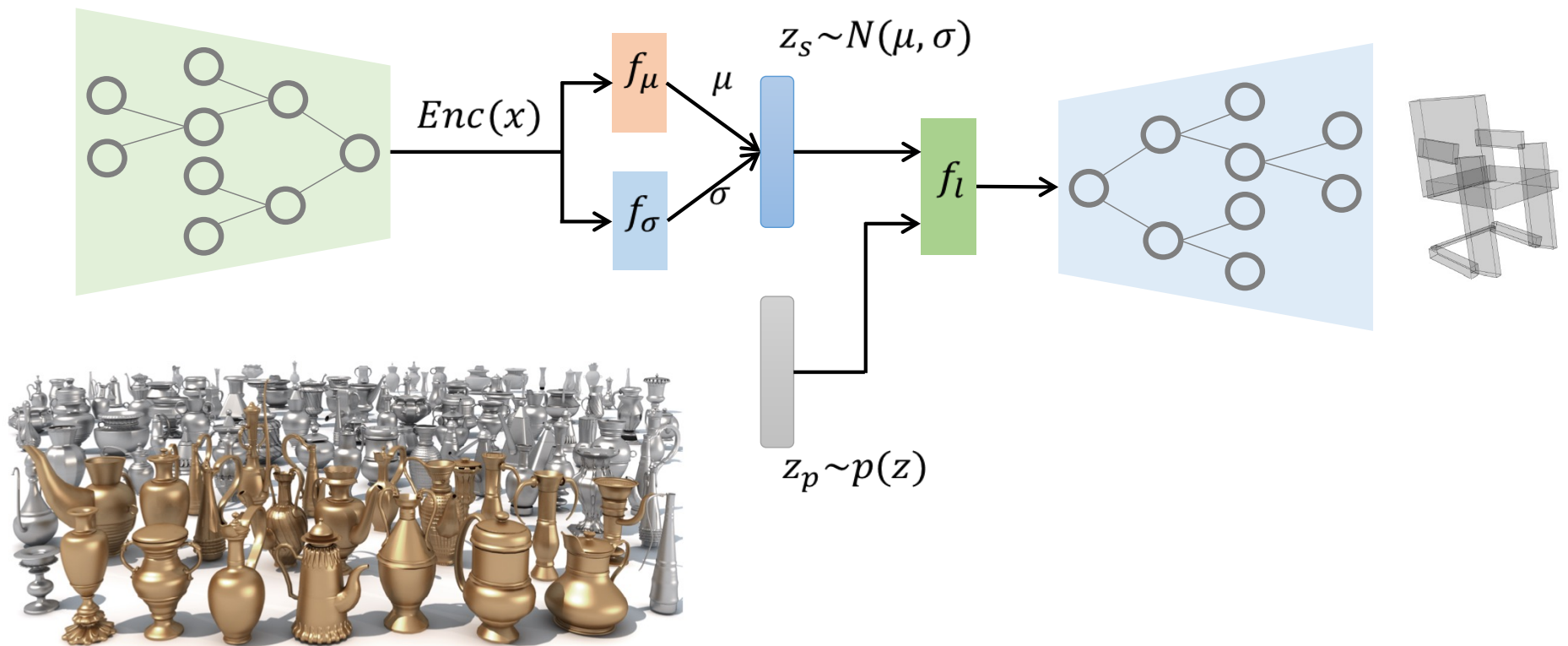
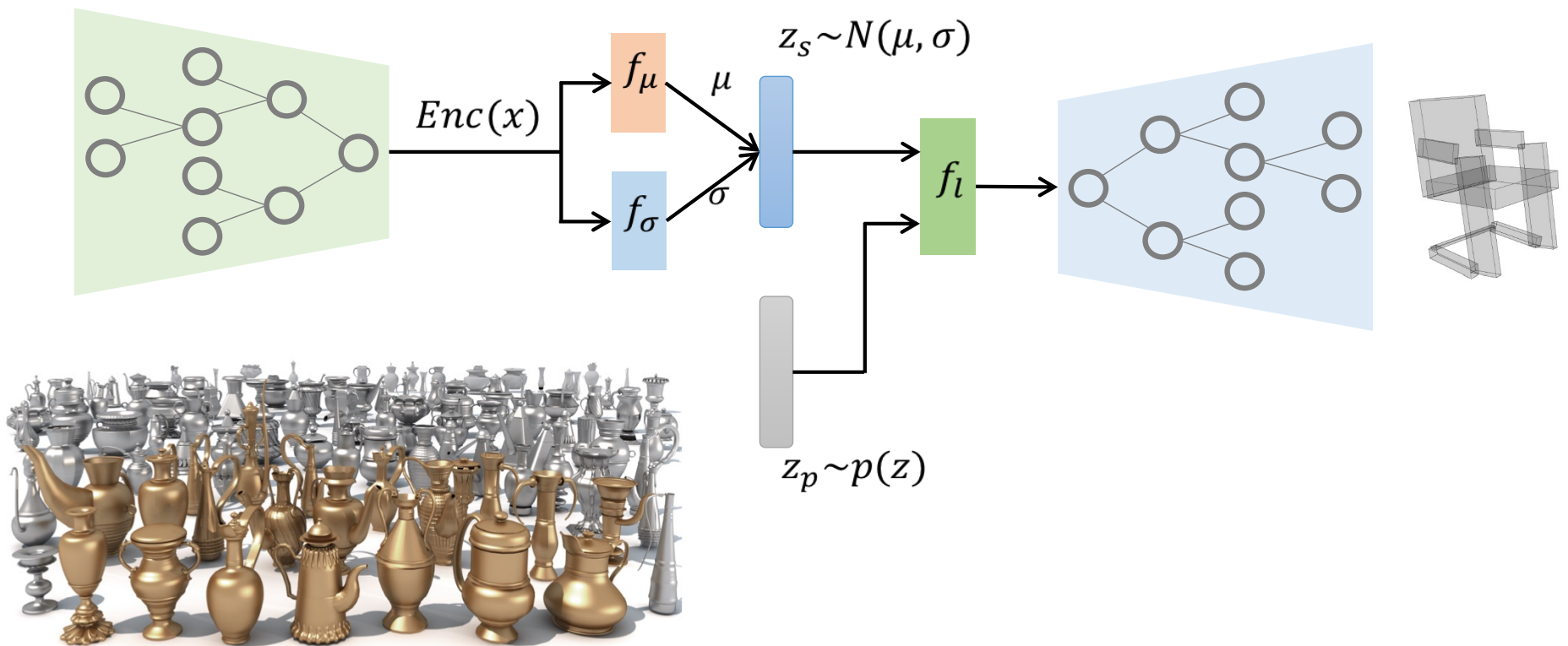


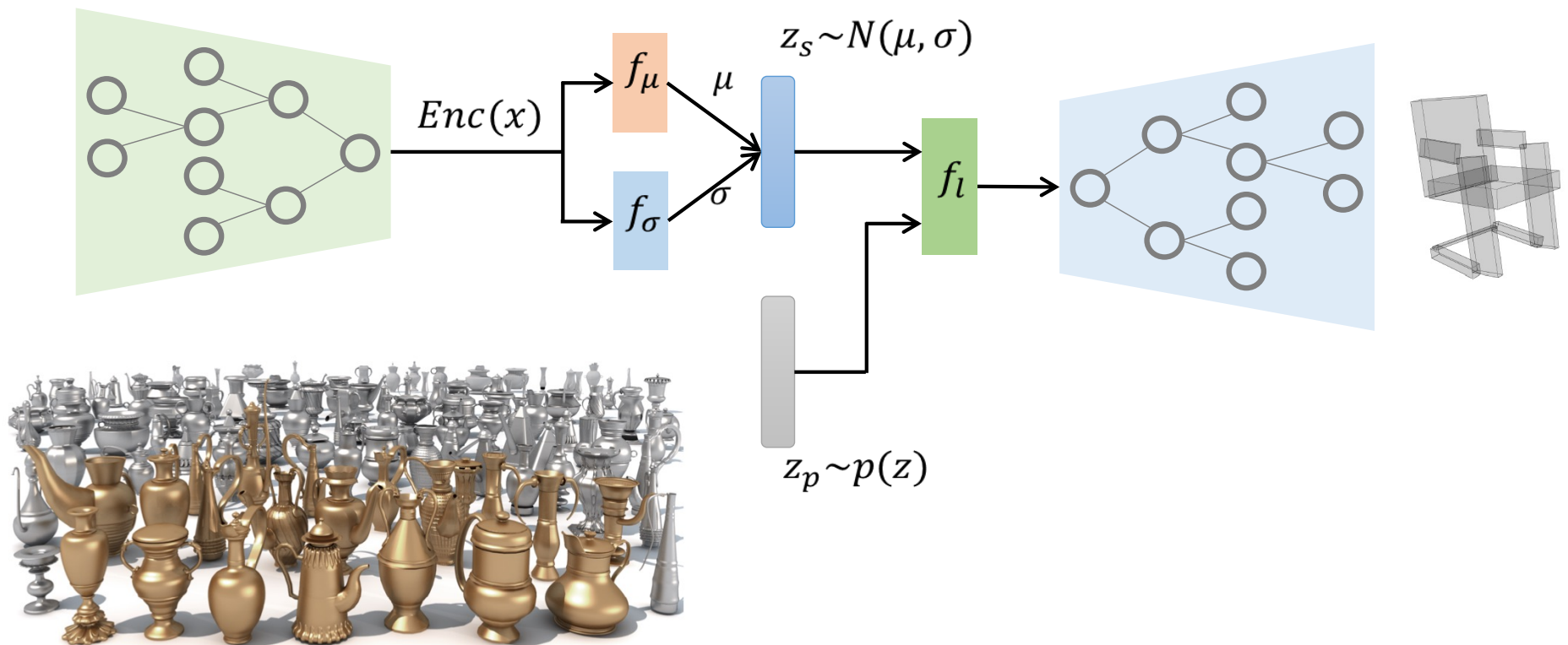
Computer Graphics and Geometric Modeling in the AI Era



Why is geometric modeling such an important topic in graphics?



What is computer graphics – then and now – the BIG picture



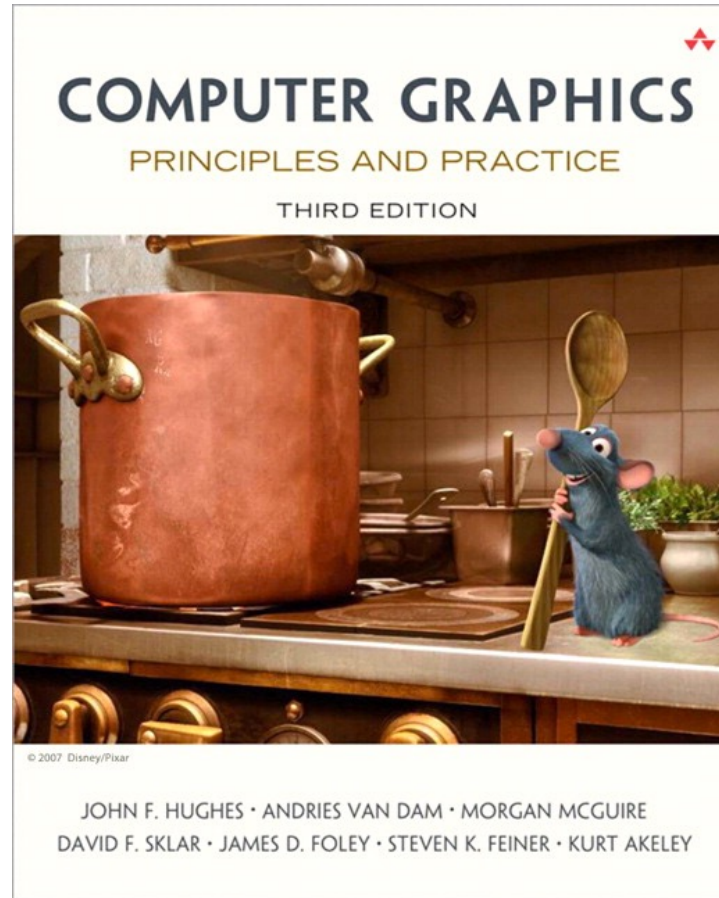
What is computer graphics?

- **Wikipedia:** computer graphics (computer science)

The image shows a screenshot of the Wikipedia article for "Computer graphics (computer science)". A large yellow callout box with a blue border contains the text: "It focuses on the *mathematical* and *computational* foundations of **image generation and processing** ...". Two blue arrows point from the corners of this callout box to the corresponding text in the article's main body. The article content includes a table of contents with sections like "3 Subfields in computer graphics" (Geometry, Animation, Rendering), "4 Notable researchers in computer graphics", "5 See also", "6 References", "7 Further reading", "8 External links" (University groups, Industry), and "Overview [edit]". The overview text states: "Computer graphics studies the manipulation of visual and geometric information using computational techniques of image generation and processing rather than purely aesthetic issues. Computer graphics is often differentiated from the field of visualization, although the two fields have". A small image of the Utah teapot is also visible in the top right corner of the article.

What do the experts say?

