Modeling Paradigms for 3D Content Creation

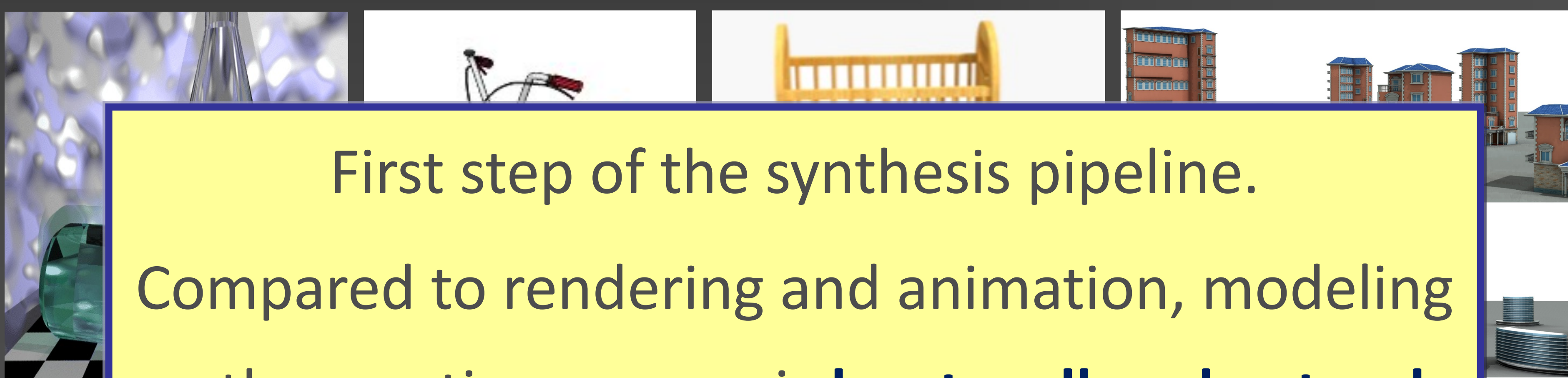
Richard (Hao) Zhang

CMPT 464/764: Geometric Modeling in Computer Graphics

Lecture 2

Recall …

- Computer graphics = synthesis of all visual content
- Today, modeling emphasizes creation, not reconstruction

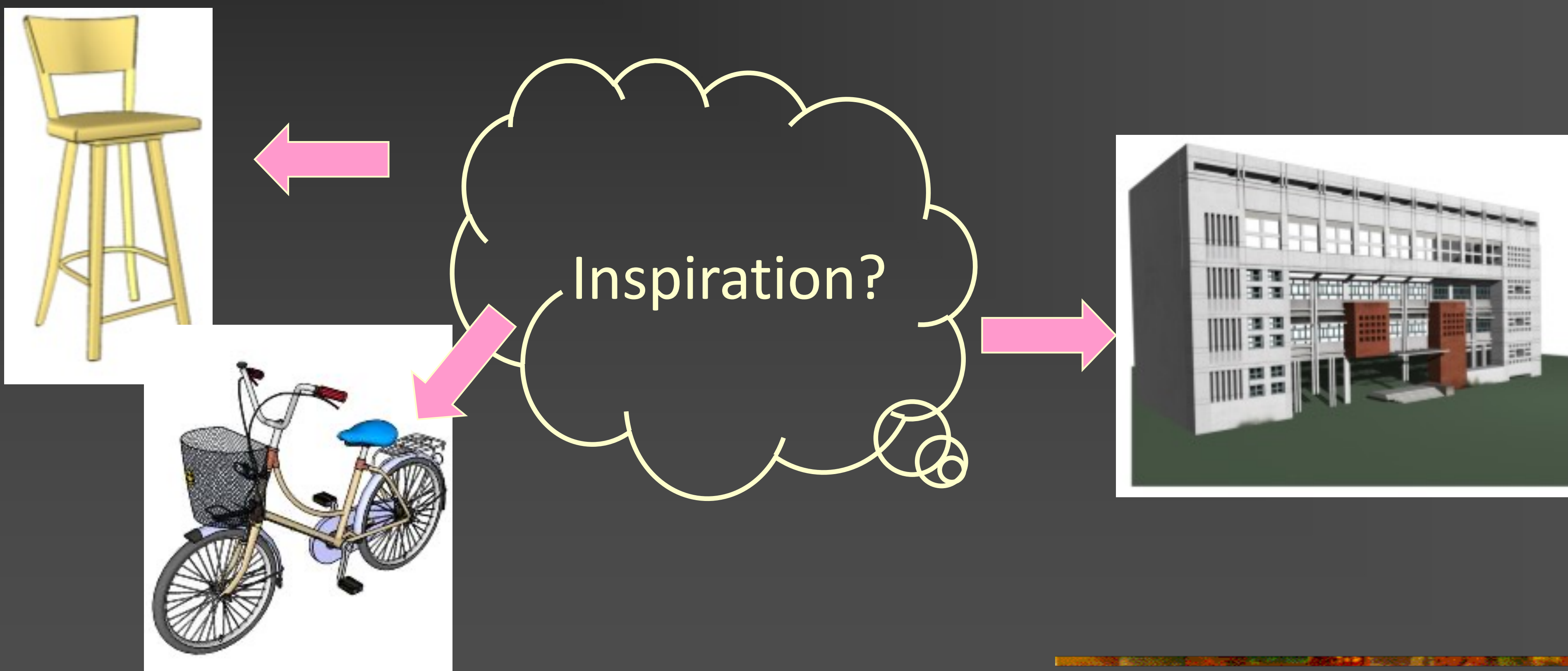


First step of the synthesis pipeline.

