Latest Research in Computer Graphics and Graphics in Games

ISTD 50.017
Sai-Kit Yeung

Final Project

- Presentation
  - Next Wednesday 8:40-10:30am
  - 15 mins: 11-13 mins + 2-4 mins Q&A
  - Criteria: Organization, Style & Pace, Content: depth and accuracy, Use of Visual aids
  - Tips for giving clear talk:
    - http://www.cs.cmu.edu/~kayvonf/misc/cleartalktips.pdf

Final Project

- Report
  - Due on 4th May 11:55pm
  - Hand in your code, data, etc
  - Refine your proposal with
    - Methodology
    - Actual implementation
    - Results
    - Explain any change you have made

Game Development Outline

- Game Development
  - Typical Process
- What's in a game?
  - Game Simulation
  - Numeric Computation
  - Shading

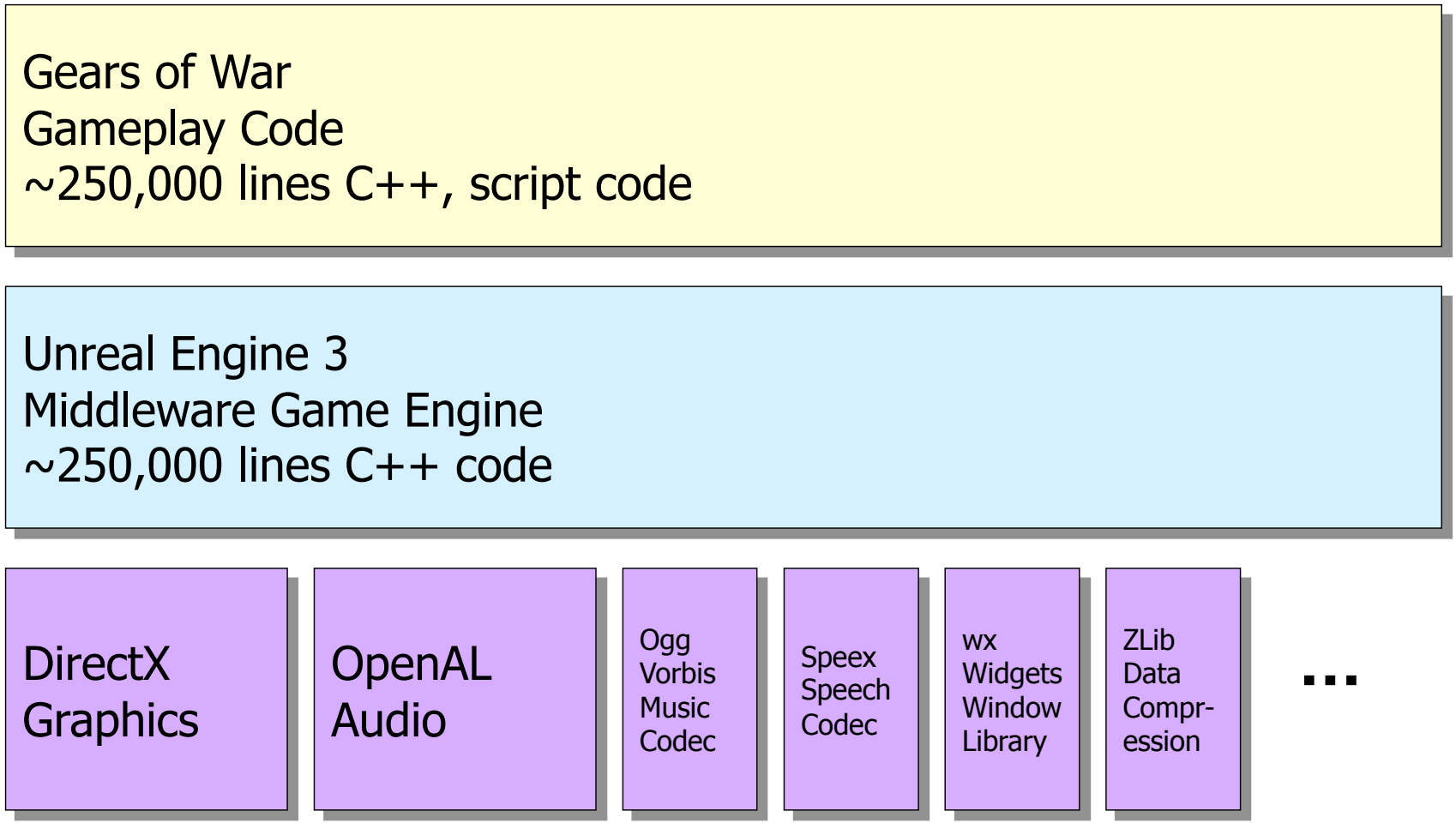
http://www.cs.princeton.edu/~dpw/popl/06/Tim-POPL.ppt