Perhaps the most classic computer graphics textbook

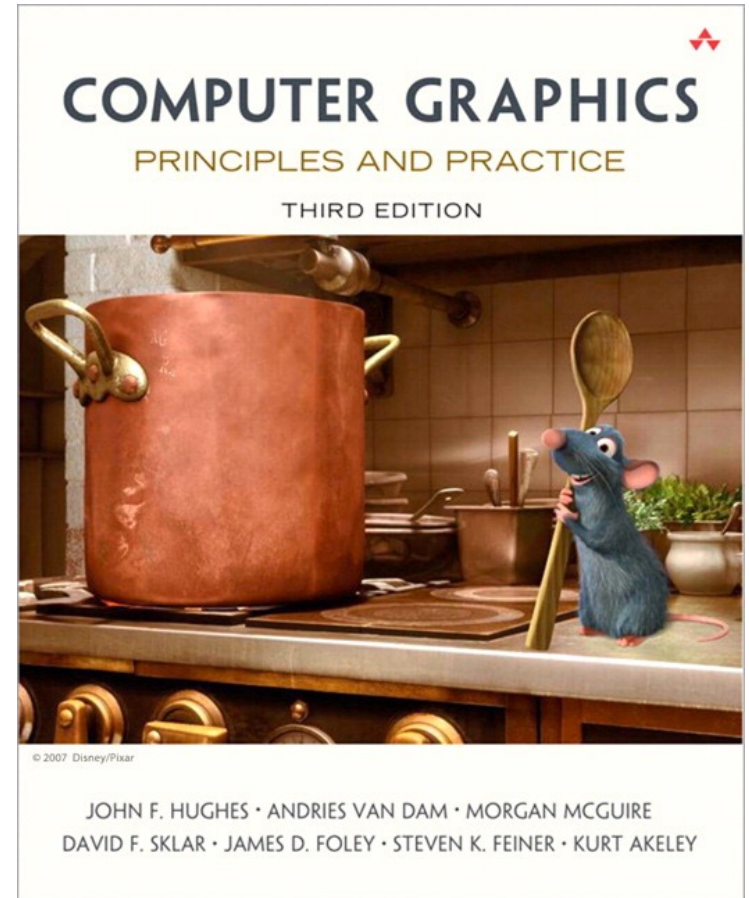


Third edition @ 2014

What do the experts say?

- Hughes, van Dam, et al.:
 - “Computer graphics is the science and art of communicating visually via a computer’s display and its interaction devices.”

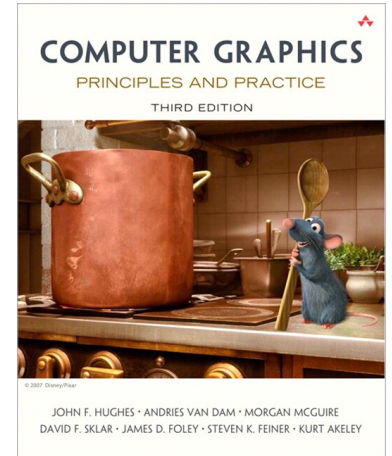
----- *page 1.*



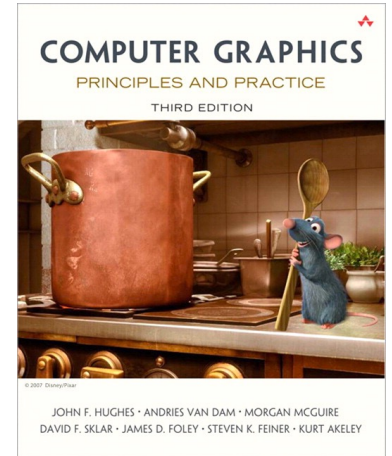
Third edition @ 2014

Classical computer graphics

- Hughes, van Dam, et al.:
 - “Taking a model of the objects in a scene and a model of the light emitted into the scene and producing a representation of a particular view of the scene.”



Classical computer graphics



- Hughes, van Dam, et al.:
 - “Taking a model of the objects in a scene and a model of the light emitted into the scene and producing a representation of a particular view of the scene.”
 - “A glorified multiplication: multiplying incoming light by reflectivity of objects ... for all light reaching the camera”

----- page 2, “A narrow definition”.

List of topics from CMPT 361

- The graphics (vertex & pixel) pipeline
- Transformation, viewing, projection, clipping & visibility
- Light, color, local & global illumination
- Sampling and reconstruction: Fourier transform; aliasing
- Image representation, manipulation, and texture mapping
- Curves, surfaces, meshes, and other geometry reps

All about model **representation** and **rendering**


Classical computer graphics

- **Explicit** scene description is given
- Key problem #1: how to best represent **geometry**, texture, and lighting for the given scene

Classical computer graphics

- **Explicit** scene description is given
- Key problem #1: how to best represent **geometry**, texture, and lighting for the given scene
- Key problem #2: how to render the scene with
 - **Efficiency**

A **forward** problem:

Explicit model description  rendered image

The “forward” problem

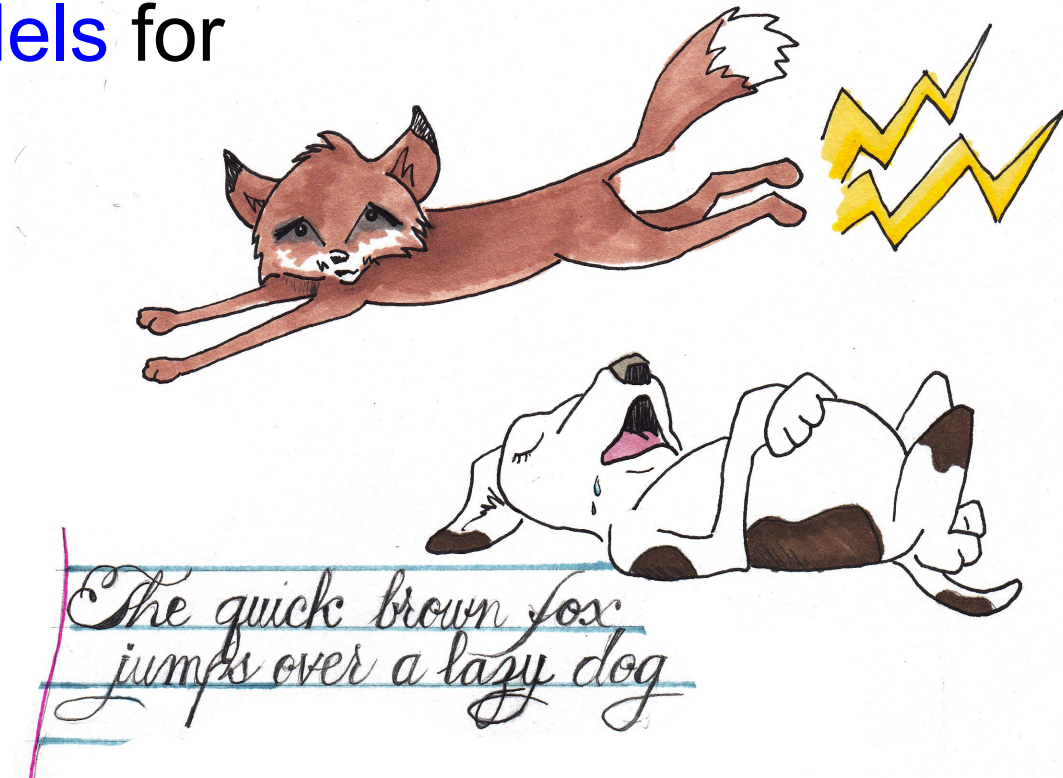
- “The quick brown fox jumps over a lazy dog.”
- Need **explicit models** for
 - A brown fox
 - A dog
 - Quick jump
 - Sleeping dog ...

The “forward” problem

- “The quick brown fox jumps over a lazy dog.”

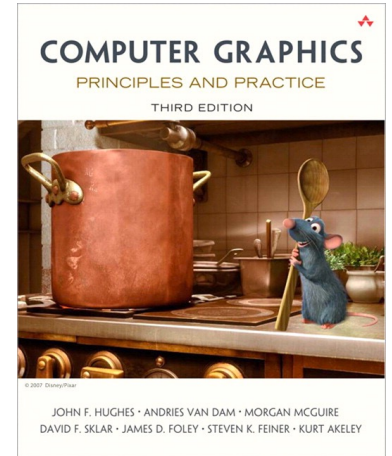
- Need **explicit models** for

- A brown fox
- A dog
- Quick jump
- Sleeping dog ...

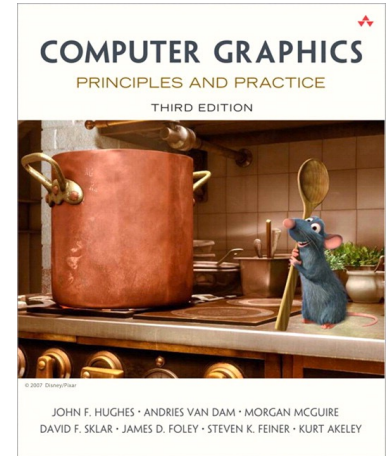


What about computer vision?

- Lower level:
 - **Analysis**: given one view of a scene, determine the illumination and the scene's content, which a graphics system could use to produce the scene



What about computer vision?



- Lower level:
 - **Analysis**: given one view of a scene, determine the illumination and the scene's content, which a graphics system could use to produce the scene
- Higher level: infer an **understanding** of what are

An **inverse** problem:
From a rendered image to a model description

The “inverse” problem

- Ask Claude: Please describe this image.



The “inverse” problem

- Ask Claude: Please describe this image.
 - There is a fox
 - There is a dog
 - Fox jumps over dog
 - Fox is quick
 - Dog is lazy ...



Graphics vs. vision

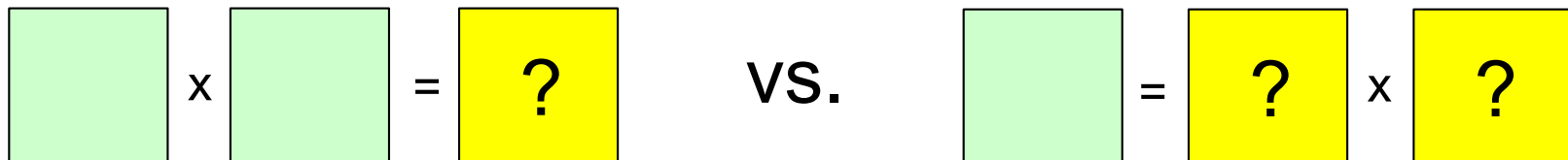
- Graphics is about **synthesis**
 - Classical graphics is about image synthesis

Graphics vs. vision

- Graphics is about synthesis
 - Classical graphics is about image synthesis
- Vision is about image **analysis**

Graphics vs. vision – classically

- Graphics is about synthesis
 - Classical graphics is about image synthesis
- Vision is about image **analysis**
- In classical setting, they **were opposite problems**
 - Forward vs. inverse problems: **which is harder?** 😊



Graphics vs. vision – diff in DATA

- Before jumping to “the new graphics”, what would be the **one big difference** between “data in graphics” and “data in computer vision”?

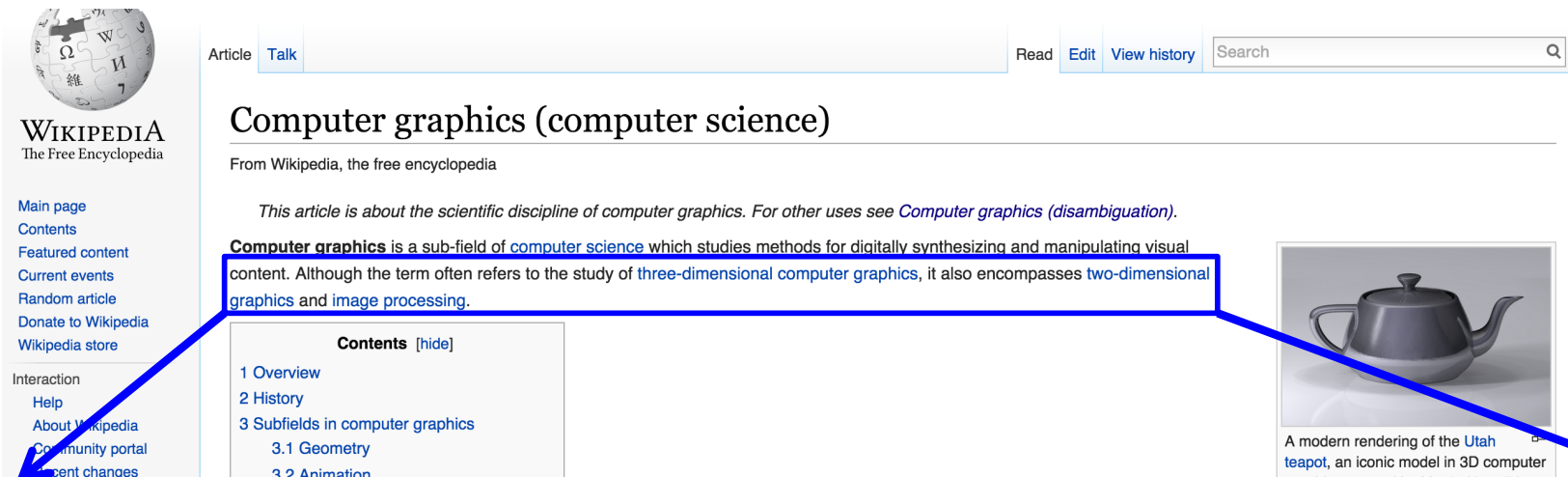
2D image data for computer vision

vs.

3D shape/scene data for computer graphics

Graphics vs. vision – diff in DATA

- **Wikipedia:** computer graphics (computer science)



Article Talk

Read Edit View history Search

Computer graphics (computer science)


From Wikipedia, the free encyclopedia

This article is about the scientific discipline of computer graphics. For other uses see Computer graphics (disambiguation).

