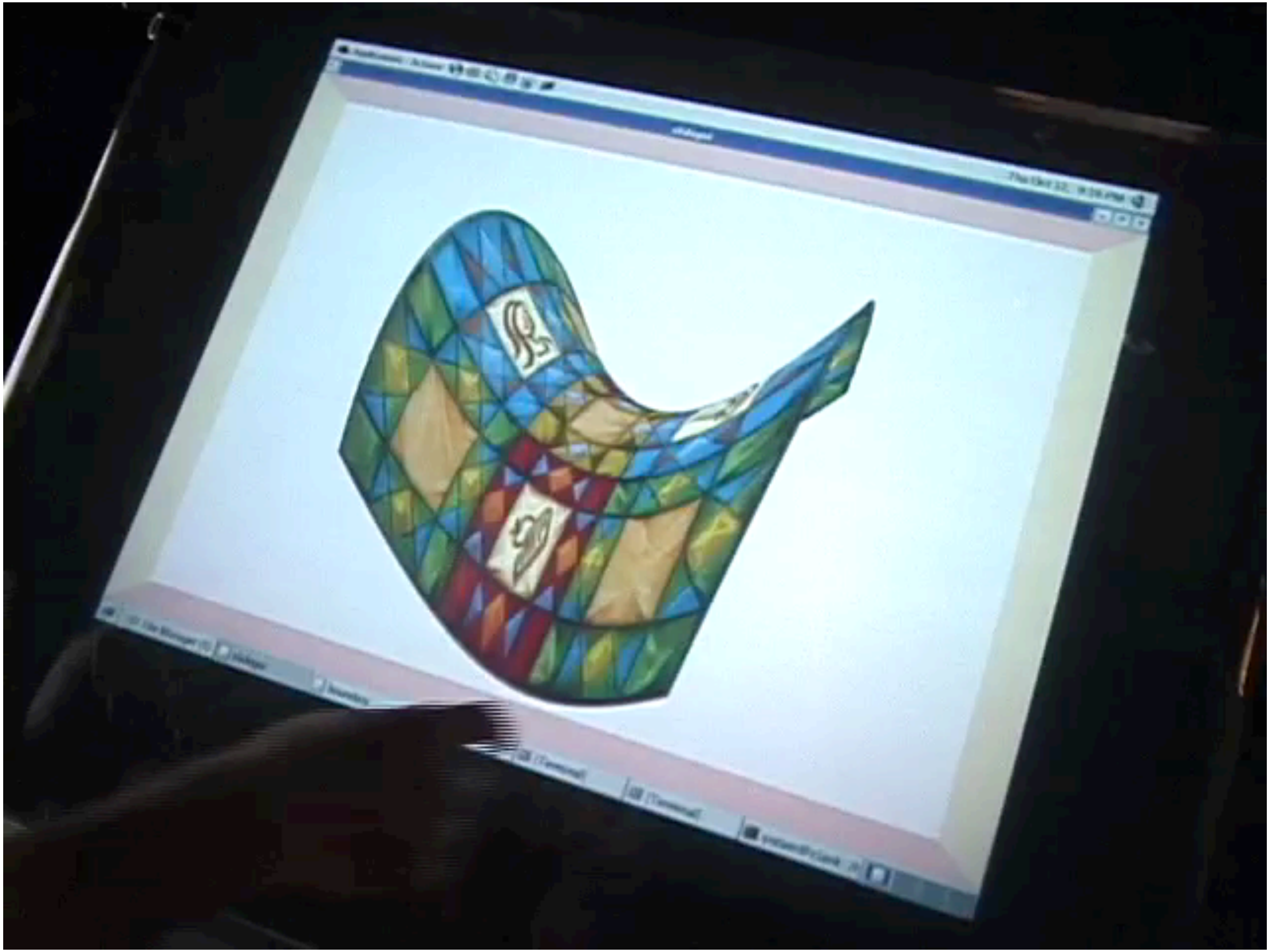
Texture Placement



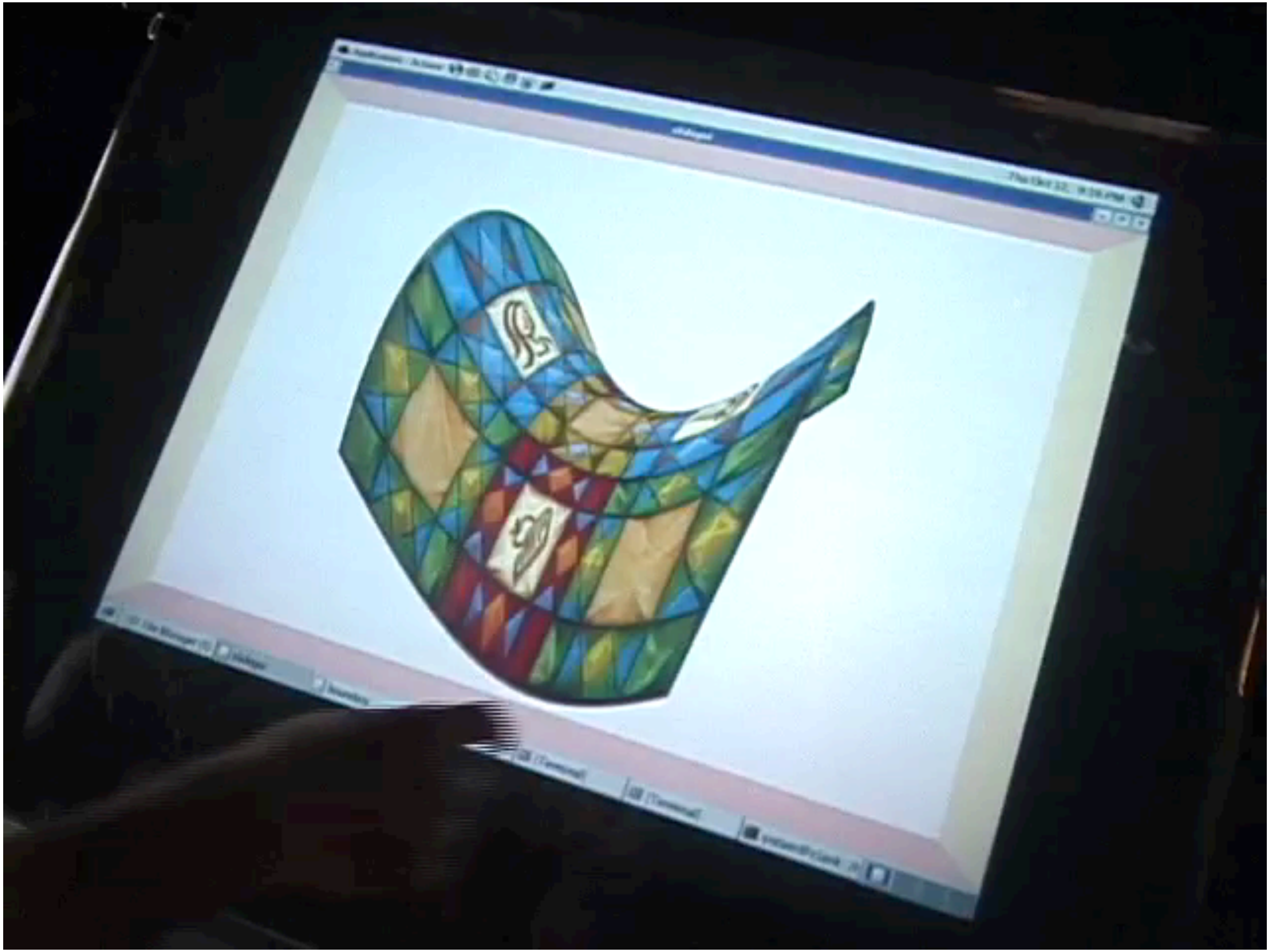
Texture Placement



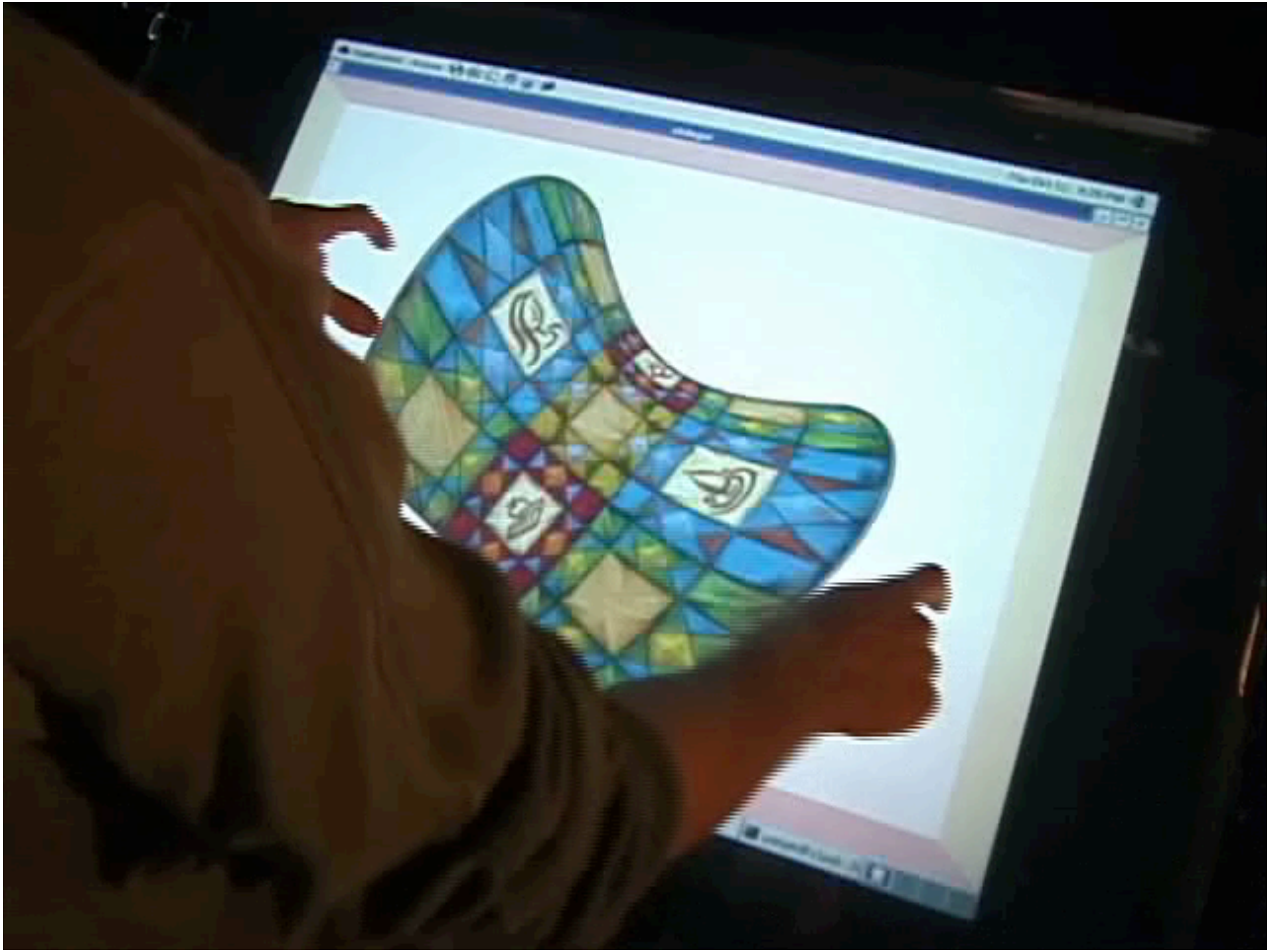
Feature Alignment



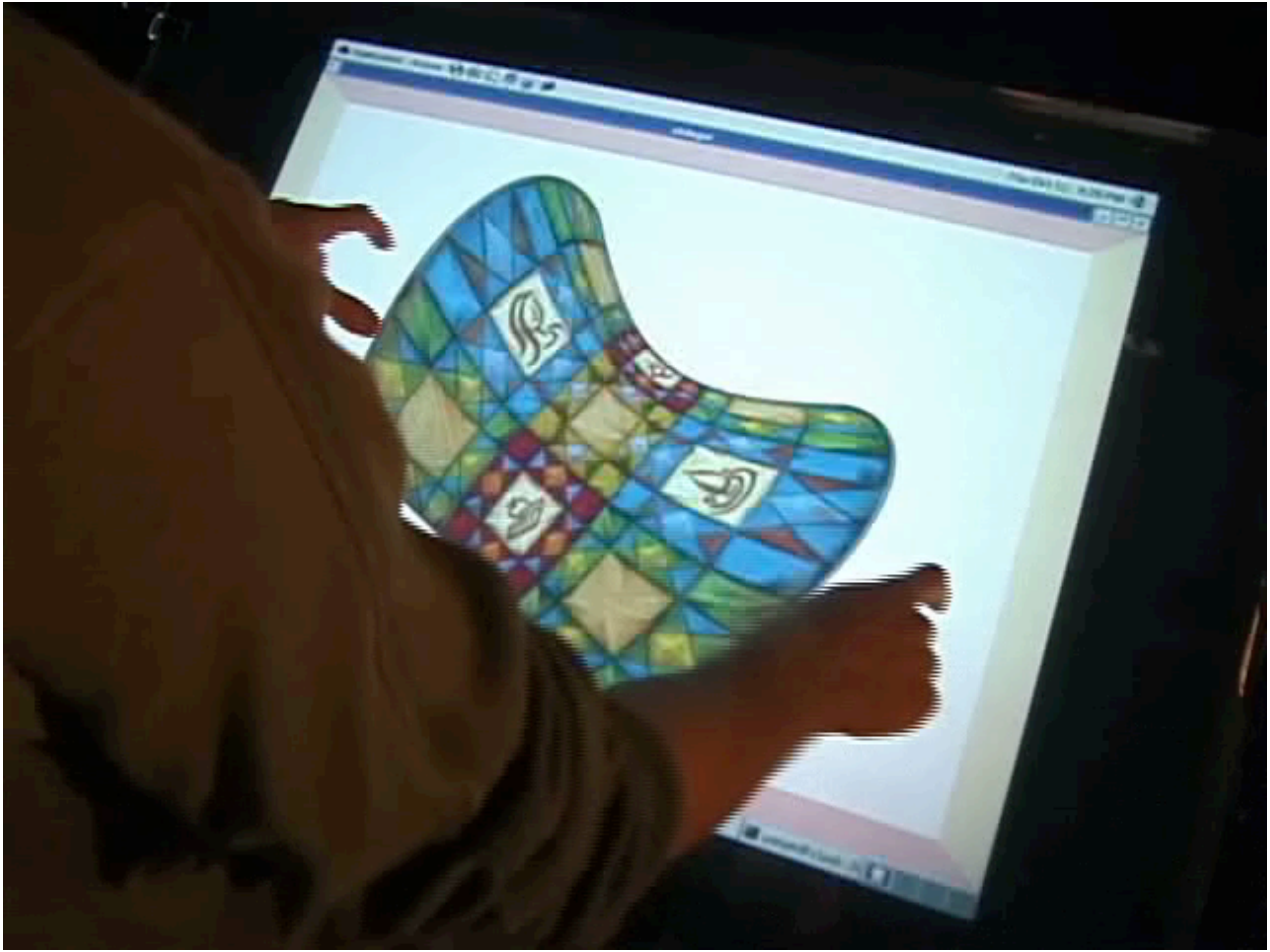
Feature Alignment