Tim Sweeney

Game Development: Gears of War

- Resources
  - ~10 programmers
  - ~20 artists
  - ~24 month development cycle
  - ~$10M budget
- Software Dependencies
  - 1 middleware game engine
  - ~20 middleware libraries
  - OS graphics APIs, sound, input, etc

Software Dependencies



Game Development – Platforms

- The typical Unreal Engine 3/4 game will ship on:
  - Xbox 360/One
  - PlayStation 3/4
  - Windows
- Some will also ship on:
  - Linux
  - MacOS

What is in a game?

The obvious:

- Rendering
- Pixel shading
- Physics simulation, collision detection
- Game world simulation
- Artificial intelligence, path finding

But it’s not just fun and games:

- Data persistence with versioning, streaming
- Distributed Computing (multiplayer game simulation)
- Visual content authoring tools
- Scripting and compiler technology
- User interfaces

Three Kinds of Code

- Gameplay Simulation
- Numeric Computation
- Shading

Gameplay Simulation

- Models the state of the game world as interacting objects evolve over time
- High-level, object-oriented code
- Written in C++ or scripting language
- Usually garbage-collected

Gameplay Simulation – The Numbers

- 30-60 updates (frames) per second
  - VR needs higher!
- ~1000 distinct gameplay classes
  - Contain member functions
  - Highly dynamic
- ~10,000 active gameplay objects
- Each time a gameplay object is updated, it typically touches 5-10 other objects

Numeric Computation

- Algorithms:
  - Scene graph traversal
  - Physics simulation
  - Collision Detection
  - Path Finding
  - Sound Propagation
- Low-level, high-performance code
- Written in C++ with Single instruction, multiple data (SIMD)
- Essentially functional
  - Transforms a small input data set to a small output data set, making use of large constant data structures.

Shading

- Generates pixel and vertex attributes
- Written in High Level Shader Language (HLSL)/CG shading language
- Runs on the GPU
- Inherently data-parallel
  - Control flow is statically known
  - “Embarrassingly Parallel”

Shading – The Numbers

- Game runs at 30 FPS @ 1280x720p
- ~5,000 visible objects
- ~10M pixels rendered per frame
  - Per-pixel lighting and shadowing requires multiple rendering passes per object and per-light
- Typical pixel shader is ~100 instructions long
- ~500 giga FLoating-point Operations Per Second (GFLOPS) compute power

Three Kinds of Code

	Game Simulation	Numeric Computation	Shading
Languages	C++, Scripting	C++	CG, HLSL
CPU Budget	10%	90%	n/a
Lines of Code	250,000	250,000	10,000
GPU Usage	0.5 GFLOPS	5 GFLOPS	500 GFLOPS

Latest Research in Computer Graphics

- Modeling
- Physics based Animation
- Images/Computational Photography
- Fabrication
- Fast Rendering
- Hair, Face, Fluid…
- Non-Photorealistic Rendering
- Interactive tools
- Architecture

Latest Research in Computer Graphics

- How Do Humans Sketch Objects?
- SIGGRAPH 2012
- Related to Quick, Draw! released by Google recently

How Do Humans Sketch Objects?