Computer graphics is a sub-field of **computer science** which studies methods for digitally synthesizing and manipulating visual content. Although the term often refers to the study of three-dimensional computer graphics, it also encompasses two-dimensional graphics and image processing.

Contents [hide]

- 1 Overview
- 2 History
- 3 Subfields in computer graphics
 - 3.1 Geometry
 - 3.2 Animation



A modern rendering of the Utah teapot, an iconic model in 3D computer graphics.

“Although the term (computer graphics) often refers to the study of **three-dimensional** computer graphics, it also encompasses two-dimensional graphics and image processing ...”

The 3D data challenges

- Acquisition of 3D models is hard
- 3D modeling is hard
- Spatial reasoning and computation in 3D is hard
- Interaction in 3D is hard

3D challenge: acquisition

- Much harder to get 3D chairs than chair images

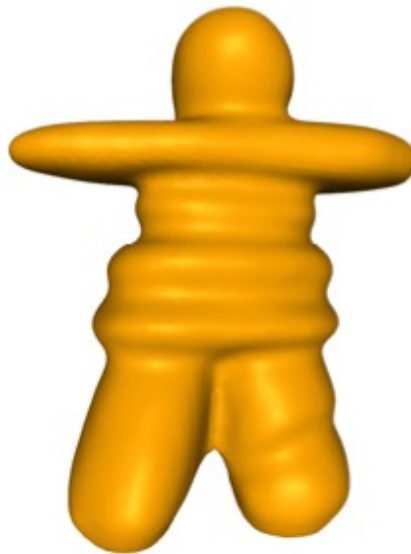


3D challenge: acquisition

- Reconstruction or modeling from 3D or 2D inputs
- Missing data



Laser scan



Poisson

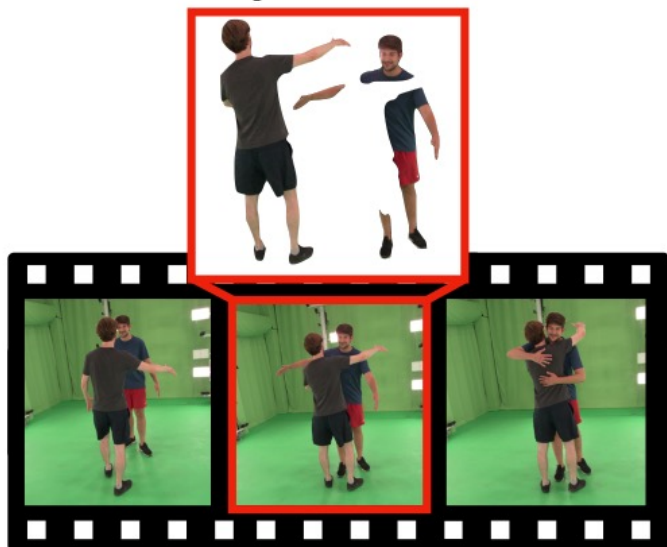


MPU

3D challenge: acquisition

- Reconstruction or modeling from 3D or 2D inputs
- Missing data + **dynamic data**

Occluded partial observations



Input: Monocular Video



Novel View Synthesis

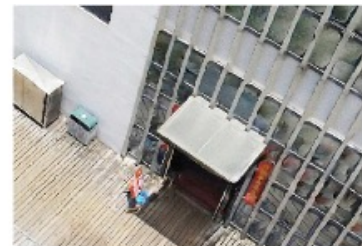
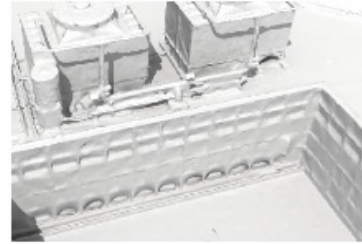
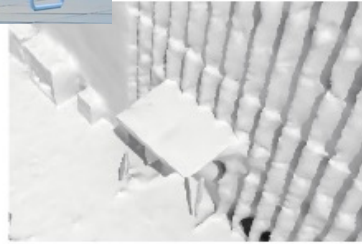
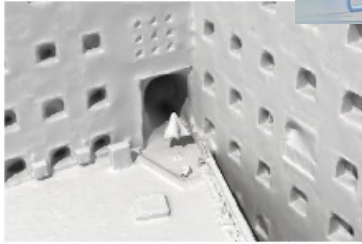


Novel Pose Synthesis

Output: Animatable Person 3D-GS

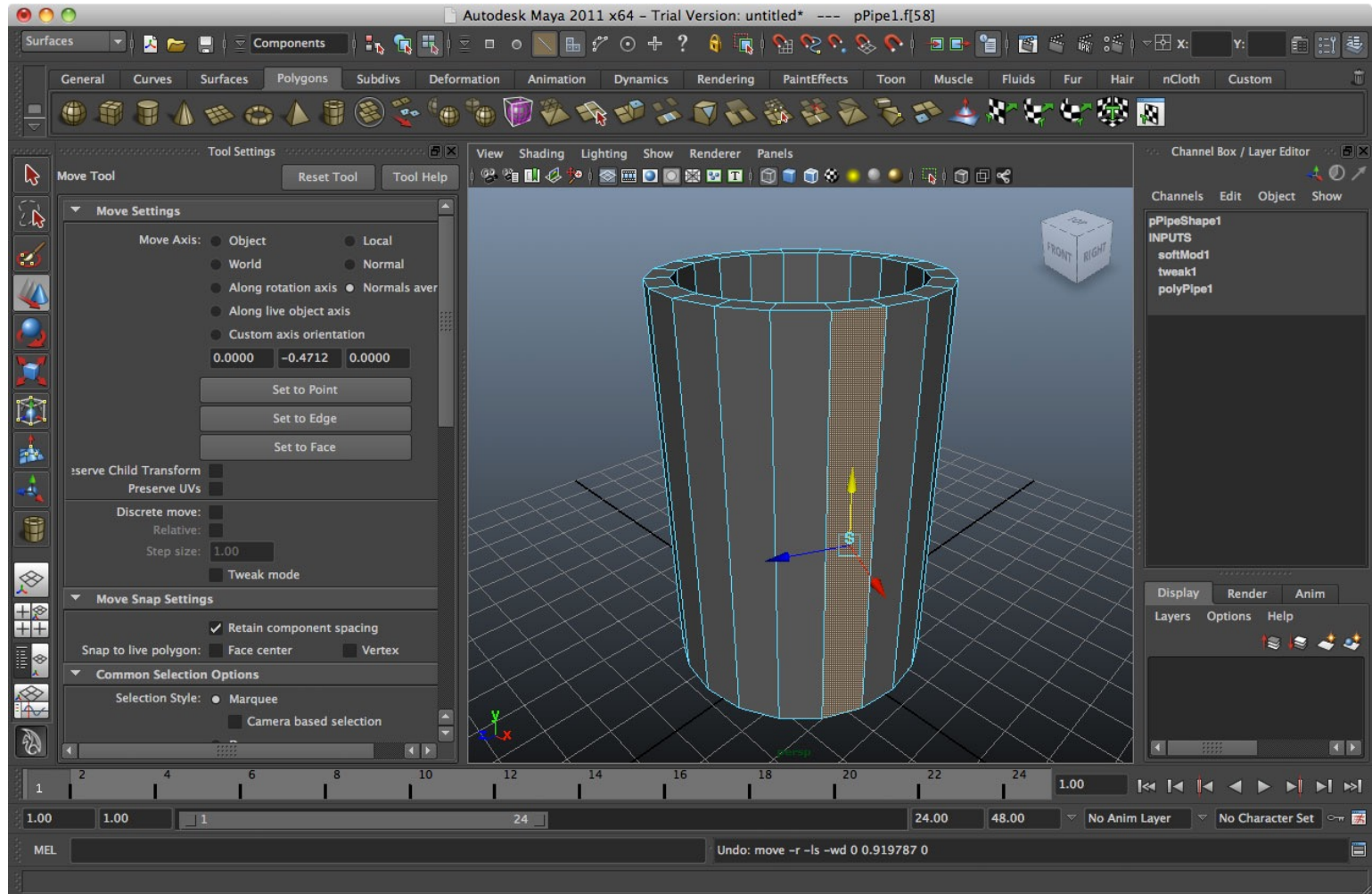
3D challenge: acquisition

- Reconstruction or modeling from 3D or 2D inputs
- Missing data + dynamic data + **large scales**



3D challenge: modeling

Autodesk
Maya



3D challenge: modeling

Autodesk
Maya

Maya User Interface Overview

1 Menu Sets -
While Maya's first seven menus are always available, the remaining menus change depending on which Menu Set you choose. This helps focus your work on related tools.

2 Menus -
Menus contain tools and actions for creating and editing objects and setting up scenes. There is a main menu at the top of the Maya window and individual menus for the panels and option windows.

3 Status Line -
The Status Line contains shortcuts for a number of menu items as well as tools for setting up object selection and snapping. A Quick Selection field is also available that can be set up for numeric input.

4 Shelf -
The Shelf is available to you to set up customized tool sets that can be quickly accessed with a single click. You can set up shelves to support different workflows. Press **Shift + Ctrl** when selecting a menu item to add it to a Shelf.

5 Panel Toolbar -
The panel toolbar rests below the panel menu in each view panel. It lets you readily access many of the frequently used items in the panel menu with a button click. You can toggle view the toolbar by pressing **Ctrl + Shift + M**.

6 Channel Box -
The Channel Box lets you edit and key values for selected objects.

7 Layers -
Maya has three types of Layers.
Display Layers - used to manage a scene.
Render Layers - used to set up render passes for compositing.
Anim Layers - used to blend, lock, or mute multiple levels of animation.

8 QWERTY Tool Box -
The QWERTY hot keys can be used to Select (a), Move (w), Rotate (e), Scale (r) and Show Manipulators (t), as well as access the last tool used (y) in the scene.

9 Quick Layout Buttons -
The Quick Layout Buttons provide predefined configurations of the Maya Workspace. Hold the Right Mouse button over these buttons to give access to more options.

10 Help Line -
The Help Line gives a short description of tools and menu items as you scroll over them in the UI. This bar also prompts you with the steps required to complete a certain tool workflow.

11 Time Slider -
The Time Slider shows you the time

12 Range Slider -
This bar has an area to the left for inputting simple MEL commands and an area to the right for feedback. You will use these areas if you choose to become familiar with Maya's MEL scripting Language.

Workspace -
The workspace consists of multiple panels that offer different ways of creating and evaluating your scenes.

14 Animation/Character Menus -
The Animation or Character menus allow you to quickly switch the animation layer or current character set.

13 Playback Controls -
The Playback controls let you move around time and preview your animations as defined by the Time Slider range.

14 Animation/Character Menus -
The Animation or Character menus allow you to quickly switch the animation layer or current character set.

15 Default Layer -
In all cases, there is a default layer where objects are initially placed upon creation.

3D modeling is definitely not a job for everyone!

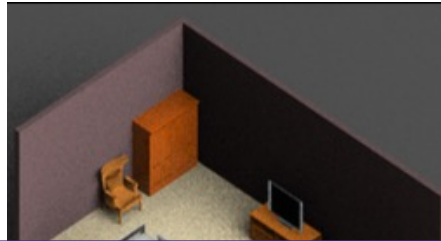
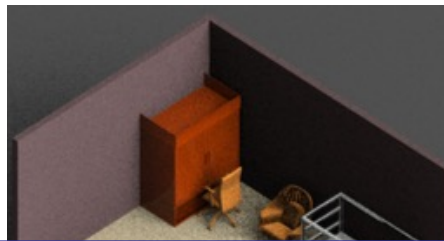
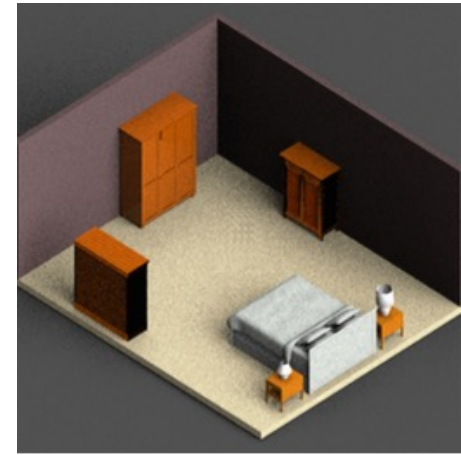
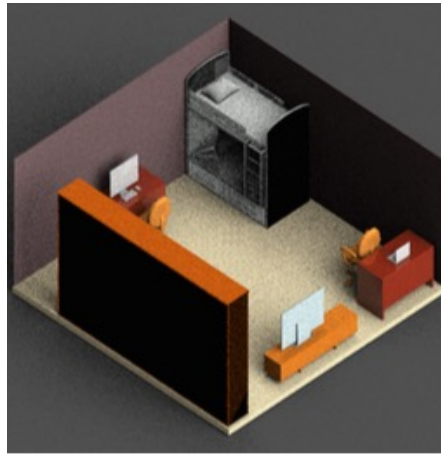
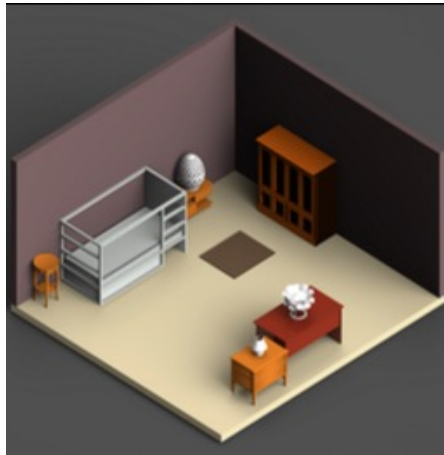
3D reasoning is hard

- Humans very good at pattern recognition (vision)
- But not so good at 3D reasoning or manipulation



But we live in a **3D world** and many real-world problems are 3D!