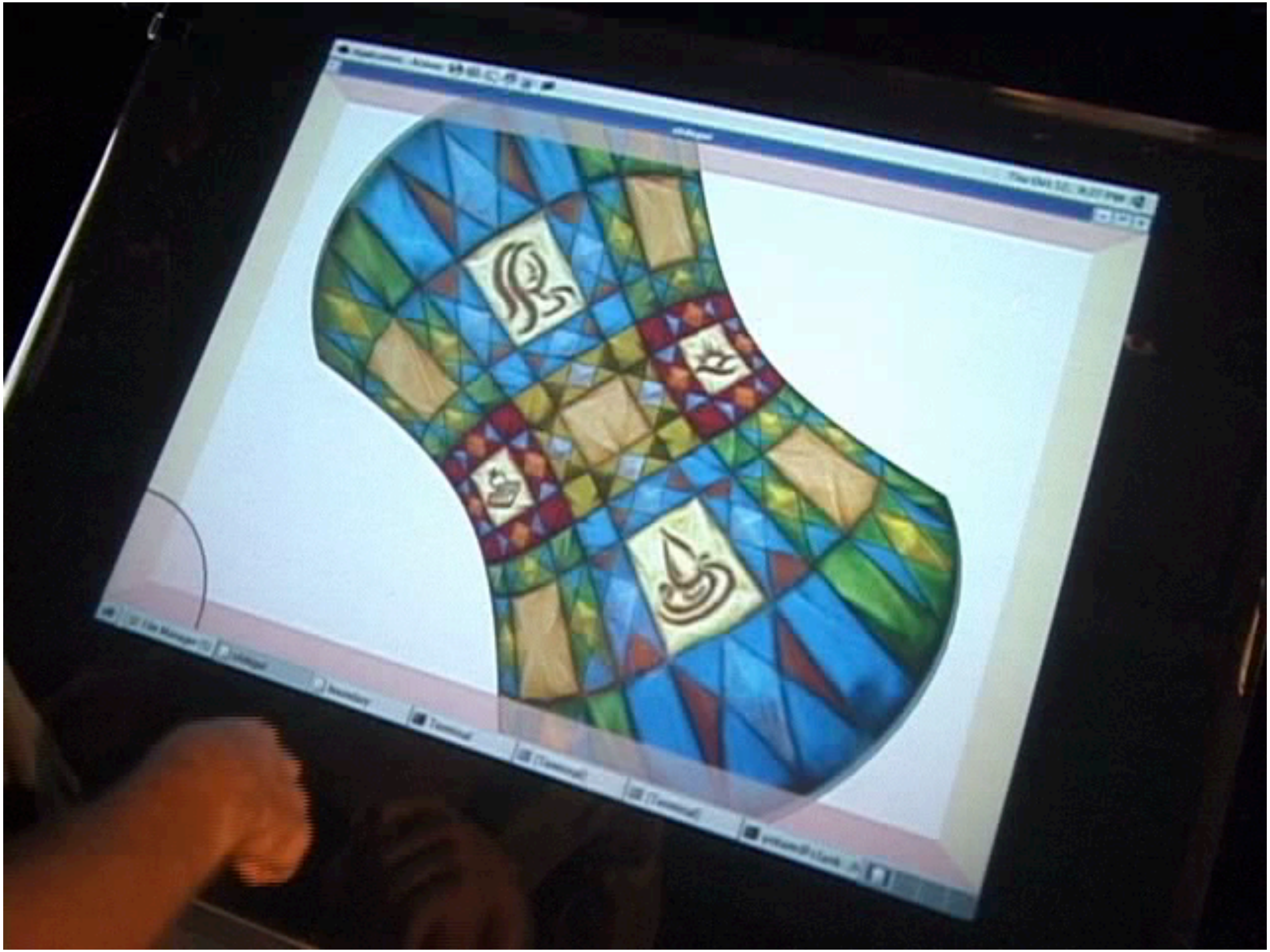
Pushpin Constraints



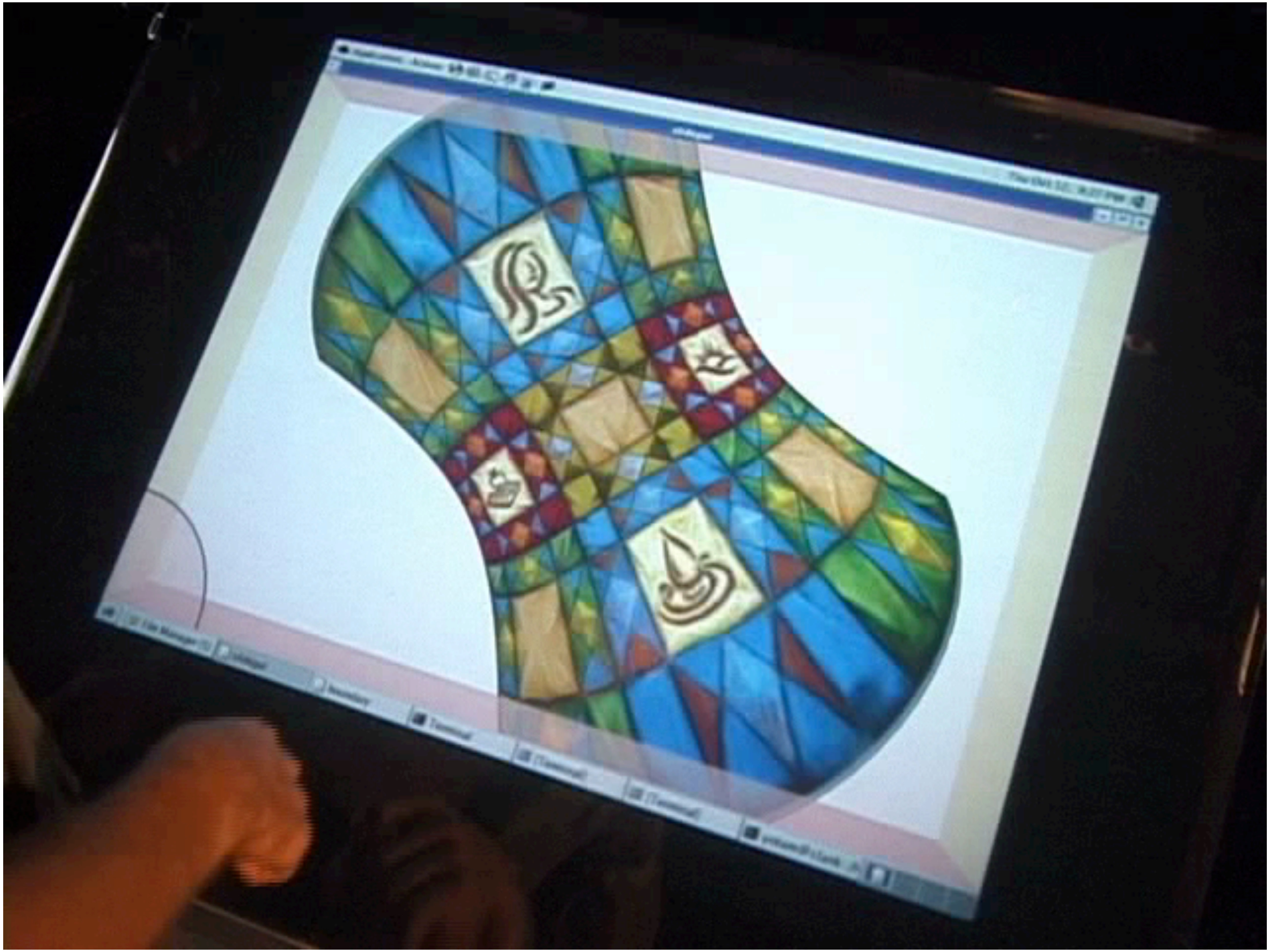
Pushpin Constraints



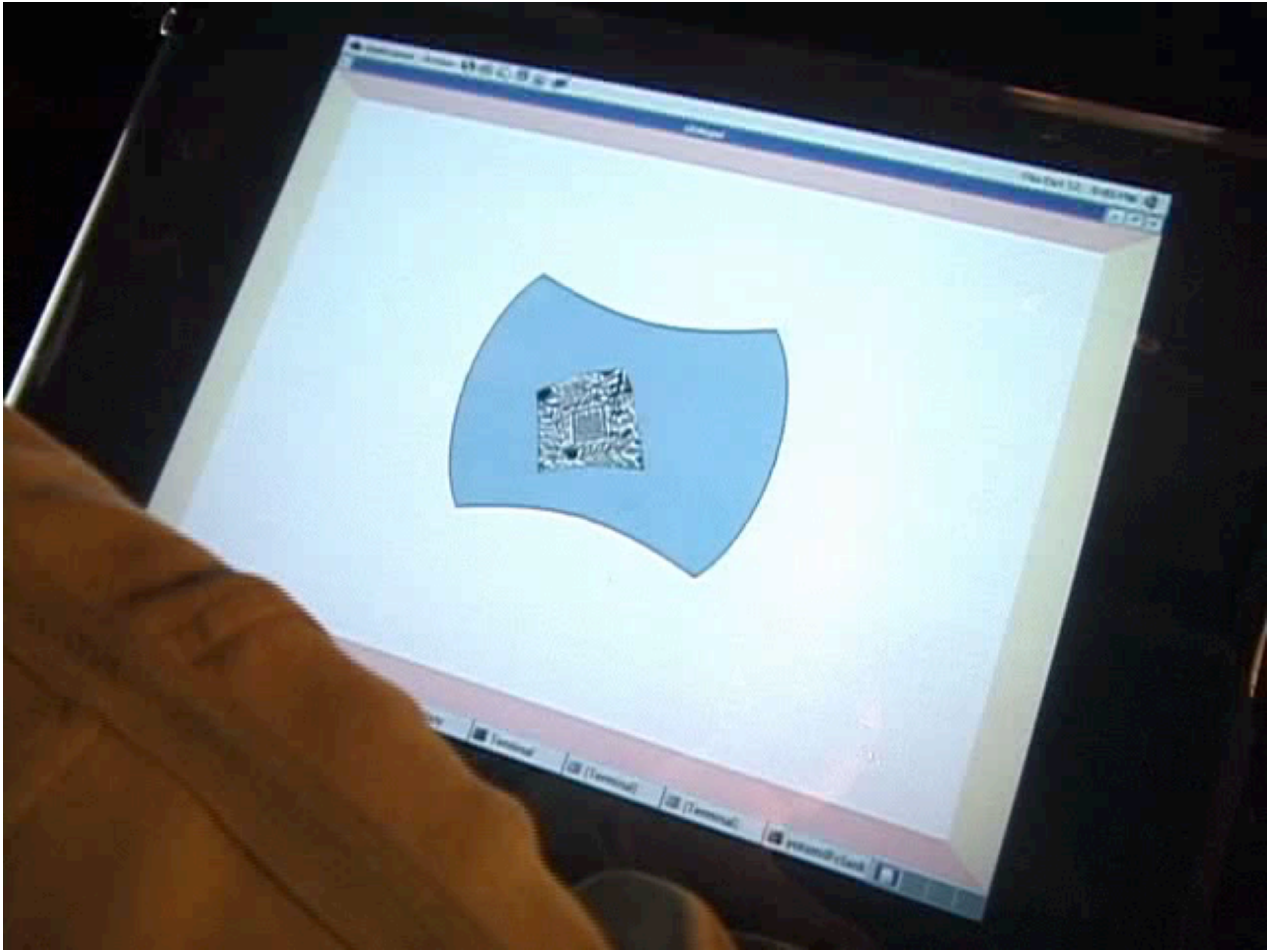
Local Deformations



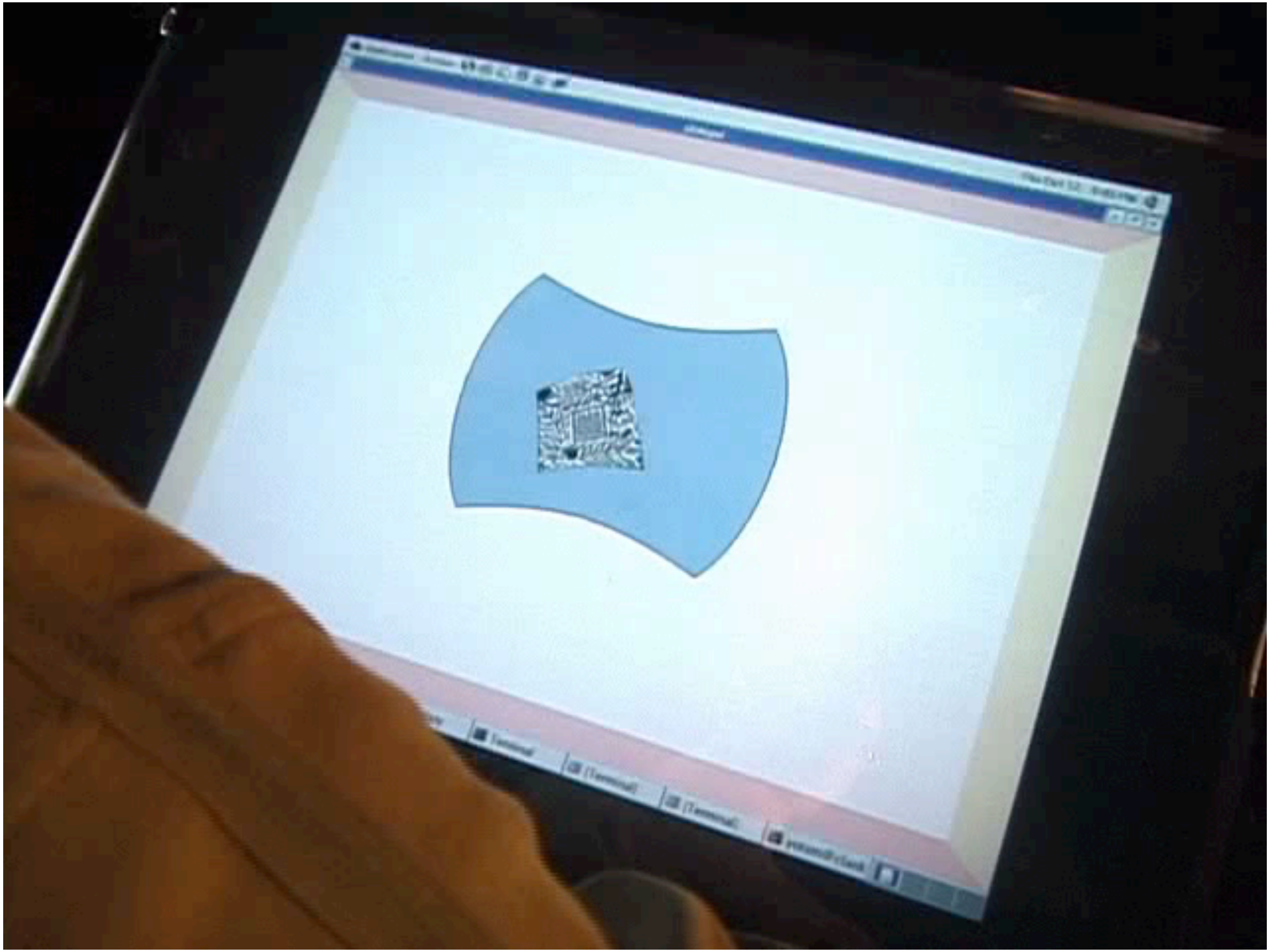
Local Deformations



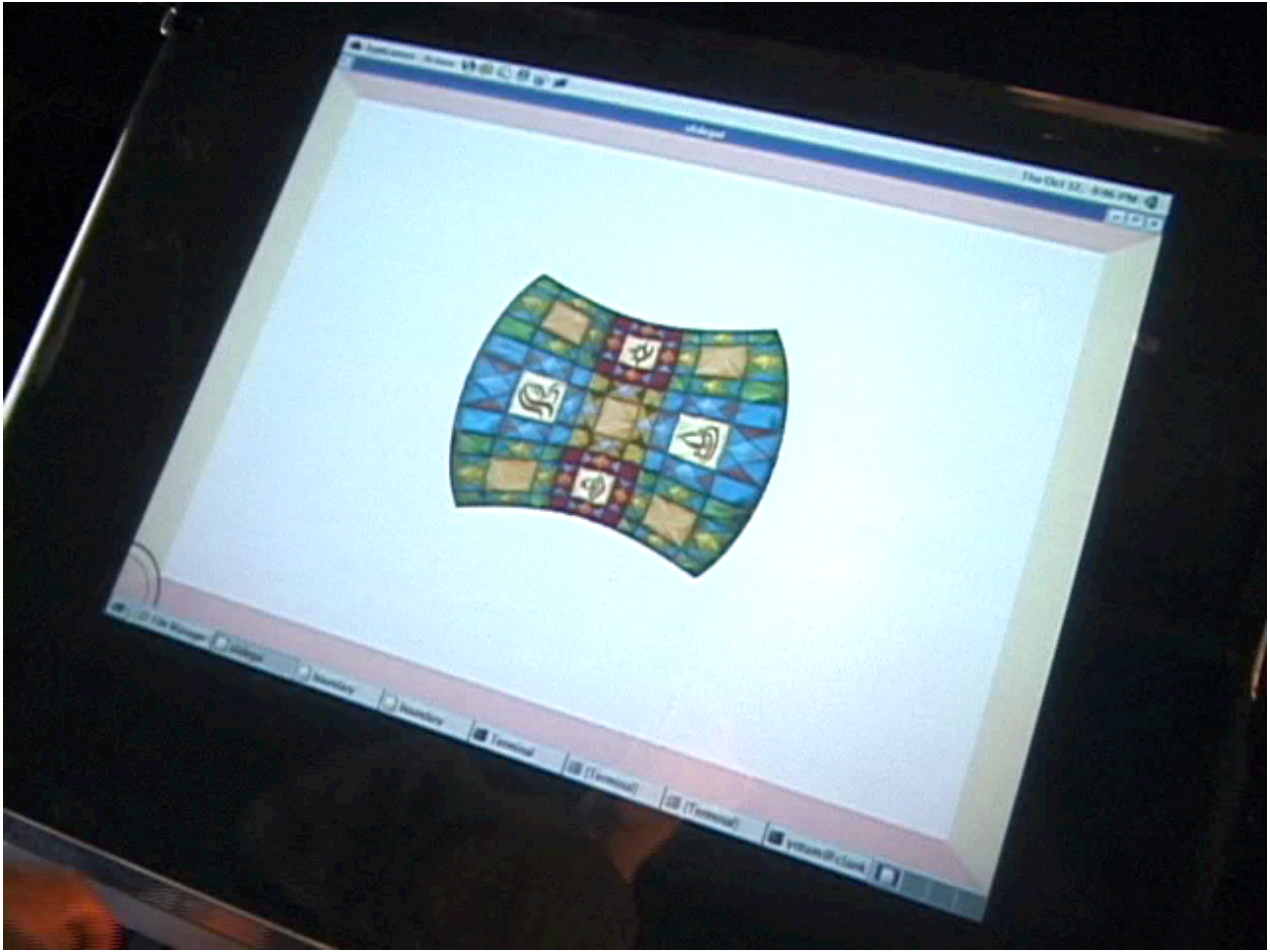