Compared to rendering and animation, modeling or the creative process is **least well understood.**

3D Content Creation

Inspiration ⇒ a **readily usable** digital 3D model



Realistic 3D reconstruction

- Inspiration = real-world data, e.g., a laser point scan

[Nan et al., SIG 2010]

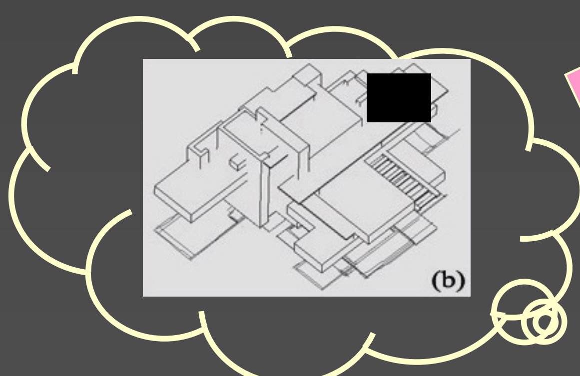
Novel 3D content creation

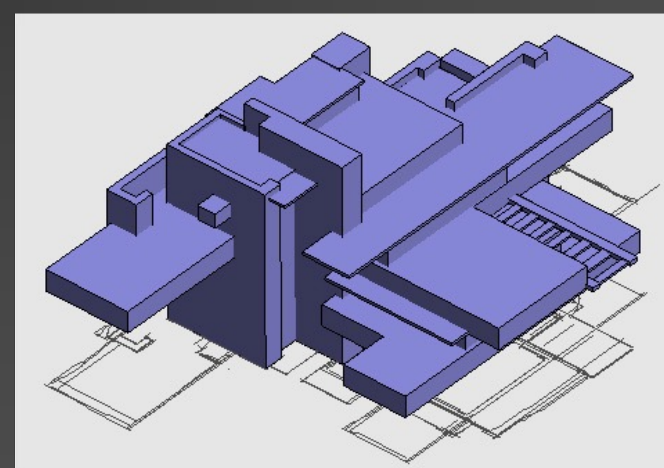
- Creation of novel 3D shapes, not from real scans/images

Novel 3D content creation

- Creation of novel 3D shapes: not from real scans/images
- Inspiration: text description, design concept (e.g., sketch) or constraint (e.g., compactness), usage scenario, existing model(s), or a picture, …

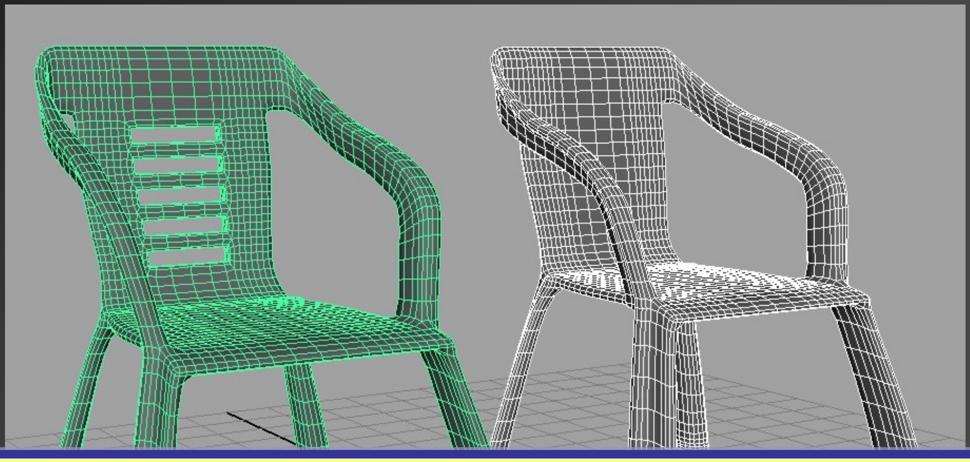
sketch





[Yu and Zhang 2007]