Many real-world problems are 3D



How to easily generate many 3D indoor scenes that are realistic and diverse, e.g., for AR/VR?

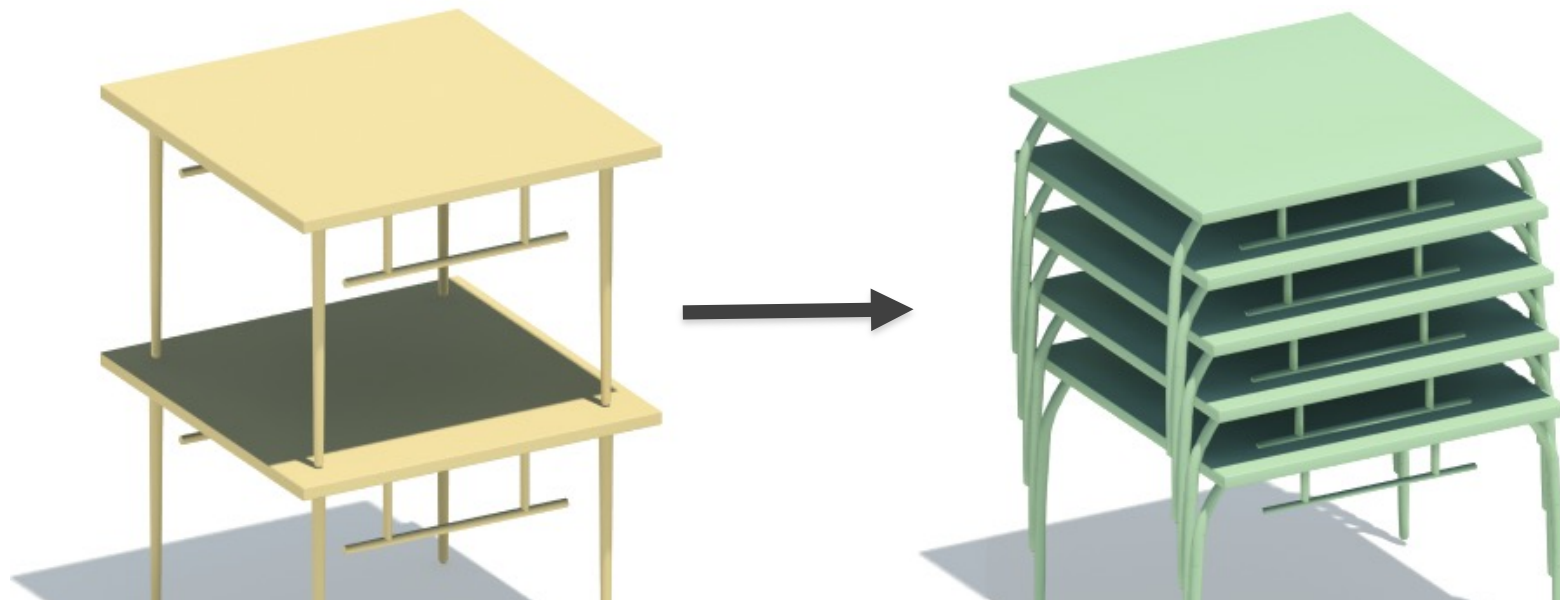
Many real-world problems are 3D

- How to **subtly** make the table **stackable**?



Stackabilization

- How to **subtly** make the table **stackable**?



Requires **precise measurement and transform of 3D** objects: difficult for human users to model

Foldabilization

- How to **subtly** make the chair **foldable**?

Like solving a puzzle:
acute 3D spatial reasoning
skills are needed

Computationally hard with
very large search space



[Li et al. *Siggraph* 2015]

Foldabilization

EMBRY-RIDDLE
Aeronautical University™
SCHOLARLY COMMONS

International Journal of Aviation,
Aeronautics, and Aerospace

Volume 8 | Issue 1

Article 1

2021

Folding Methodology for Flexible Aircraft Interiors

Aditya Venkatesh

Ryerson University, Toronto, Canada, avenkatesh@ryerson.ca

Fengfeng (Jeff) Xi

Ryerson University, Toronto, Canada, fengxi@ryerson.ca

Joon Chung

Ryerson University, Toronto, Canada, j3chung@ryerson.ca

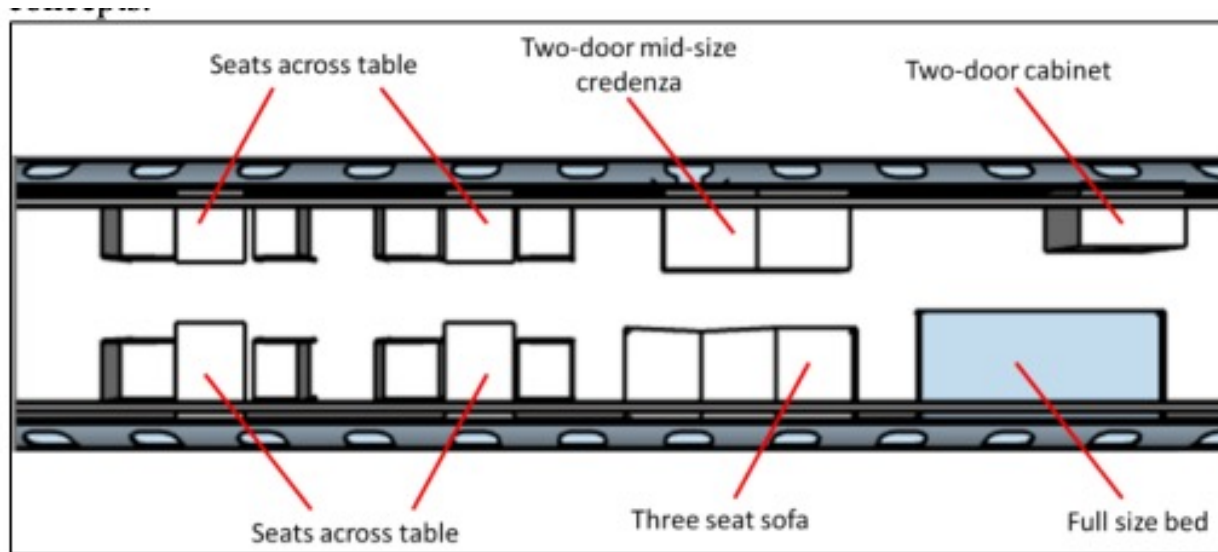


Figure 3. Partially sectioned top view with extended furniture pieces.

Figures 4 and Figure 5 illustrate the Open-on-Demand concept along the starboard and port sides of the partially sectioned cabin with zero-thickness furniture pieces. The area highlighted in yellow refers to the floor space the furniture pieces occupy in their extended forms.

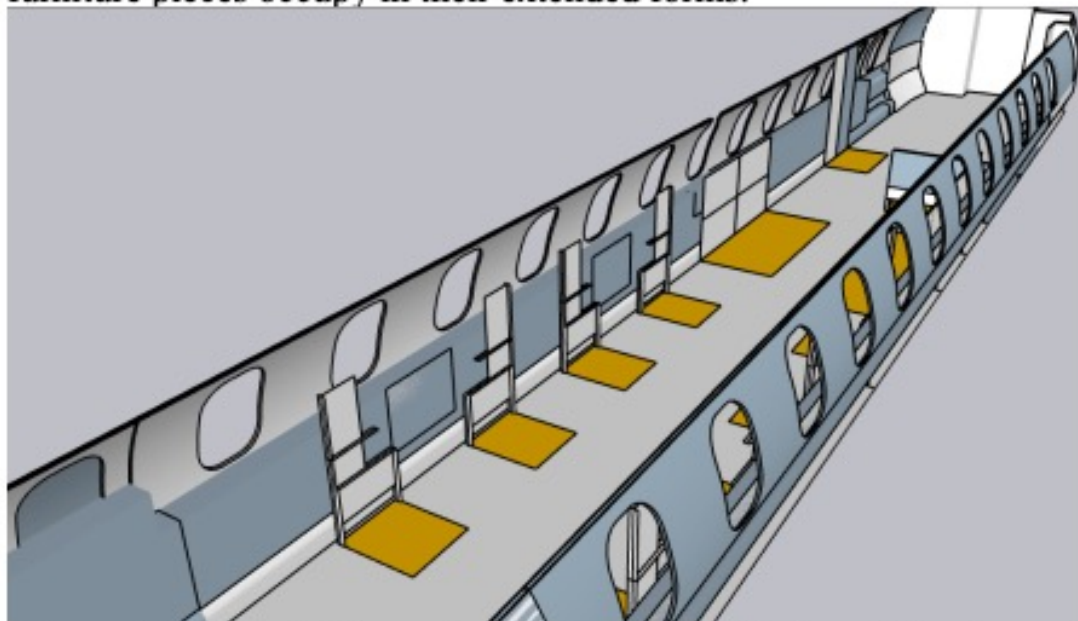


Figure 4. Starboard side partial section view with stowed furniture pieces.

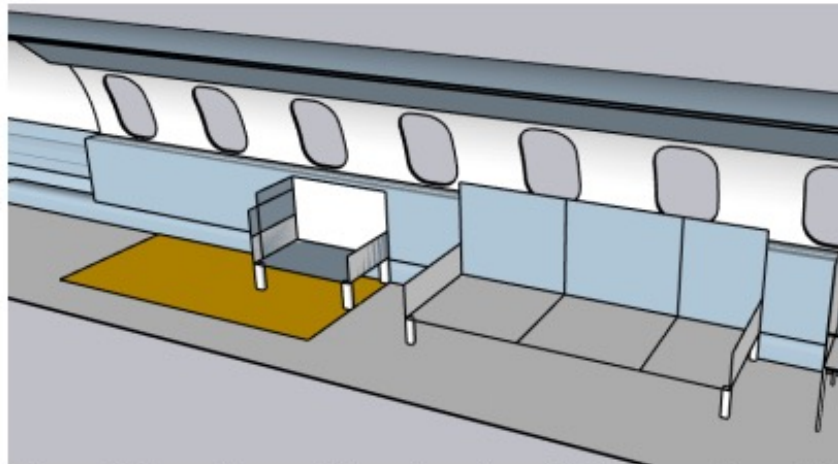


Figure 9. Port side partial section view of a bed reconfigured into a seat.

The furniture folding methodology can also be quantitatively assessed to analyze advantages. A zero-thickness seat frame with measurements reflecting the lounge seat on a business jet was recreated using SketchUp 2020. Figure 10 illustrates an extended seat and stowed seat, following the furniture folding methodology, with a length of 27.25", width of 31.33" and height of 44.23" and new a length of 2", width of 31.33" and height of 44.23", respectively.

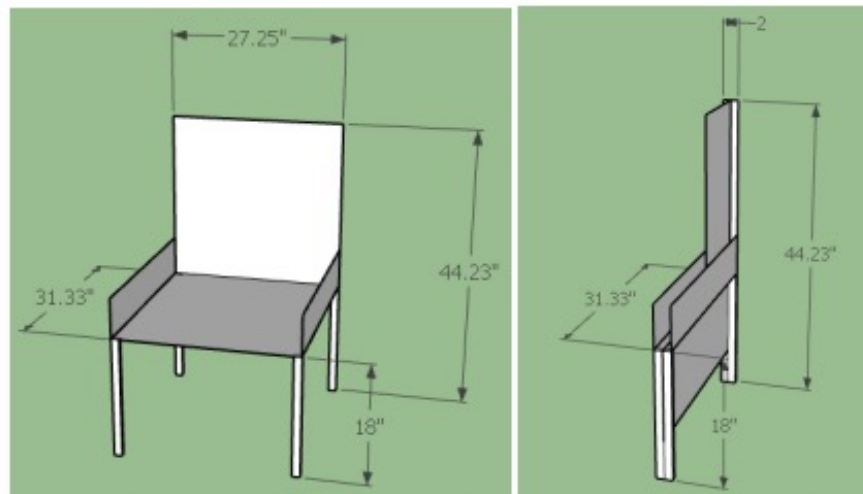
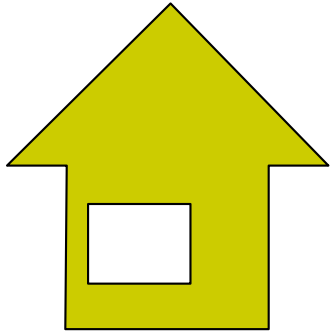


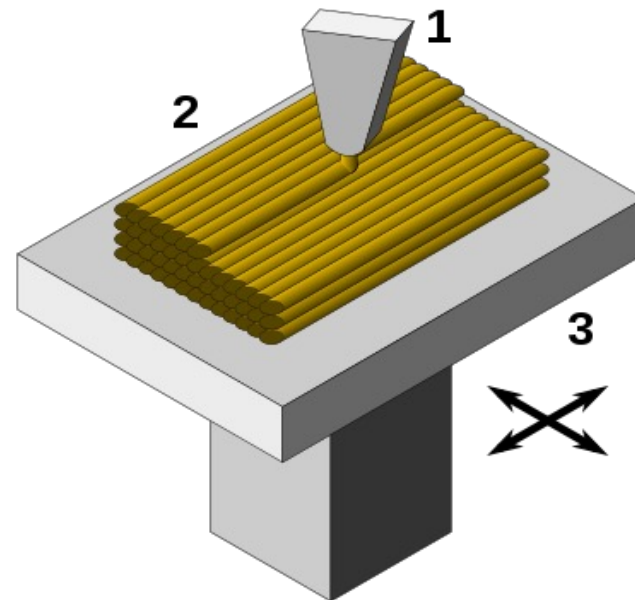
Figure 10. Recreation of an extended(left) and stowed(right) zero-thickness business jet lounge seat frame.