Texture Layers



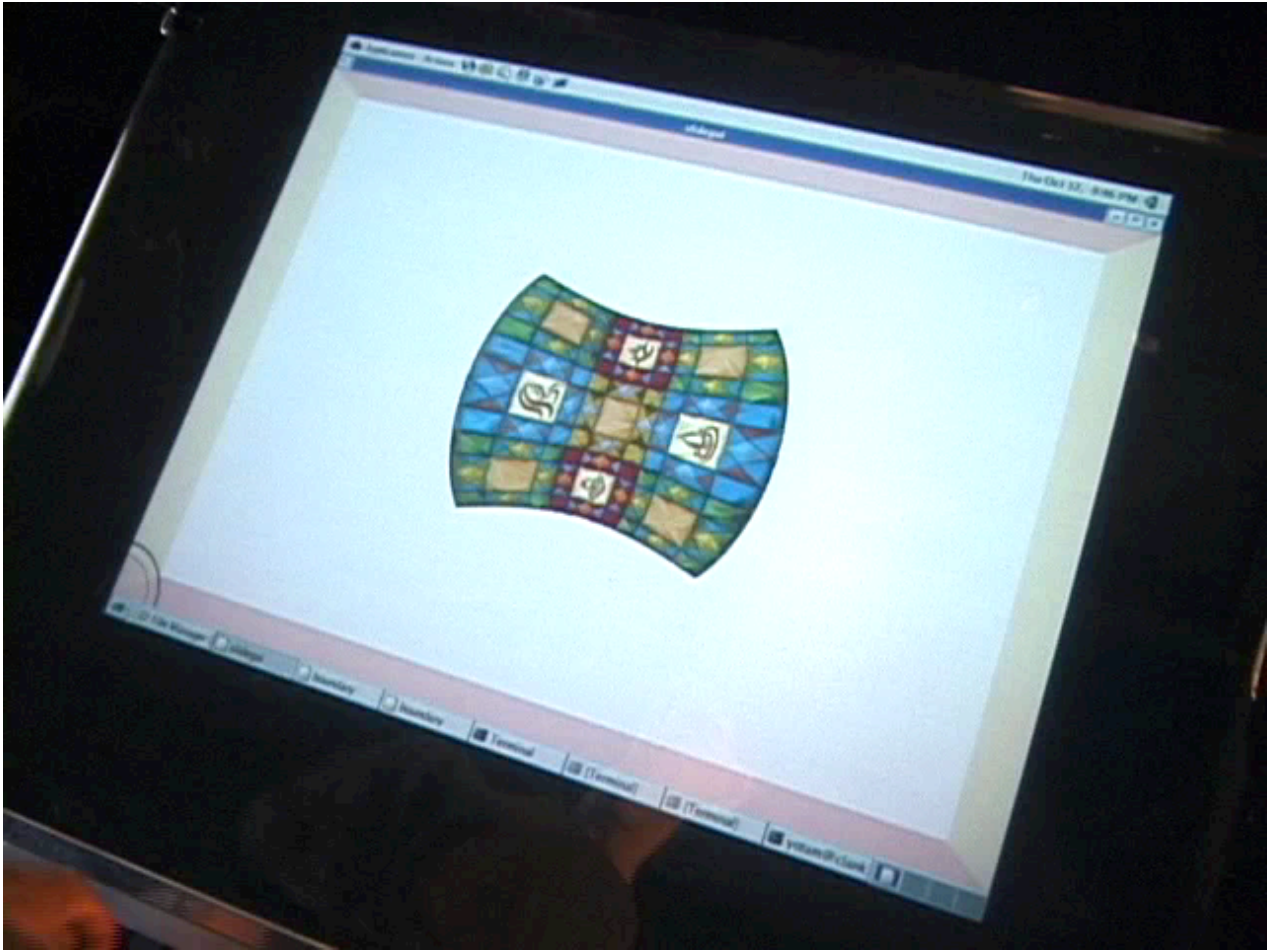
Texture Layers



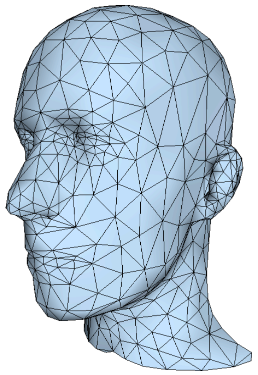
Alpha Airbrush

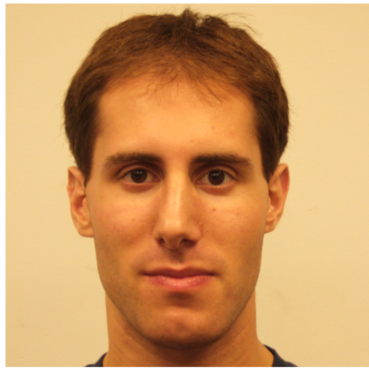


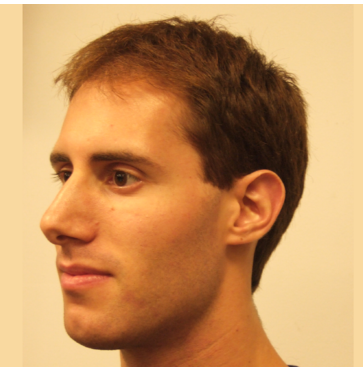
Alpha Airbrush