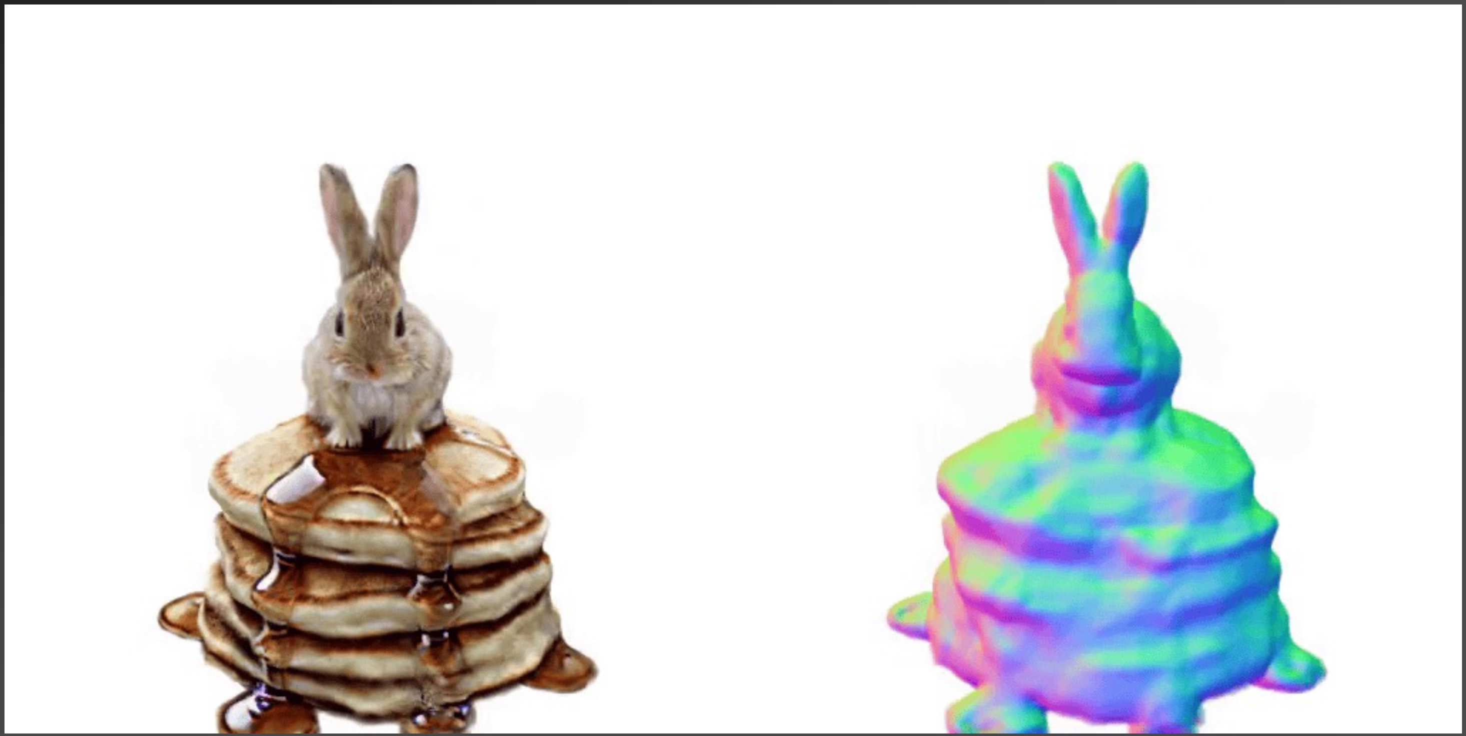
3D content creation is generally hard



One of the most central problems in graphics

Jim Kajiya's award talk: main reason why graphics is not as ubiquitous as we would have liked it to be.

How good is state of the art in 2024?