A cool decomposition for 3D printing

- How to decompose into few terrain-like parts?



Terrain-like part



Zero material waste for
layer-based 3D printing

Pyramidal decomposition

- How to decompose into few terrain-like parts?



[Hu et al., *Siggraph Asia* 2014]

Pyramidal decomposition

- How to decompose into few terrain-like parts?

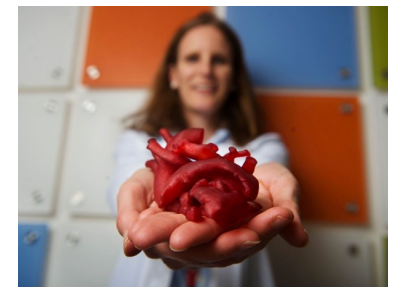
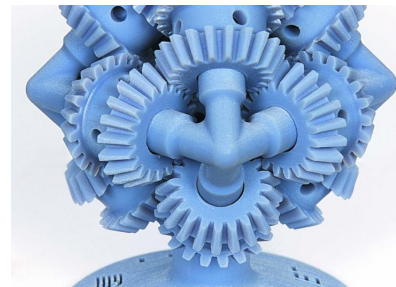
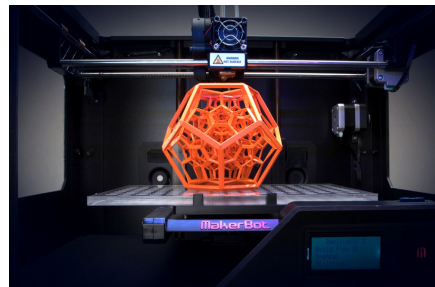


3D problem is provably NP-hard

Computer graphics is responsible for addressing the various **3D data challenges**

3D printing may be a blessing

- Graphics likes 3D to be wanted & used everywhere
- The internet has not made 3D data ubiquitous as promised: remember VRML around 15 years ago?
- **3D printing** just might!



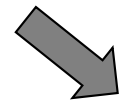
New graphics

- Keep doing synthesis, but **focus on modeling**
- Synthesis of **all visual contents**, not just images

explicit model
description



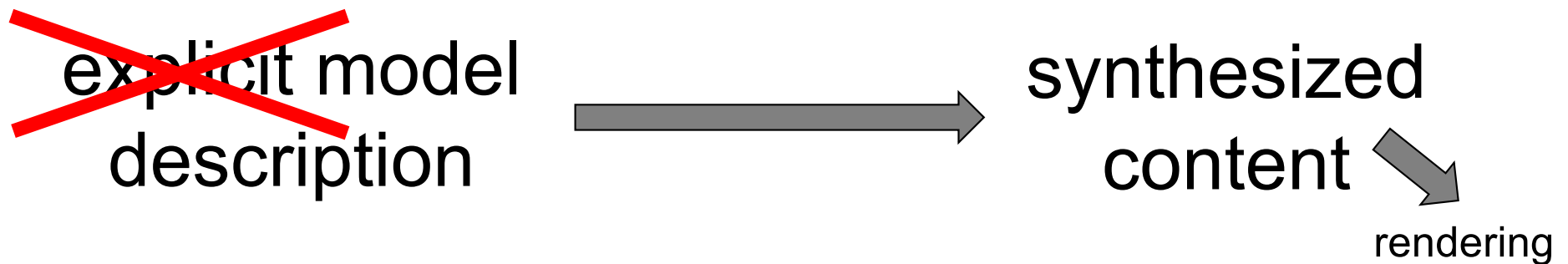
synthesized
content



rendering

New graphics: no explicit model

- Keep doing synthesis, but focus on modeling
- Synthesis of all visual contents, not just images



Model description is only abstract (e.g., **texts** or a **sketch**), hard to quantify (**functional** or **creative**), or unknown entirely (input = set of **examples**)

New graphics: novel content

- Synthesis and manipulation of images

novel + **visual contents**

Implicit or abstract
inputs, examples, ...



novel content

New kinds of inputs



A rough sketch

New kinds of inputs



A rough sketch



One or more
images

New kinds of inputs



A rough sketch



One or more
images



Just some examples

Text-to-3D



“A baby bunny sitting on top of a stack of pancakes”
3D model generated from text [Zhu et al. 2023]

Text-driven 3D scene synthesis

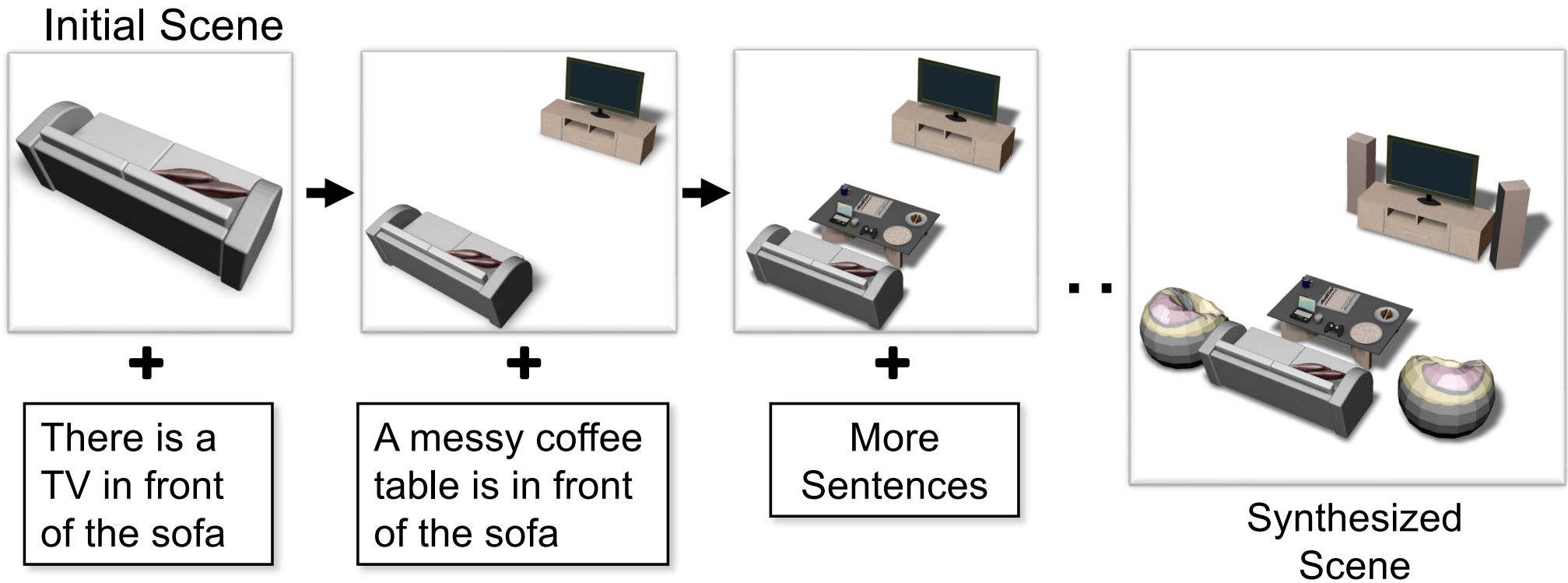


Photo-inspired modeling

- 3D model from a **single** photograph



?



Has to be data-driven

- Abstract inputs: **ill-posed** synthesis problem
- Needs extra **knowledge**, e.g., pre-existing dataset



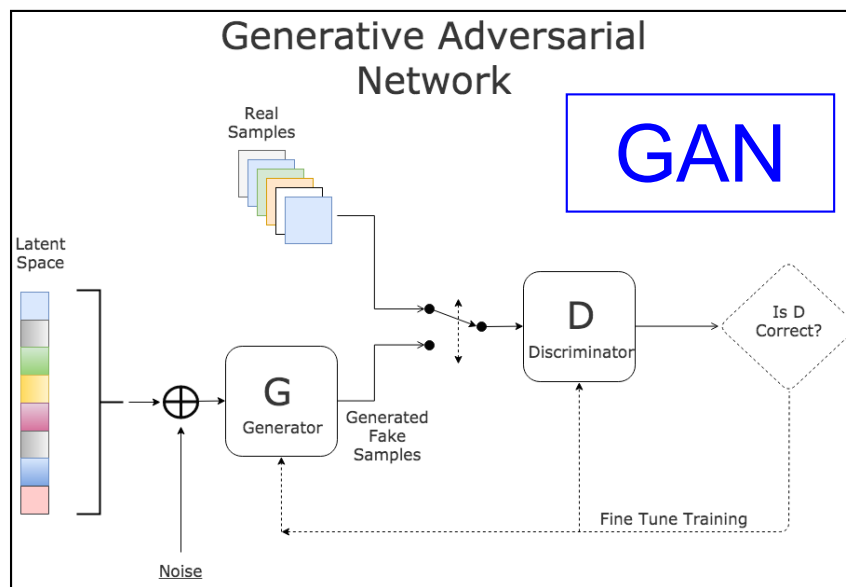
Example-based modeling

- Key: **understand** the set of examples by **learning**
- To infer **commonality** and **diversity** in the set



Inverse modeling

- Learn a **generative model**, e.g., from examples
- Apply the model forwardly, maybe with a **random input**, to synthesize novel contents





In the new graphics

We generate/synthesize objects/scenes that are

- **Plausible**: “What makes a chair a chair?”
- **With the right style**: “Essence of Gothic style?”
- **Functional**: “Shape vs. functional similarities?”
- **Ergonomic**: “How to quantify human comfort?”
- **Creative**: “What is a model of human creativity?”
- ...



To generate/synthesize

2D/3D objects or scenes that are

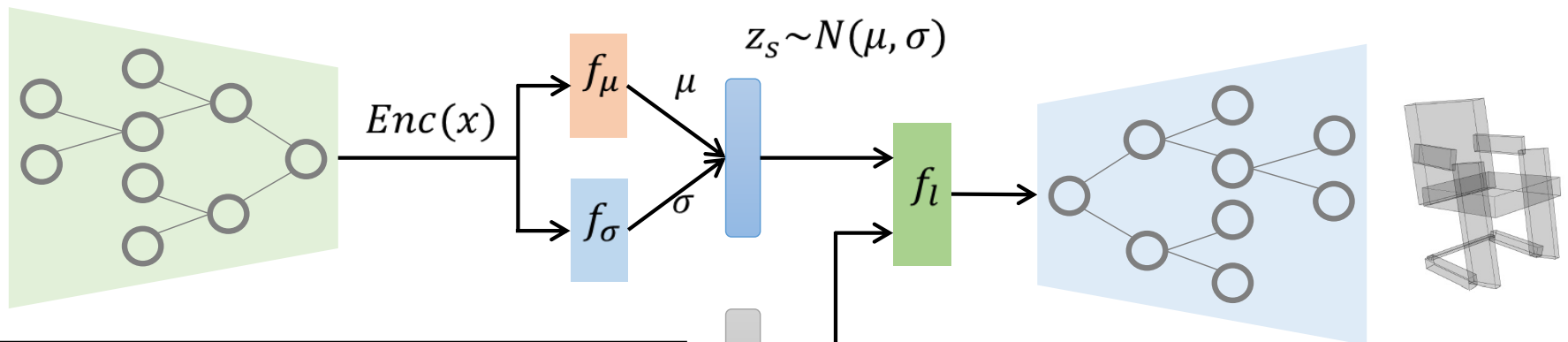
- Plausible, stylistically compatible, aesthetically pleasing, functional, ergonomic, or creative, etc.
- We must first **learn** plausibility, style, function, etc.
- To generate from texts/sketches, we must **learn** the right mapping/regression model

Workflow of new graphics research

- **Analysis** to acquire **understanding** of
 - grouping/clustering patterns, object/scene composition, human activities, styles, functionality, creativity, etc.
- **Synthesis** of images, shapes, scenes via
 - interactive modeling, genetic algorithm, statistic models, deep regression and generative **neural networks**, etc.