Mathias Eitz, James Hays and Marc Alexa

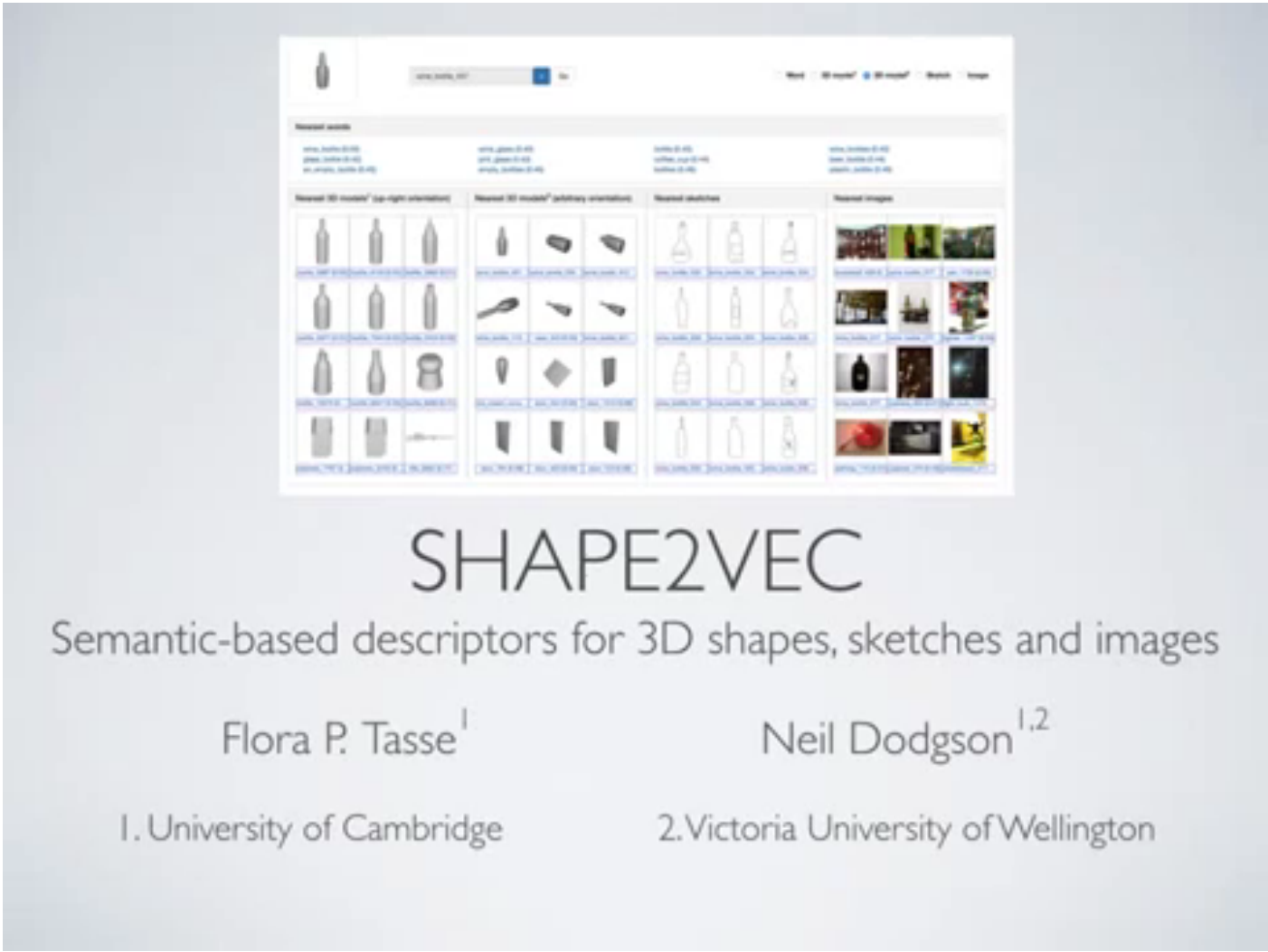
SIGGRAPH 2012

http://cybertron.cg.tu-berlin.de/eitz/projects/classifysketch/

Latest Research in Computer Graphics

- Shape2Vec: semantic-based descriptors for 3D shapes, sketches and images
- SIGGRAPHAsia 2016

Latest Research in Computer Graphics

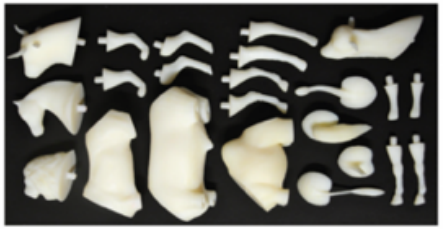


https://www.youtube.com/watch?v=oVR4af9UWio

Latest Research in Computer Graphics: Modeling

- Interchangeable Components for Hands-On Assembly Based Modeling
- SIGGRAPHAsia 2016