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

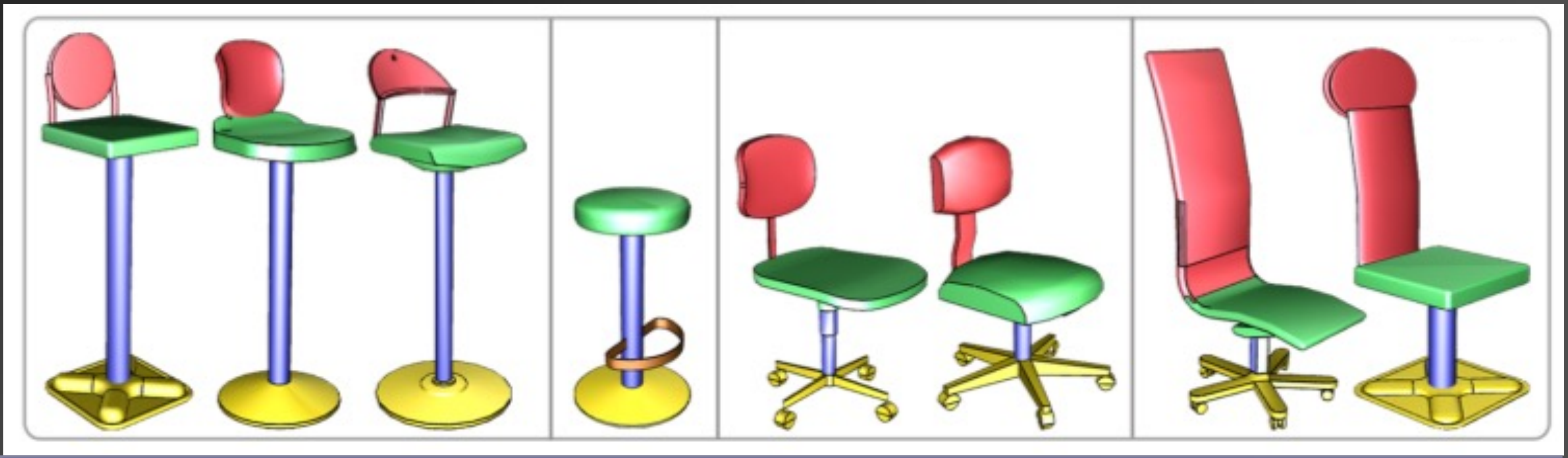
(Re)usable 3DCC is even harder

- Models created are meant for subsequent use
- Creation of readily (re)usable 3D models, with useful info beyond low-level primitives, e.g., points, triangles, etc.

(Re)usable 3DCC is even harder

- Models created are meant for subsequent use
- Creation of readily (re)usable 3D models, with useful info beyond low-level primitives, e.g., points, triangles, etc.
- Higher-level information is more meaningful and reusable

(Re)usable 3DCC is even harder



High-level information to extract or **learn** — analysis problems, which we will study in this course

Key: data reuse

- Exploit existing, possibly pre-analyzed, models → a data-driven approach
- New models = variations of existing 3D models
- Existing models can be inspiration to start with

Modeling paradigms

- Human-in-the-loop editing
  - User interaction plays a central role
  - Various editing paradigms
  - Can start from scratch or from existing 3D models

Modeling paradigms

- Human-in-the-loop editing
  - User interaction plays a central role
  - Various editing paradigms
  - Can start from scratch or from existing 3D models
- Semi-automatic to automatic synthesis
  - Classical: data-driven variation of existing 3D models
  - User can provide sparse/abstract constraints
  - Modern: generative modeling from scratch, even noise