Example: generative structural reps

- Generative autoencoders via recursive neural nets



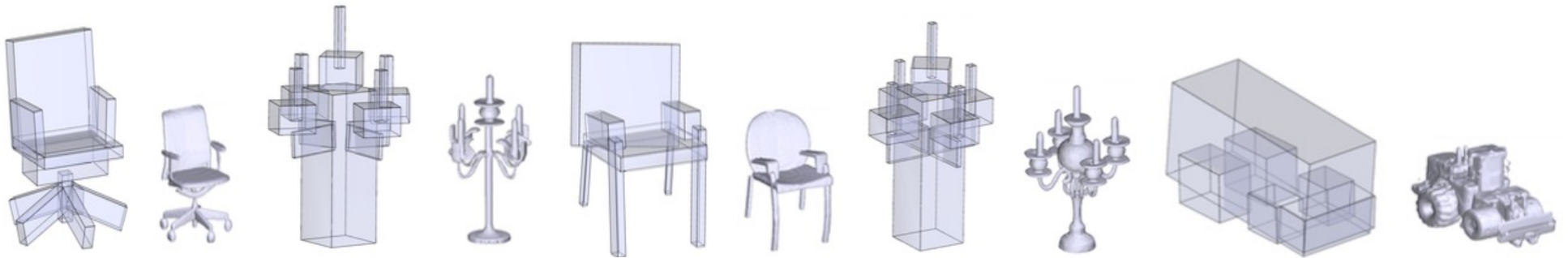
Encoder is trained to learn recursive/hierarchical grouping of words in a sentence, regions in an image, parts in a 3D shape, or objects in an indoor scenes

Decoder, the generative or synthesis model, is trained to convert a random code into a hierarchy

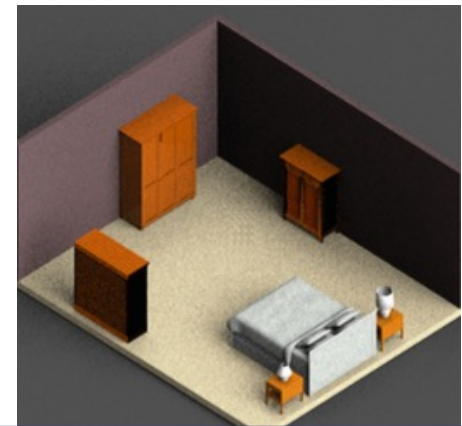
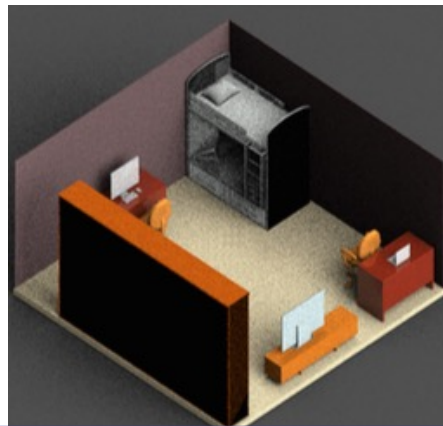
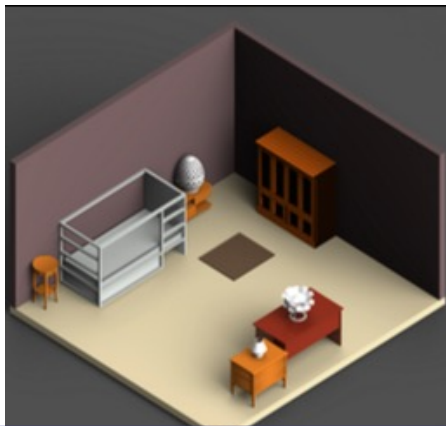
GRASS: Generative Recursive Autoencoders for Shape Structures [SIG 2017]

Example: generative structural reps

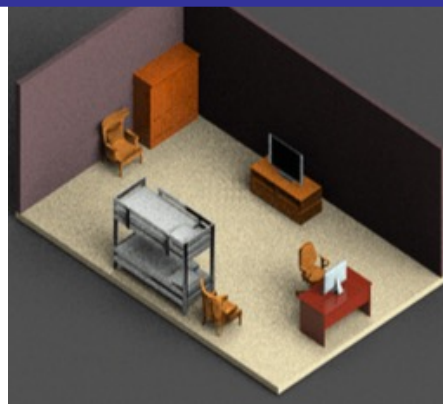
- Combine autoencoder with GAN: **VAE-GAN**
- **Structure-aware; coarse-to-fine; clean parts; high res**



Example: scenes from random codes



Takes less than one second to generate a 3D scene!



GRAINS: Generative Recursive Autoencoders for INdoor Scenes [2018]

New graphics: not forward problem

- Inverse analyses and learning generative models
- Keying on shape/scene understanding
- Only with a good understanding of a shape/scene category (“bicycle” or “kitchen”) can one recreate!

Important problems in new graphics

- Modeling from abstract description
- Modeling from few examples
- Inverse procedural modeling
- Learning generative neural networks

Knowledge, learning, and data
play the key roles!

Our new view of graphics



- Wikipedia was already catching up:
 - Something I hid: Computer graphics = methods for digitally **synthesizing** and manipulating **visual content**
- From image production to all (3D) visual content

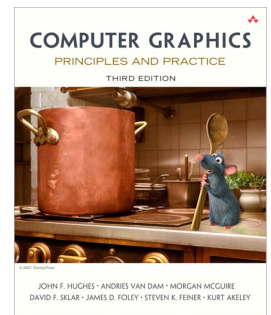
Novelty

of the synthesized content is the BIG challenge!

Current graphics and computer vision

- Hughes, van Dam, et al.:
 - “Much of **current research in graphics** is in methods for **creating geometric models**, methods for representing surface reflectance, the animation of scenes ..., and in recent years, an **increasing integration of techniques from computer vision.**”

----- *page 2 of*



Computer graphics vs. vision again

- Shape **understanding and inverse modeling** are very much “vision-like” research problems
- So, graphics is “catching up” 😊
- But could we do more?

Steven A. Coons Award



weta
DIGITAL

Jim Kajiya: What human capabilities does each CS discipline try to enhance/replace?

Artificial intelligence:
human intelligence

Computer vision:
pattern recognition

What about computer graphics?

Human imagination!

Steven A. Coons Award



Graphics and imagination

- Think of the various VFX we see in films and games
- Think of the VR/AR/MR CG has helped create

Computer graphics allows our imagination to be realized into (virtual) reality!

It allows a mental concept to turn into a digital representation, and now fabricated!

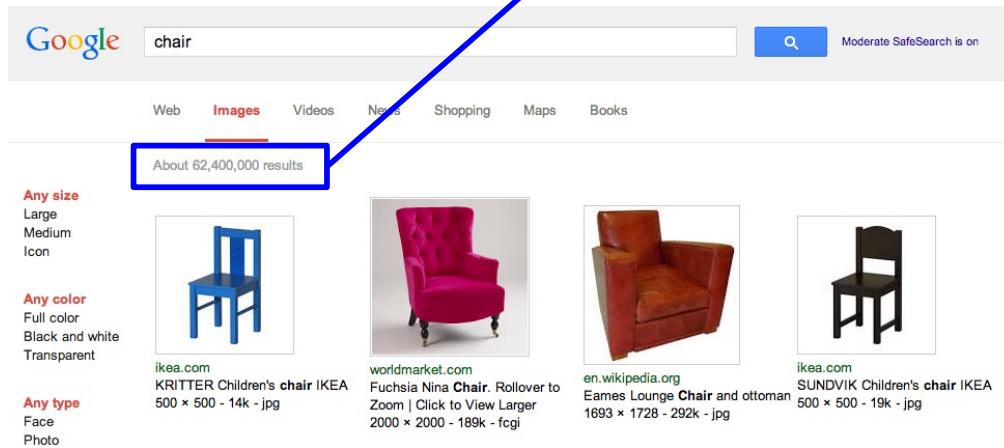
Still a long way to go

- Only scratching the surface in the new graphics
- Smart ideas: data-driven, data reuse, co-analysis, supervised learning, active learning, etc.
- Future of modeling in computer graphics

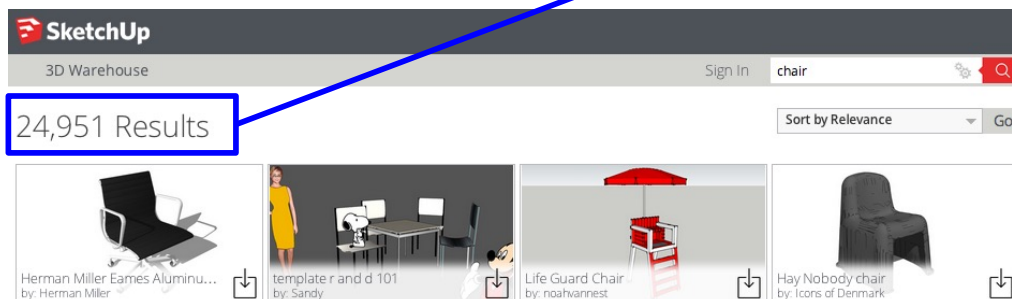
Data + knowledge + learning

A new 3D data challenge

Google image search for chair: 64,000,000 results

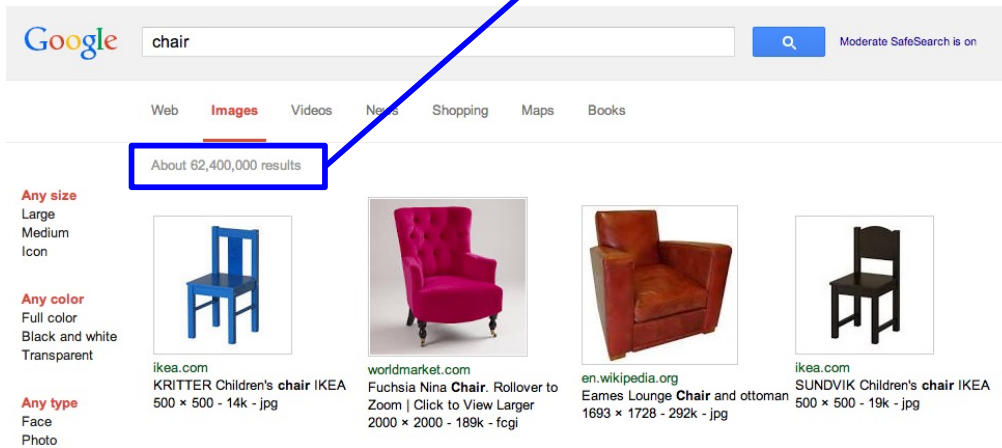


3D Warehouse: 24,951 results



No “Big 3D Data” yet

Google image search for chair: 64,000,000 results



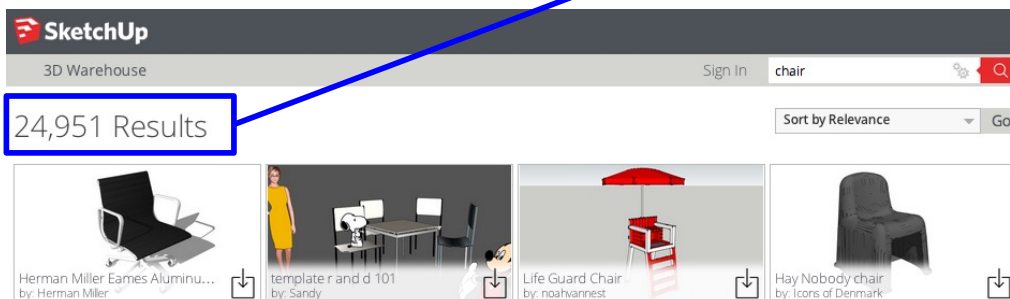
- Bicycles

29,900,000 vs. 1,225

- Strollers

5,070,000 vs. 36

3D Warehouse: 24,951 results



Problems with “_{small} 3D data”?

- Lack of knowledge for learning-based 3D analyses
- Lack of examples for example-driven 3D syntheses
- Small data is **the** detriment to

Data + knowledge + learning

To fix the “small 3D data” problem

- Need to **synthesize** more and more 3D models!
- Not only volume, but **variation, variety, and novelty!**

Novelty

of synthesized content may **enhance knowledge!**

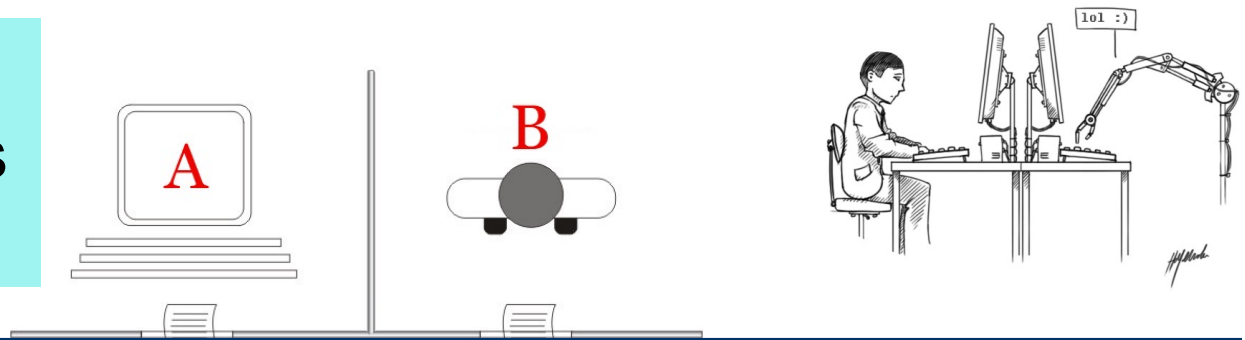
Computer graphics is responsible for producing
such Big 3D Data

What about a machine becoming human?

When is a machine becoming human?

- Well-known **Turing test**: indistinguishability between human and machine in **natural conversation**

Multiple human judges vote who is human/machine



Turing (1950's) predicted the test would be passed around **year 2000**. An easy version of the test was passed in 2014 by *Eugene*, a ChatBot.

When is a machine becoming human?

- Total Turing test: machines linguistically and physically indistinguishable from a human



Hans Moravec (1999): Total Turing Test to be passed by the year 2040

Is this Turing test too easy?