Compatibly Segmented Shapes

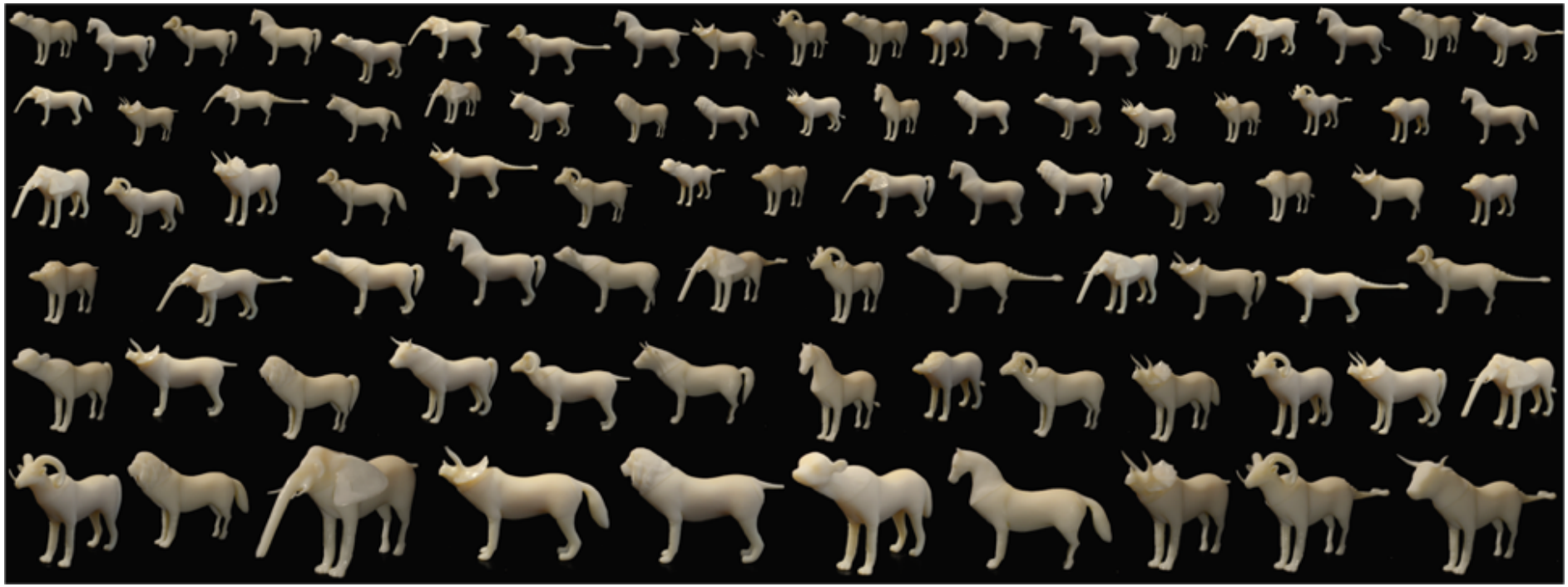


Interchangeable Components

Assembled Objects

Latest Research in Computer Graphics: Modeling

- Interchangeable Components for Hands-On Assembly Based Modeling



Latest Research in Computer Graphics: Modeling

http://people.sutd.edu.sg/~saikit/projects/handson/index.html

Latest Research in Computer Graphics: Animation

- Surface-Only Liquids
- SIGGRAPH 2016

Latest Research in Computer Graphics: Animation

http://www.cs.columbia.edu/cg/surfaceliquids/

Latest Research in Computer Graphics: Animation

- Projective dynamics: fusing constraint projections for fast simulation
- SIGGRAPH 2014