Key considerations

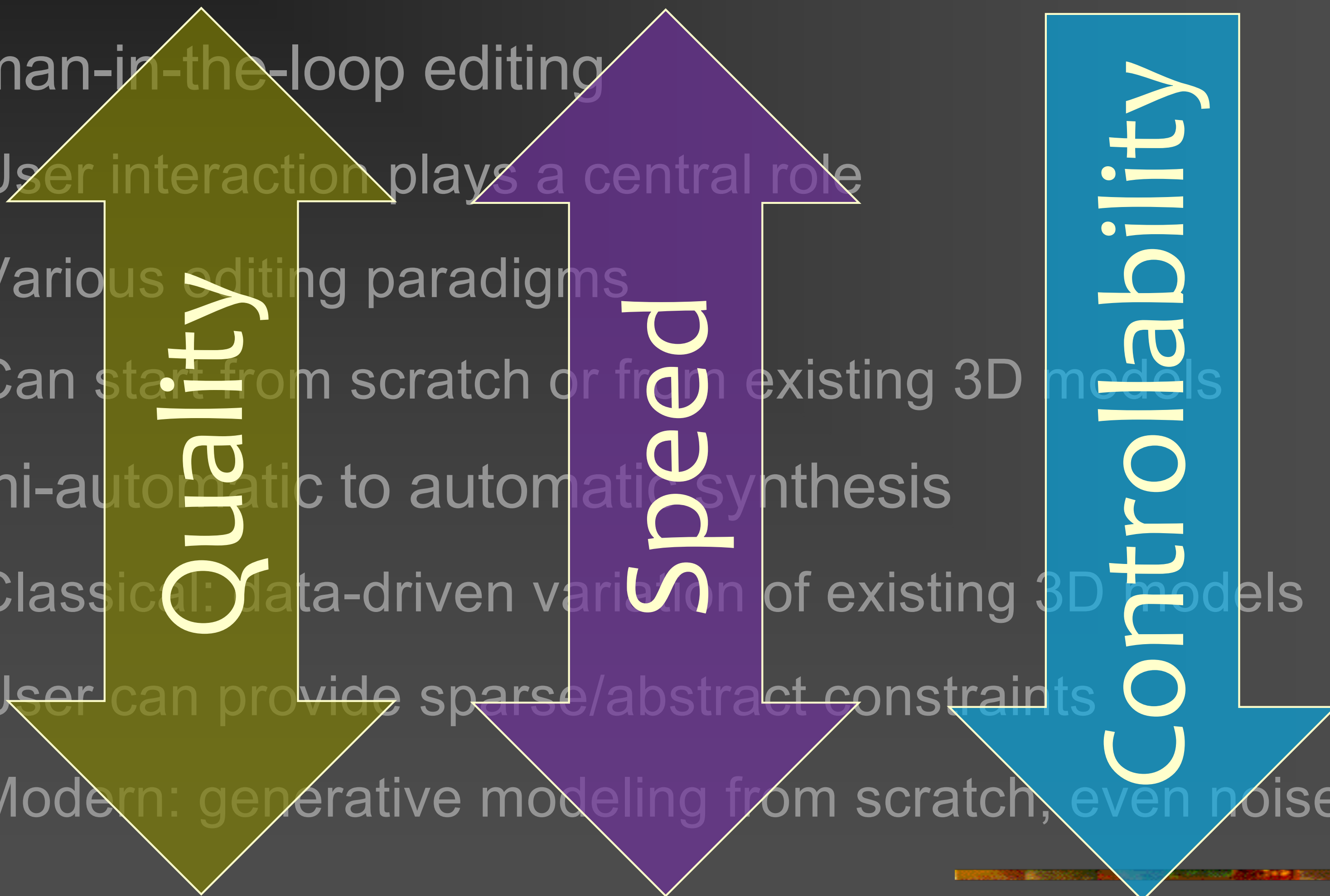
- Quality
- Speed
- Controllability

Modeling paradigms

- Human-in-the-loop editing
  - User interaction plays a central role
  - Various editing paradigms
  - Can start from scratch or from existing 3D models
- Semi-automatic to automatic synthesis
  - Classical: data-driven variation of existing 3D models
  - User can provide sparse/abstract constraints
  - Modern: generative modeling from scratch, even noise

Modeling paradigms

- Human-in-the-loop editing
  - User interaction plays a central role
  - Various editing paradigms
  - Can start from scratch or from existing 3D models
- Semi-automatic to automatic synthesis
  - Classical: data-driven variation of existing 3D models
  - User can provide sparse/abstract constraints
  - Modern: generative modeling from scratch, even noise

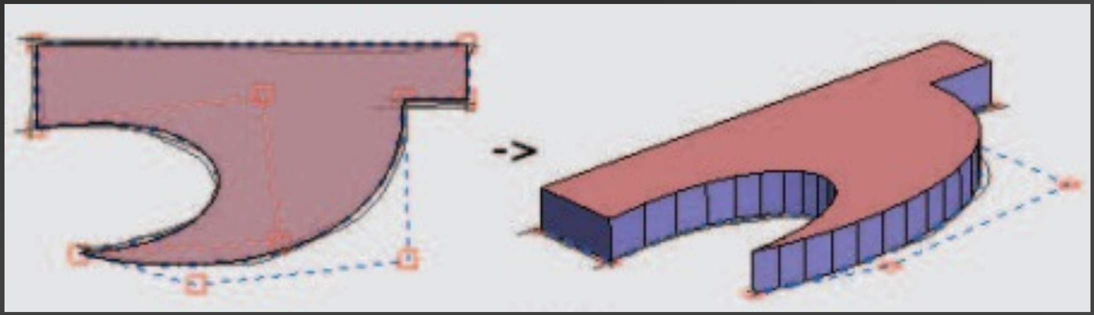


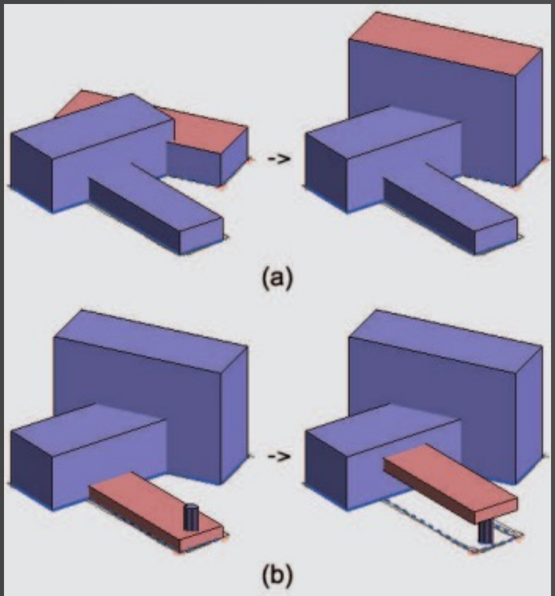
Editing

- From scratch: 2D to 3D
  - From engineering drawings
  - From freeform sketches to 3D models and for editing
- Fundamental 3D editing operators
  - Sweeping
  - Constructive solid geometry (CSG)
  - Free-form deformation: various editing handles
  - Part composition