- Ada Lovelace (1815-1852)
 - Pre-dates Turing (1912 – 1954)
 - Worked on world's first computer
 - AKA: the **first computer programmer**

*Computers can't **create** anything (Humans can!). Creation requires **originality**. But computers originate nothing; they only do that what we order them, via programs, to do.
- Ada Lovelace*





A harder test: Lovelace Test

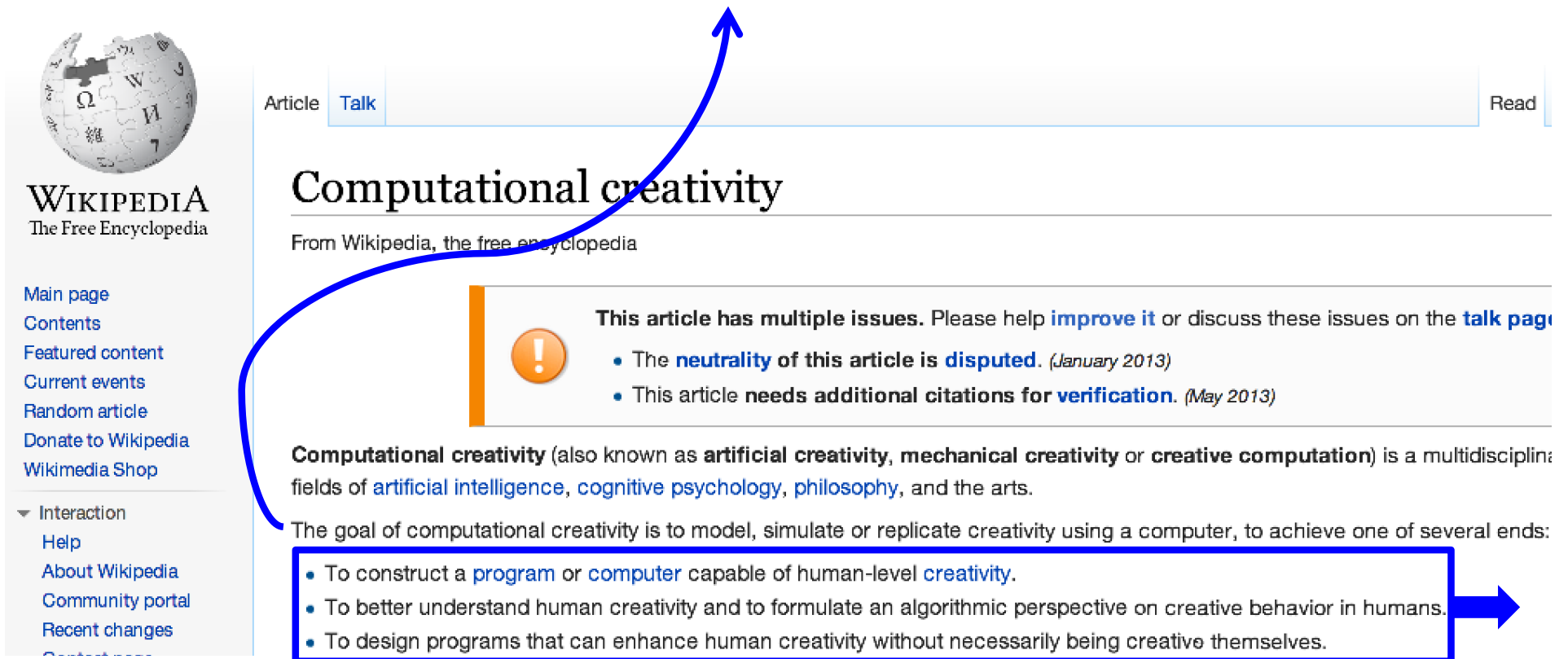
- Test on machines' **ability to create** an artifact
 - e.g., a story, poem, painting, or a **3D shape**
- Test or judging criterion can vary

Key message: What really separates humans from machines is not the ability to make conversation, but **the ability to create!**

Creation and synthesis of visual content is the goal of **computer graphics!**

Computational creativity

- Goal: model or simulate creativity using a computer



The image shows a screenshot of the Wikipedia article for 'Computational creativity'. A blue arrow points from the article title to the goal listed in the slide above. Another blue arrow points from the 'This article has multiple issues' box to the goal list. A third blue arrow points from the goal list to the right.

WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikimedia Shop

Interaction
Help
About Wikipedia
Community portal
Recent changes

Article **Computational creativity** Talk Read

From Wikipedia, the free encyclopedia

This article has multiple issues. Please help [improve it](#) or discuss these issues on the [talk page](#).

- The **neutrality** of this article is **disputed**. *(January 2013)*
- This article **needs additional citations for verification**. *(May 2013)*

Computational creativity (also known as **artificial creativity**, **mechanical creativity** or **creative computation**) is a multidisciplinary field of [artificial intelligence](#), [cognitive psychology](#), [philosophy](#), and the arts.

The goal of computational creativity is to model, simulate or replicate creativity using a computer, to achieve one of several ends:

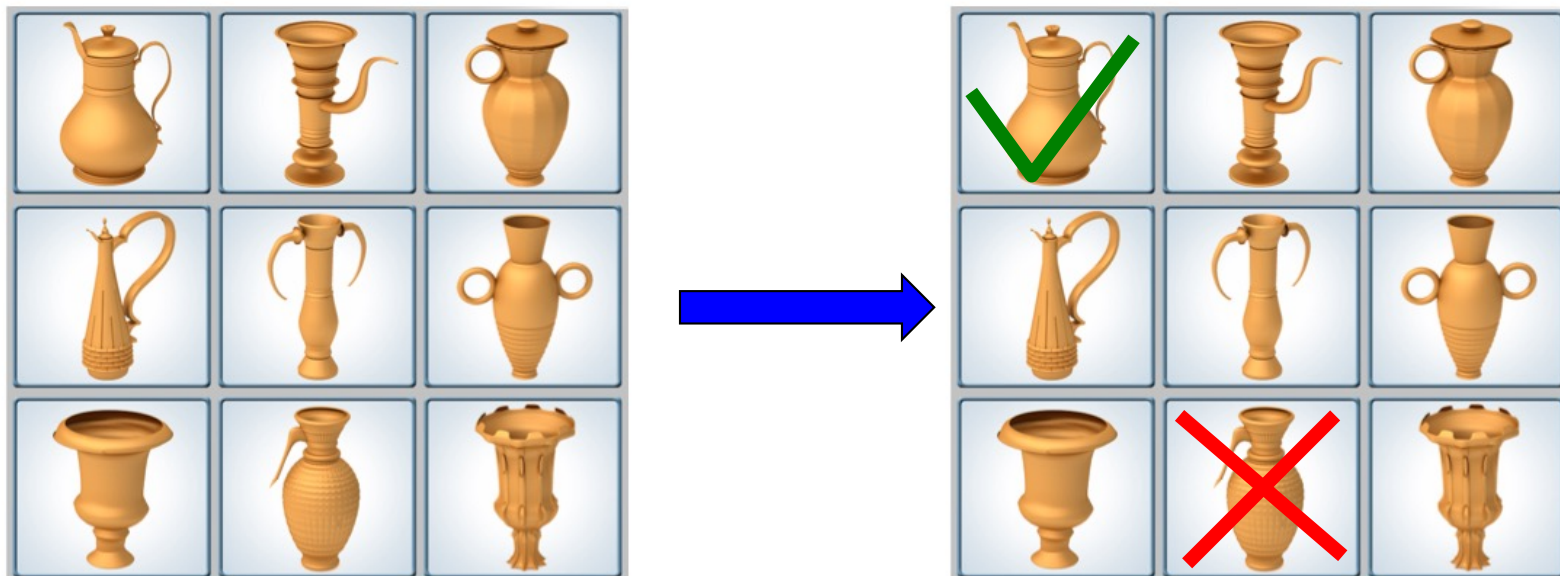
- To construct a [program](#) or [computer](#) capable of human-level [creativity](#).
- To better understand human creativity and to formulate an algorithmic perspective on creative behavior in humans.
- To design programs that can enhance human creativity without necessarily being creative themselves.

Goals of computational creativity

- *** Construct a program capable of human creativity
- ** Understand creativity and formulate an algorithmic perspective on creative behavior in humans
- * Design a program which may enhance human creativity without the program being creative itself

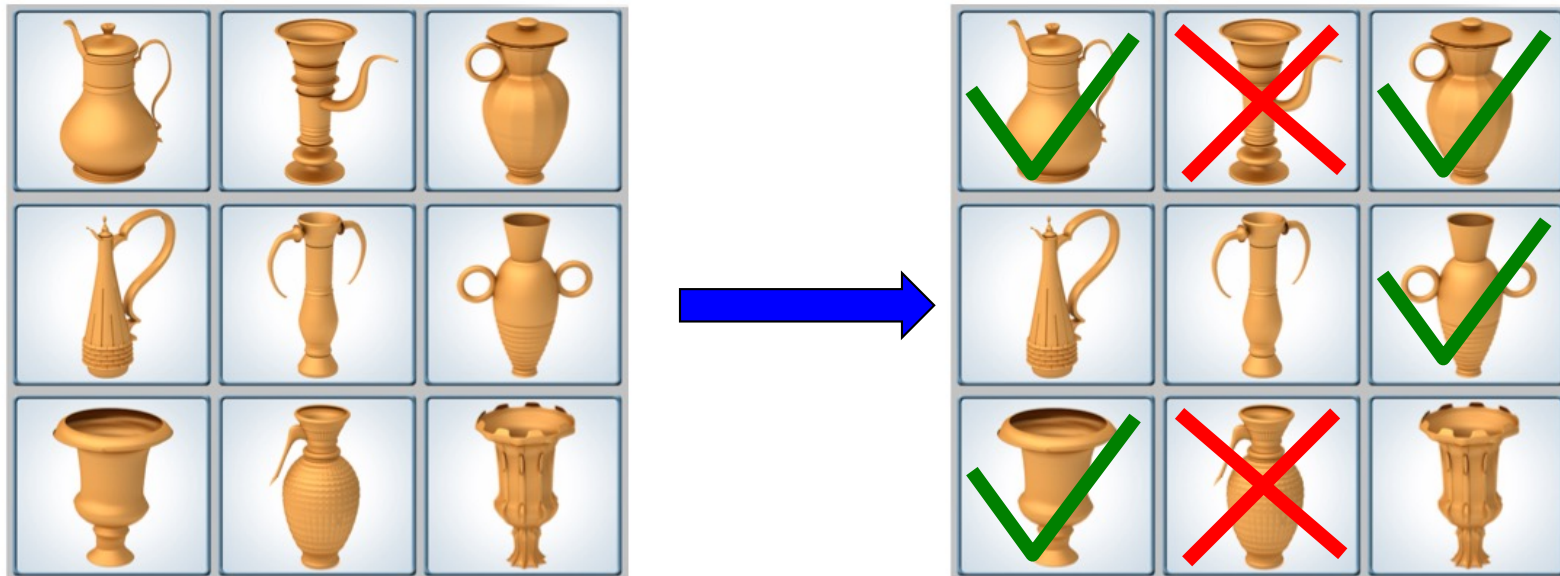
Creative 3D modeling

- Creativity: machines stochastically generate models
- Has to be controlled



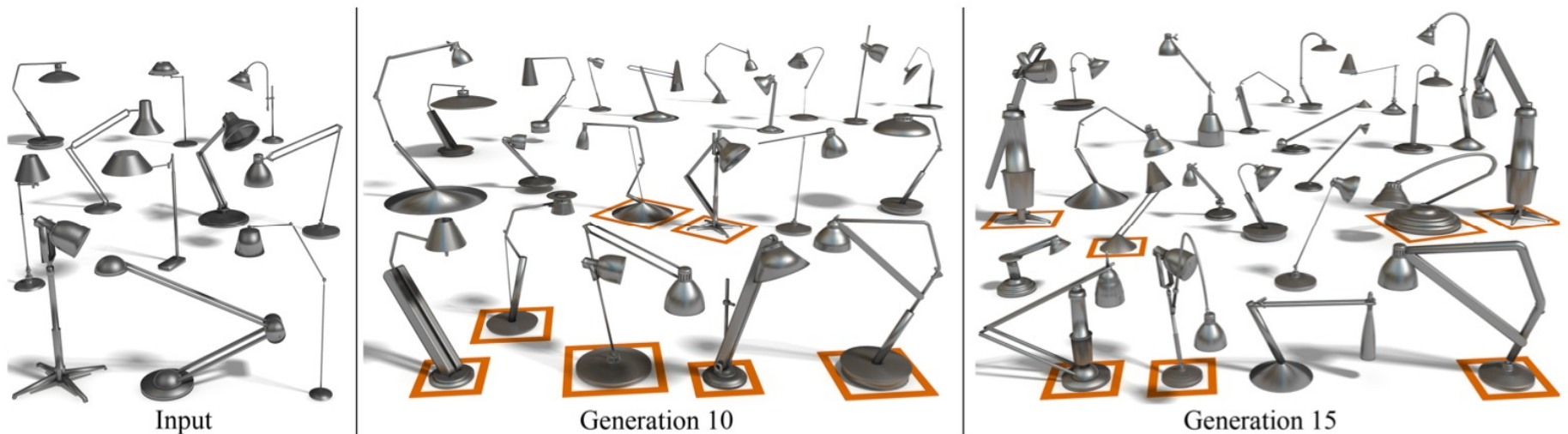
Creative 3D modeling

- Creativity: machines stochastically generate models
- Control by humans operating on a “design gallery”



Creative 3D modeling via evolution

- **Evolve** an entire **set** (initial population) to obtain generations of **fit and diverse** new creations



Creative 3D modeling

- Allows creative generation of **novel** 3D contents



Genetic algorithm + Interactive modeling

Very far from an intelligent machine that is creative

Machine to generate creative logos?