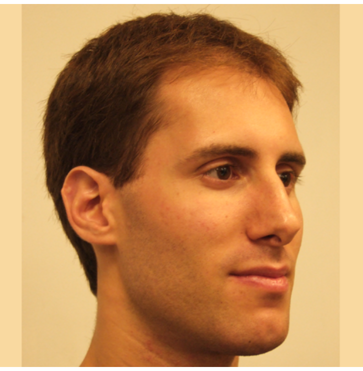


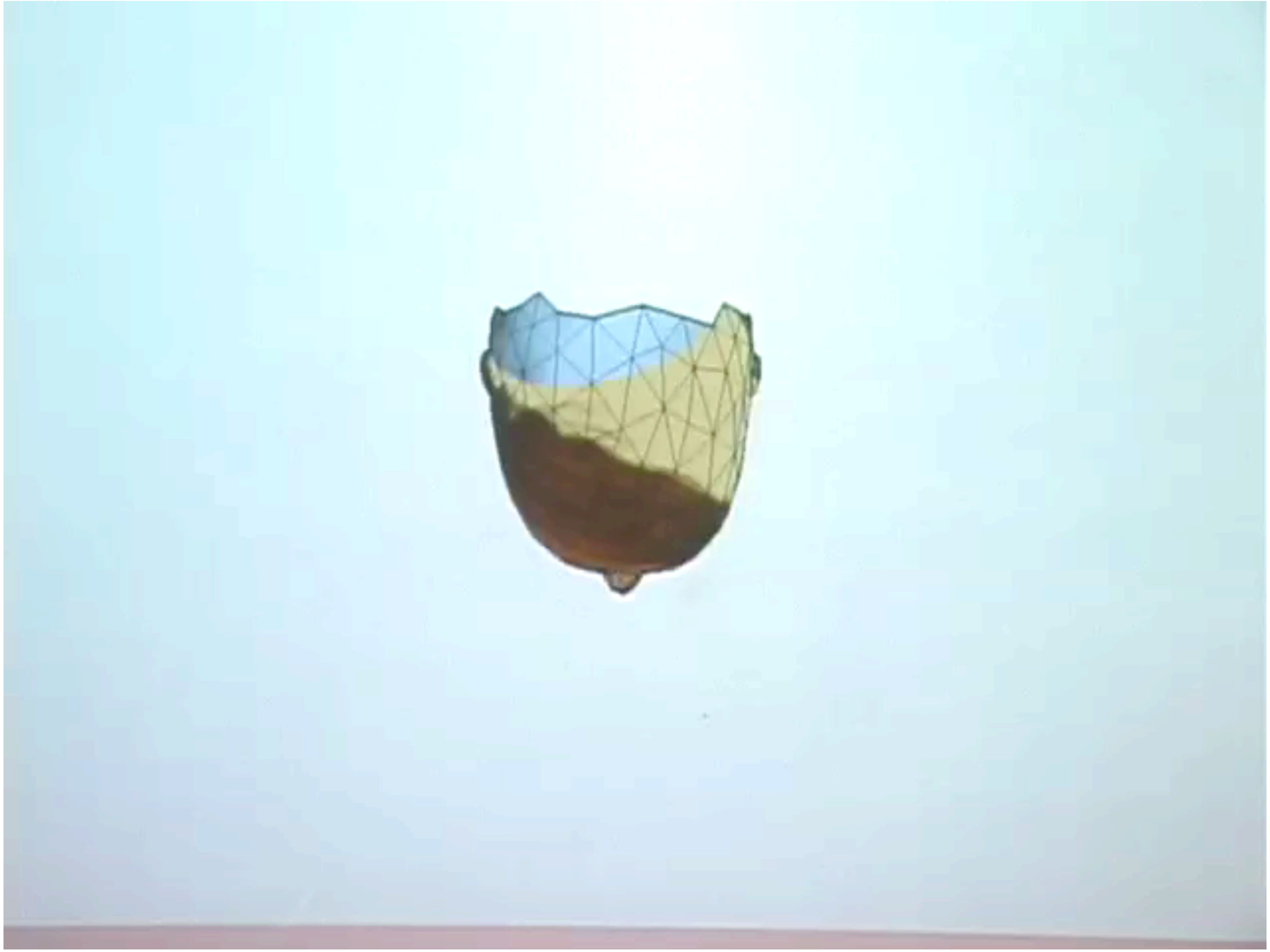
Results



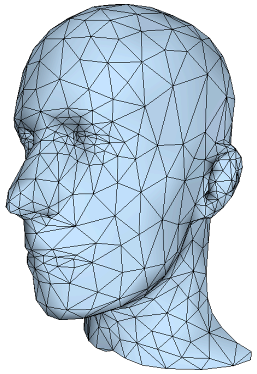


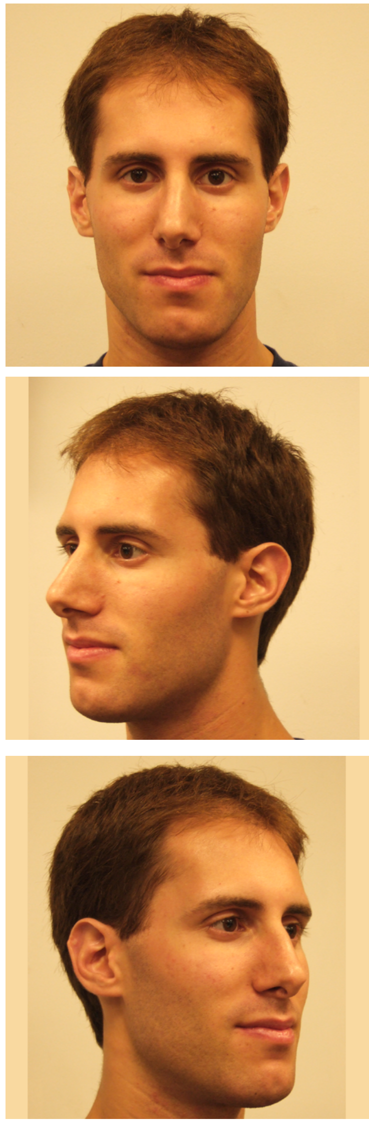


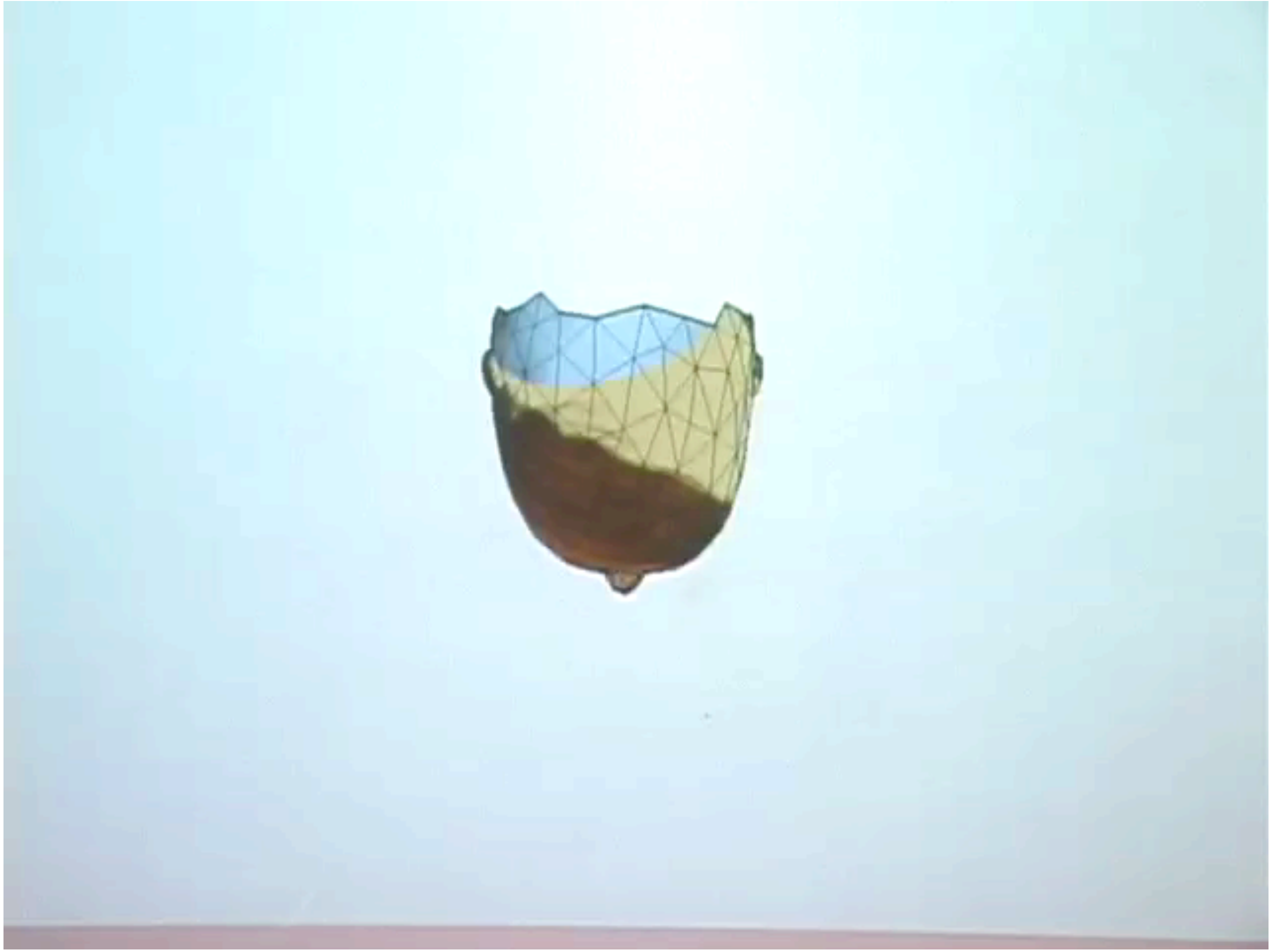




Results



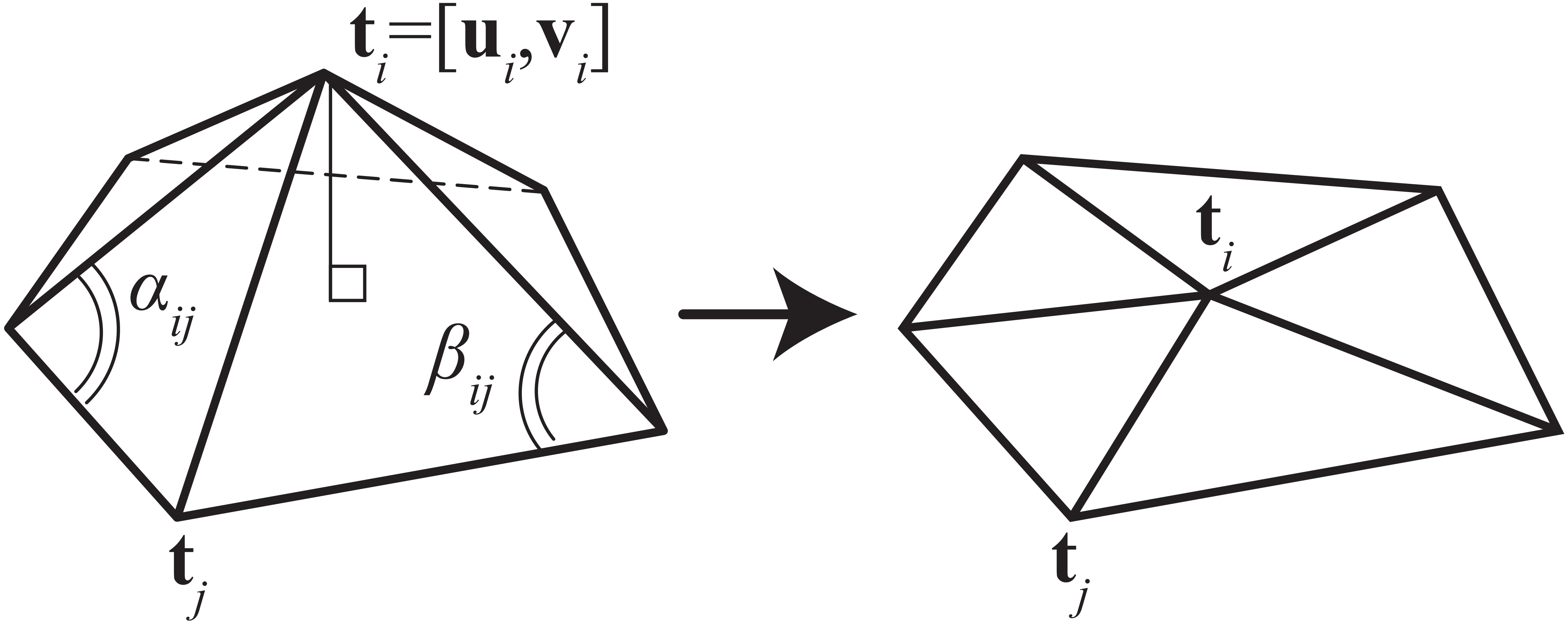