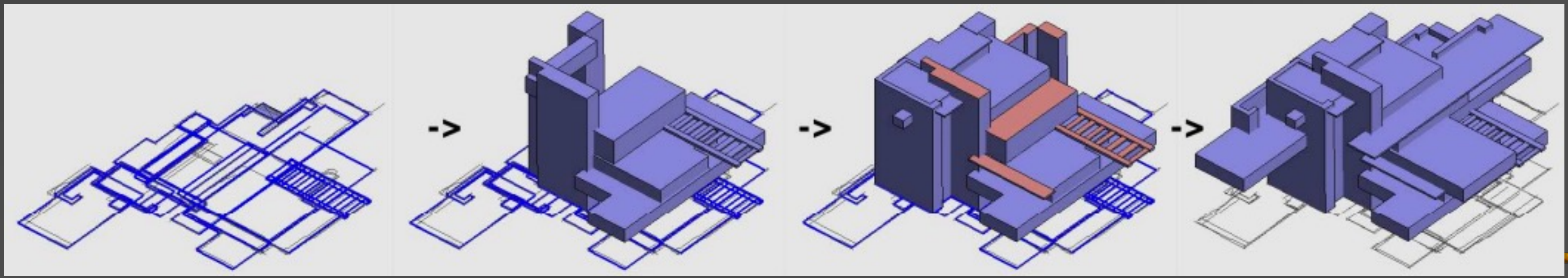
Modeling from engineering drawings

- Most fundamental operation: extrusion





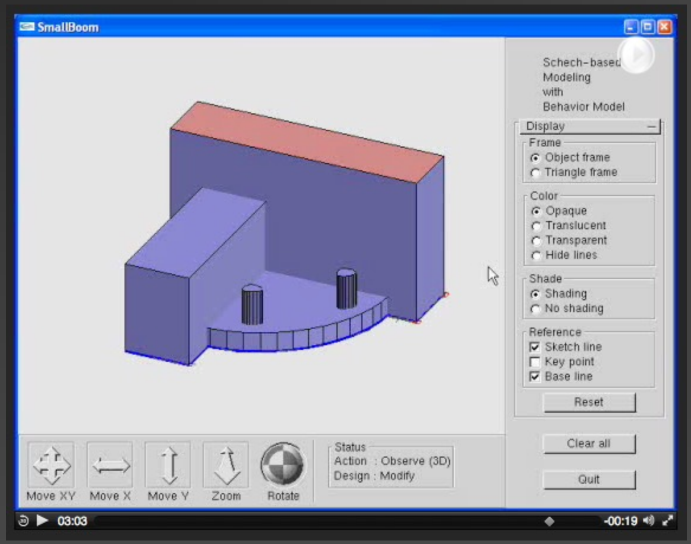
- Additional operations: e.g., lifting, etc.



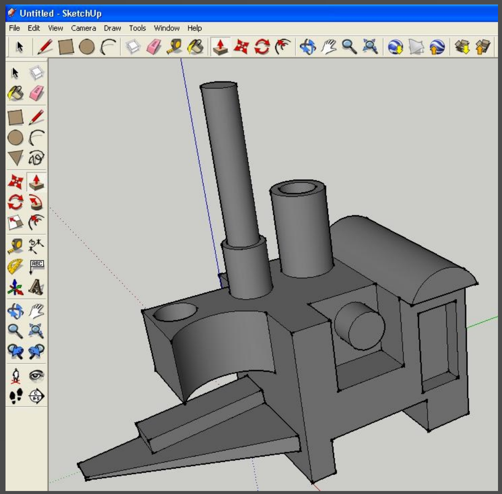
[Yu and Zhang 2007]

Modeling from engineering drawings

- Major application domain: architectural design/modeling



- Well-known example: (Google) SketchUp



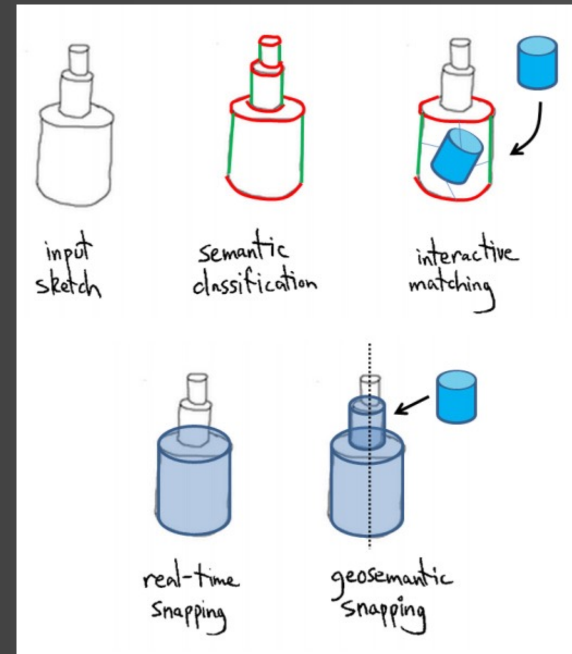
Modeling from freeform sketches

- 2D sketches most natural for editing

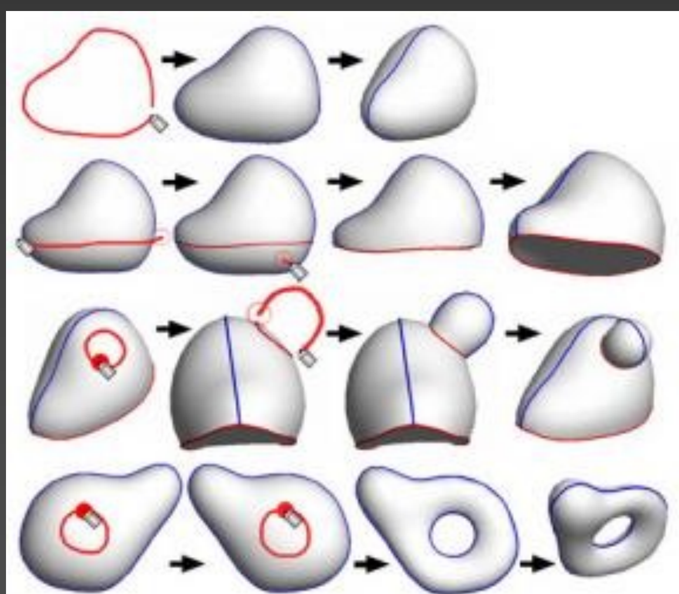
Early work: Teddy by [Igarashi SIG 1999]