The logo for 'ZAP' features the letters 'ZAP' in a bold, black, sans-serif font. The right side of the 'Z' and 'P' is filled with a black and white checkerboard pattern.The logo for 'HERO' has the word 'HERO' in white, sans-serif capital letters on a black background. Above the 'O' is a white silhouette of a person with arms raised in a 'V' shape.The logo for 'DISCO' uses the word 'DISCO' in a bold, black, sans-serif font. Each letter is filled with a black vinyl record, and the orange center labels are visible.A single black letter 'C' on a light gray background.The logo for 'CLOCK' features the word 'CLOCK' in a black, sans-serif font. The letter 'O' is replaced by a black circle with a smaller black circle inside, resembling a clock face.The word 'Smile' is written in a black, sans-serif font. The letter 'i' has a black dot above it that curves into a smile.The word 'bird' is written in a black, sans-serif font. The letter 'i' has a black dot above it that curves into a bird's beak.A black silhouette of a lightbulb on a light gray background.The logo for 'killed PRODUCTIONS' features the word 'killed' in a bold, black, sans-serif font with a white dot over the 'i'. Below it, the word 'PRODUCTIONS' is written in a smaller, white, sans-serif font on a black background.The logo for 'GUITAR STUDIO SCHOOL OF EXCELLENCE' features the word 'GUITAR' in white, sans-serif capital letters above the word 'STUDIO' in a larger, white, sans-serif font. Below 'STUDIO' is a white silhouette of a guitar body. To the right, the words 'SCHOOL OF EXCELLENCE' are written in a smaller, white, sans-serif font.The word 'HIDE' is written in a bold, black, sans-serif font. The letter 'I' is replaced by a black silhouette of a hand holding a knife.The word 'HELP' is written in a bold, black, sans-serif font. The letter 'L' is replaced by a black silhouette of a hand with fingers spread.A single black letter 'G' on a light gray background.A single black letter 'H' on a light gray background.

What can we do now?

[Tanveer et al. /ICCV 2023]

 BUTTERFLY

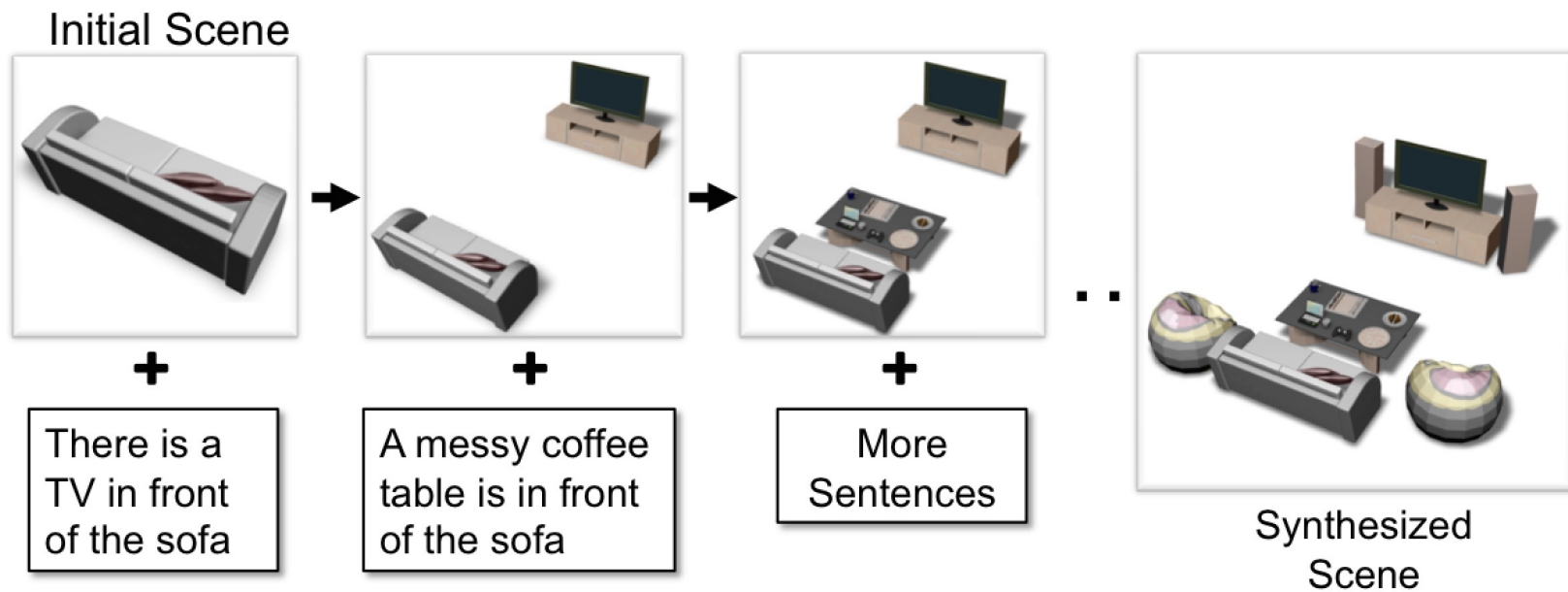
Creativity is not a talent, it is a way of **operating**.

— John Cleese

Graphics in the presence of Big data

Graphics is responsible for

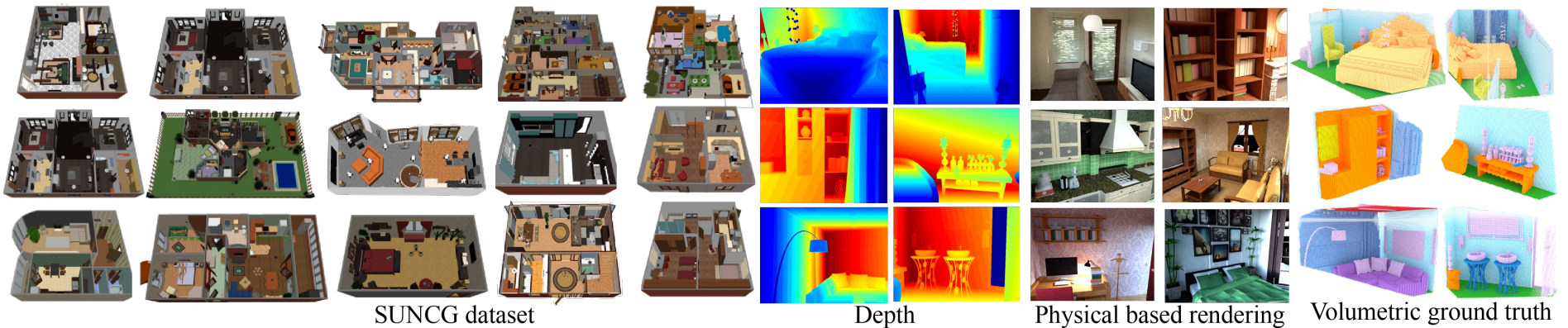
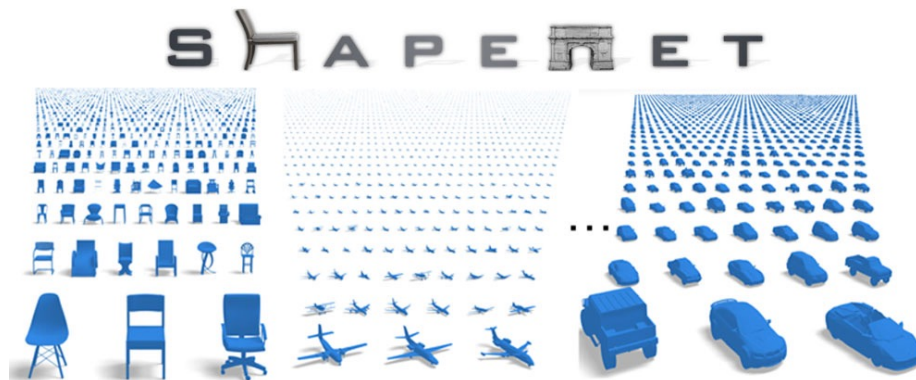
- Addressing the various **3D data challenges**:
acquisition, modeling, interaction, etc.



Graphics is responsible for

- Producing BIG 3D data

ScanNET



SUNCG dataset

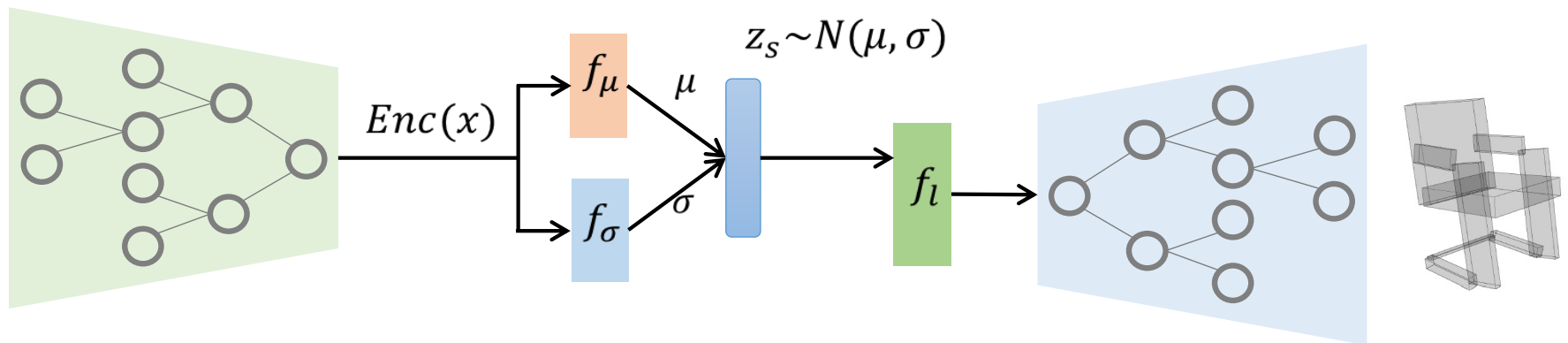
Depth

Physical based rendering

Volumetric ground truth

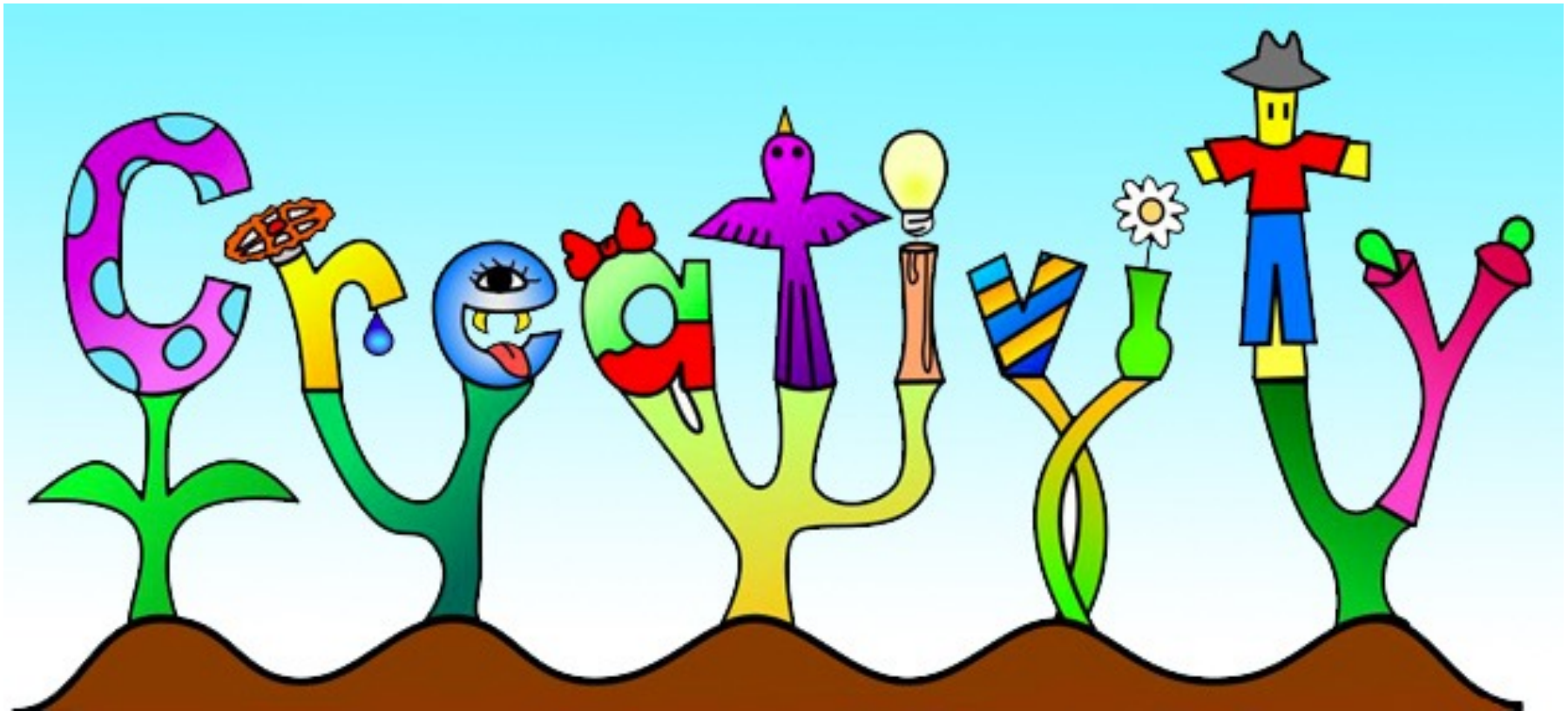
Graphics is responsible for

- Training machines and neural networks capable of generating (novel) 3D content from abstract, implicit descriptions, sets of examples, etc.



Graphics is responsible for

- Realizing and enhancing human imagination and



New graphics: synthesis challenges

- No explicit model description
- Synthesize **novel** 3D content
- Synthesize **Big** 3D Data
 - 4 V's: **V**olume + **V**ariation + **V**ariety + No**V**elty
- Synthesize **creative** 3D content

We are only scratching the surface!