3 Formulae

Parameterization Algorithm

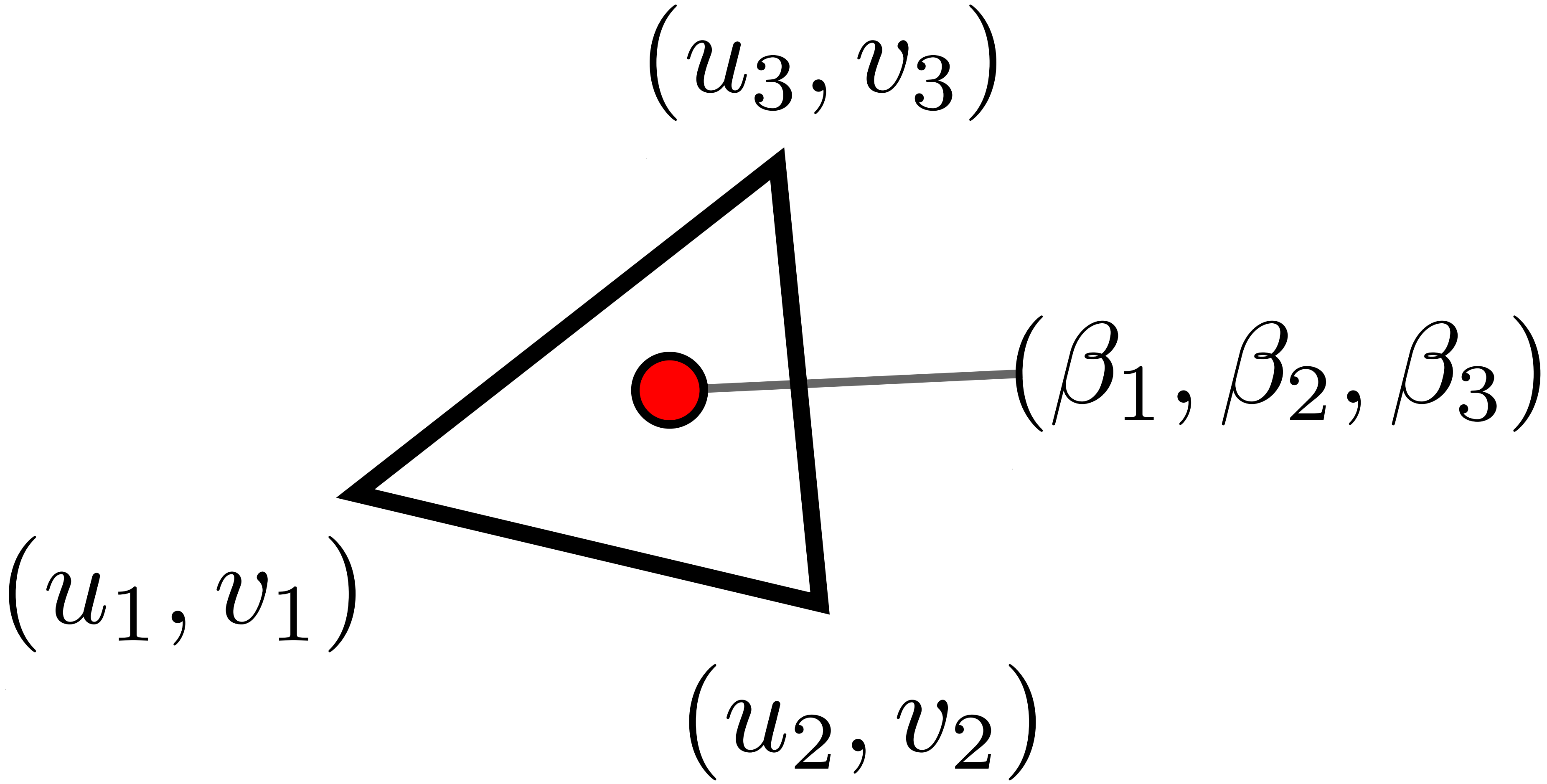
Linearized Bending Energy



$$t^T A t = E = \sum_i \frac{1}{8\,area_i}\left(\sum_{j\in N(i)}(\cot\alpha_{ij}+\cot\beta_{ij})(\mathbf{t}_i-\mathbf{t}_j)\right)^2$$

Constraints

Linear on triangles



$$\beta_1 u_1 + \beta_2 u_2 + \beta_3 u_3 = u_{fixed}$$
$$\beta_1 v_1 + \beta_2 v_2 + \beta_3 v_3 = v_{fixed}$$

Constraints

Modify system

bending energy Hessian

$$A^{ext} = \begin{pmatrix} A & C^T \\ C & 0 \end{pmatrix}$$

constraints $\left(\begin{array}{c} \beta_1 u_1 + \beta_2 u_2 + \beta_3 u_3 = u_{fixed} \\ \beta_1 v_1 + \beta_2 v_2 + \beta_3 v_3 = v_{fixed} \end{array} \right)$

Need a scheme for quickly updating inverse

Game development is many things

programming languages

software engineering

networking and security

operating systems

game design (game theory, HCI, psychology, sociology)

artwork

story

artificial intelligence

physics

animation

input (computer vision)

computer graphics (rendering)

authoring

Thank You

Questions?