Latest Research in Computer Graphics: Animation

Projective Dynamics

Fusing Constraint Projections for Fast Simulation

Sofien Bouaziz — *EPFL*

Sebastian Martin — *VM Research*

Tiantian Liu — *University of Pennsylvania*

Ladislav Kavan — *University of Pennsylvania*

Mark Pauly — *EPFL*

https://www.cs.utah.edu/~ladislav/bouaziz14projective/bouaziz14projective.html

Latest Research in Computer Graphics: Rendering

- Emptying, Refurnishing, and Relighting Indoor Spaces
- SIGGRAPHAsia 2016

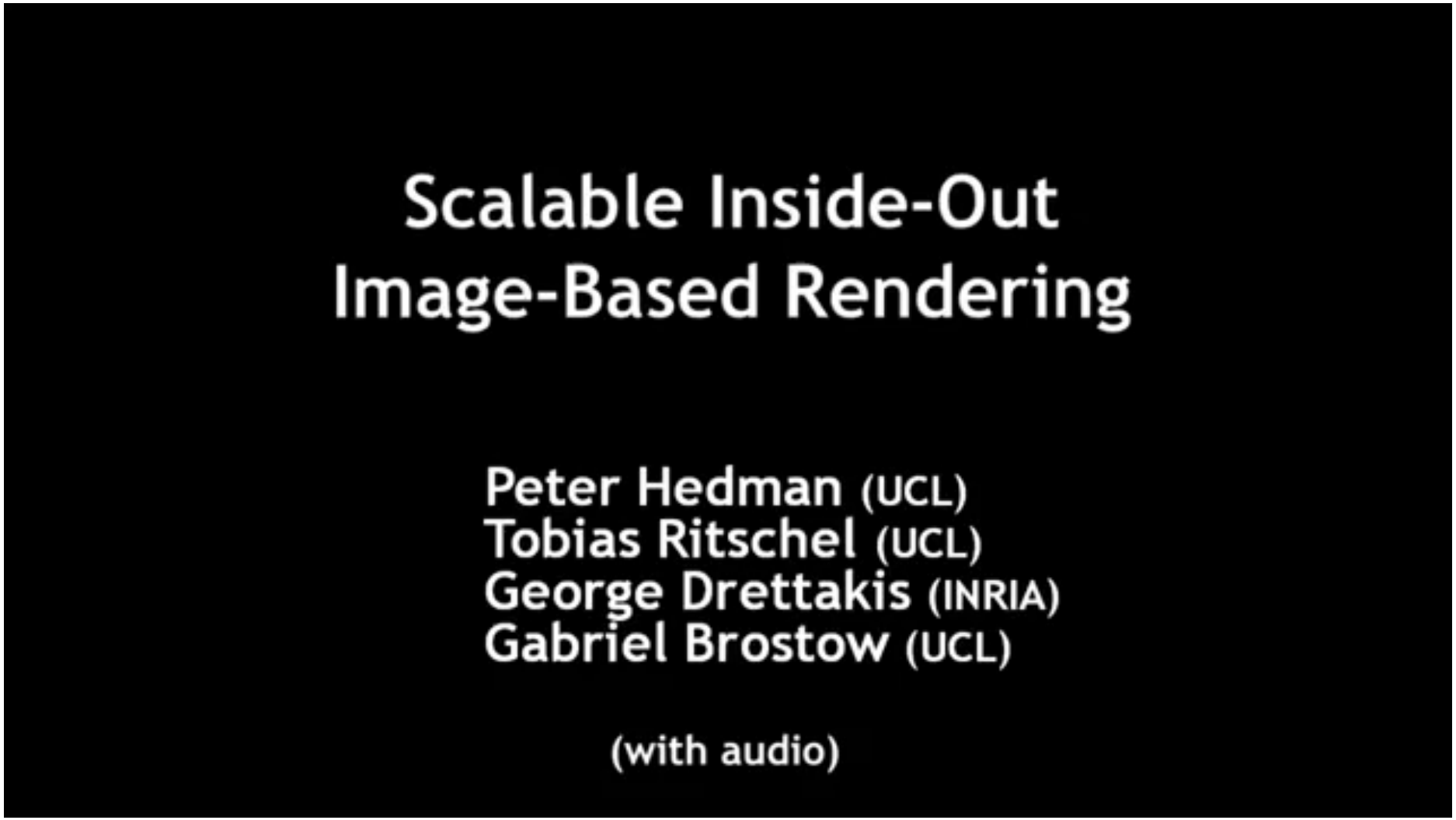
Latest Research in Computer Graphics: Rendering