Sketch-based modeling

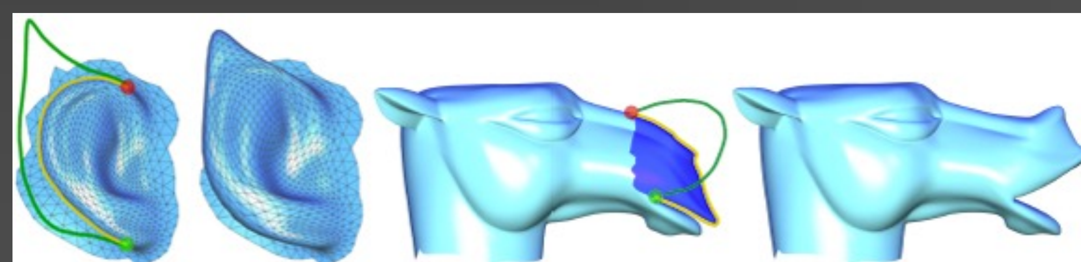
- 2D sketches most natural for editing
- Aside from extrusion



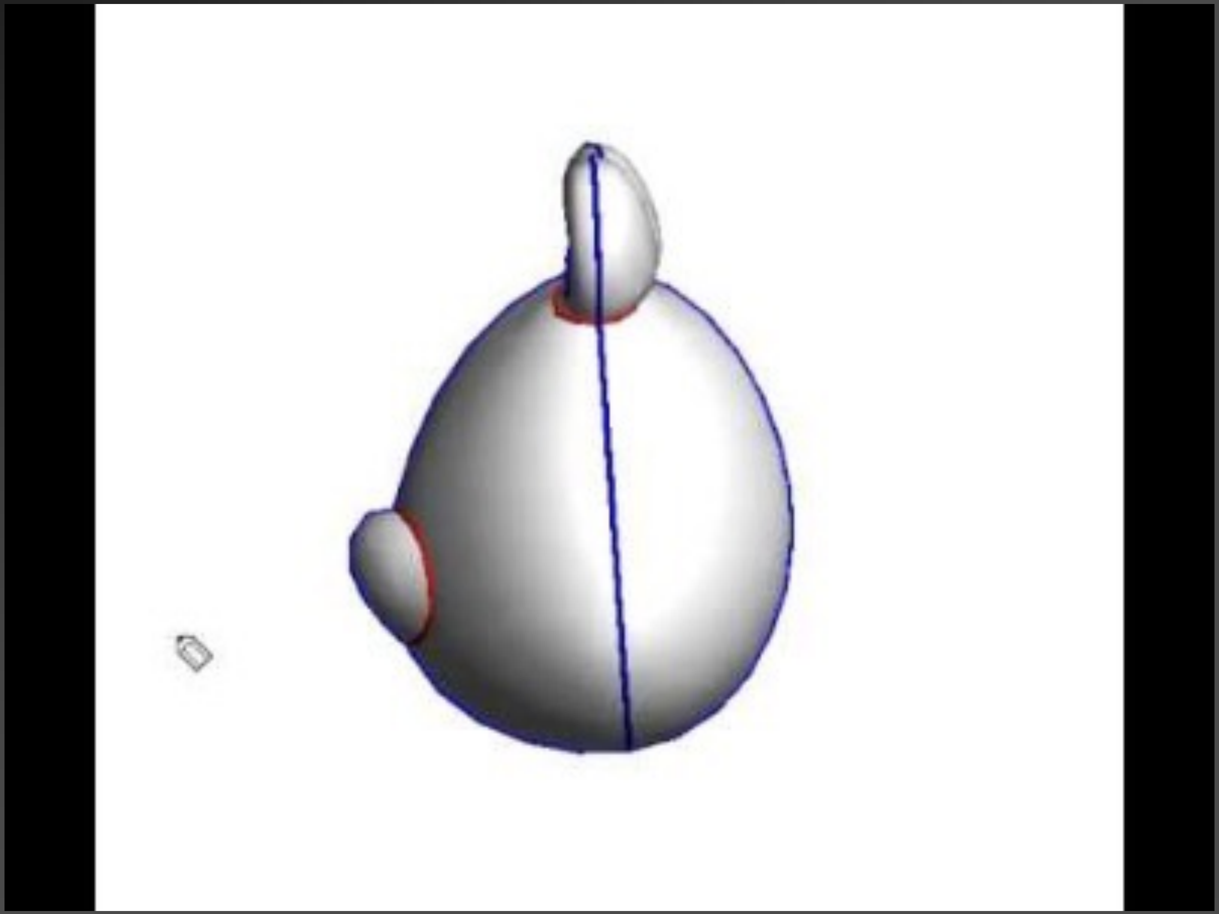
Early work: Teddy by [Igarashi SIG 1999]