http://grail.cs.washington.edu/projects/emptying/

Latest Research in Computer Graphics: Rendering

- Scalable Inside-Out Image-Based Rendering
- SIGGRAPHAsia 2016

Latest Research in Computer Graphics: Rendering



http://visual.cs.ucl.ac.uk/pubs/insideout/

Latest Research in Computer Graphics: Motion Capture

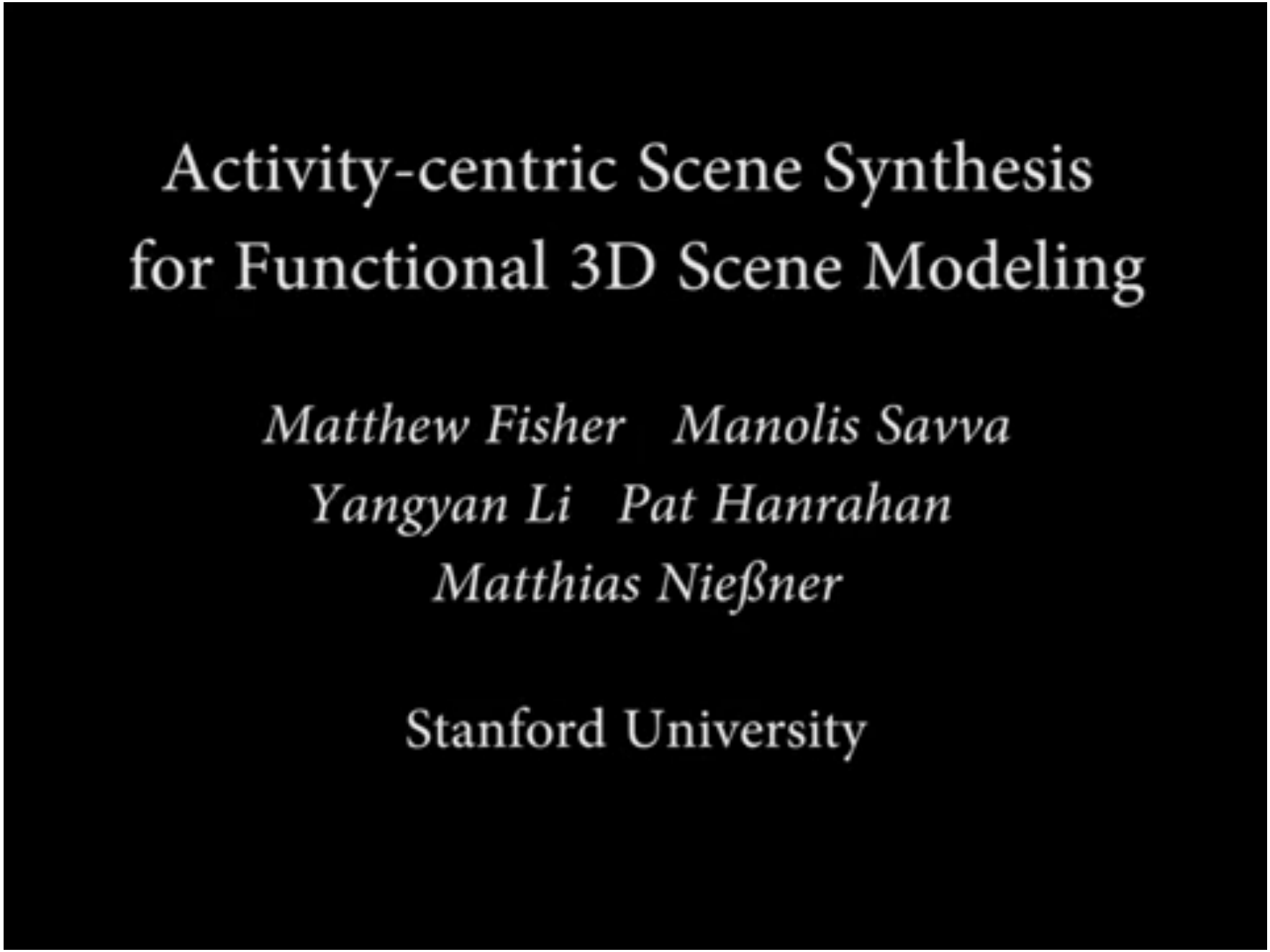
- MoSh: Motion and Shape Capture from Sparse Markers
- SIGGRAPHAsia 2014

Latest Research in Computer Graphics: Motion Capture

https://ps.is.tuebingen.mpg.de/research_projects/mosh

Latest Research in Computer Graphics: Scene Modeling

- Activity-centric Scene Synthesis for Functional 3D Scene Modeling
- SIGGRAPHAsia 2015



http://graphics.stanford.edu/projects/actsynth/

What have you learned?

- You did not only learn how to “use” Photoshop, Illustrator, Maya or Unity
- You learnt how to “make” Photoshop, Illustrator, Maya or Unity

What have you learned?

That’s all for the course!

- But definitely much more in the CG world!
- Subject Evaluation
  - Please, provide feedback about 50.017