[Nealen et al. 2007 & 2005]

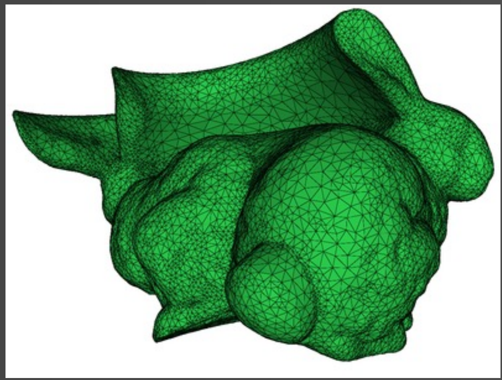


FiberMesh video



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

Sweeping

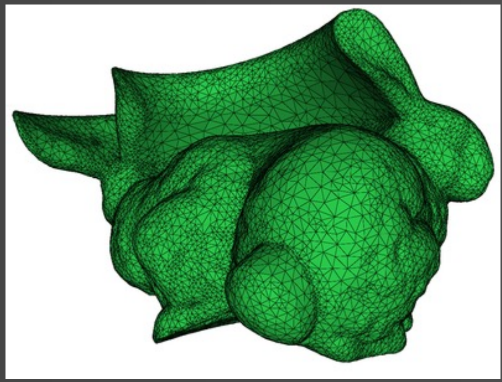


- Creates a swept volume: a means for solid modeling
- Simplest case: sweep a 2D cross-section along a curve
  - Create generalized cylinders
  - Great success: 3-sweep: 2.3M views on Youtube!

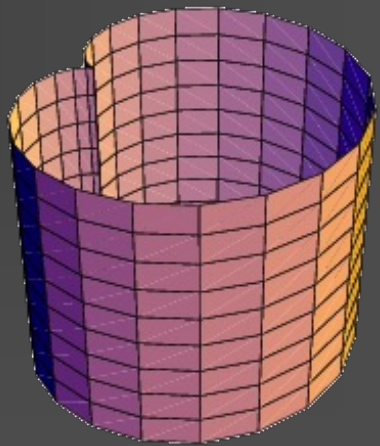
3-sweep video

https://www.youtube.com/watch?v=Oie1ZXWceqM

Sweeping



- Creates a swept volume: a means for solid modeling
- Simplest case: sweep a 2D cross-section along a curve
  - Create generalized cylinders
  - Great success: 3-sweep; 2.3M views on Youtube
- Tensor-product surfaces (next lecture)
  - Sweep a curve and each point on curve goes through a (different) curve
- More generally: sweep arbitrary shapes



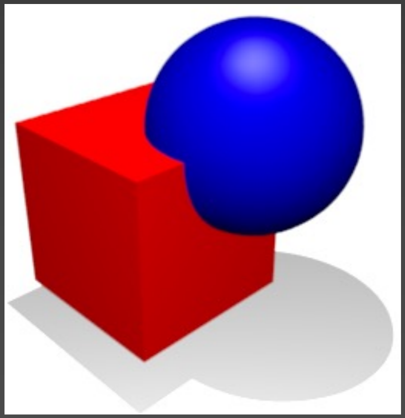
SweepNet: Unsupervised Learning Shape Abstraction via Neural Sweepers

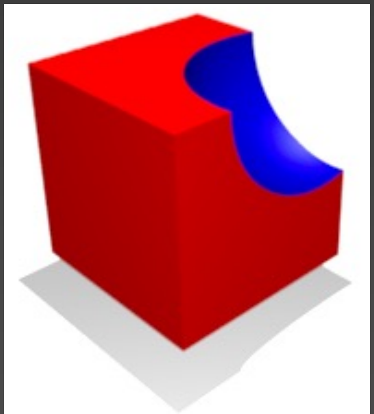
ECCV 2024

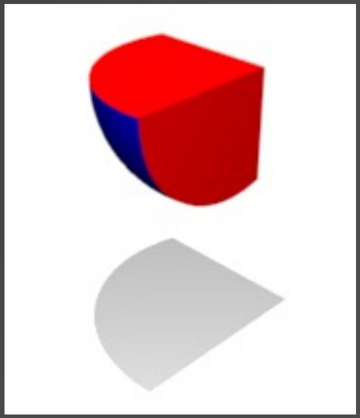
Mingrui Zhao[1], Yizhi Wang[1], Fenggen Yu[1], Changqing Zou[2], Ali Mahdavi-Amiri[1],
[1]Simon Fraser University [2]Zhejiang University

Constructive solid geometry (CSG)

- Built on a set of solid primitives, e.g., cubes, spheres, cylinders, and more general parametric models
- Construction via recursive Boolean operations


Union

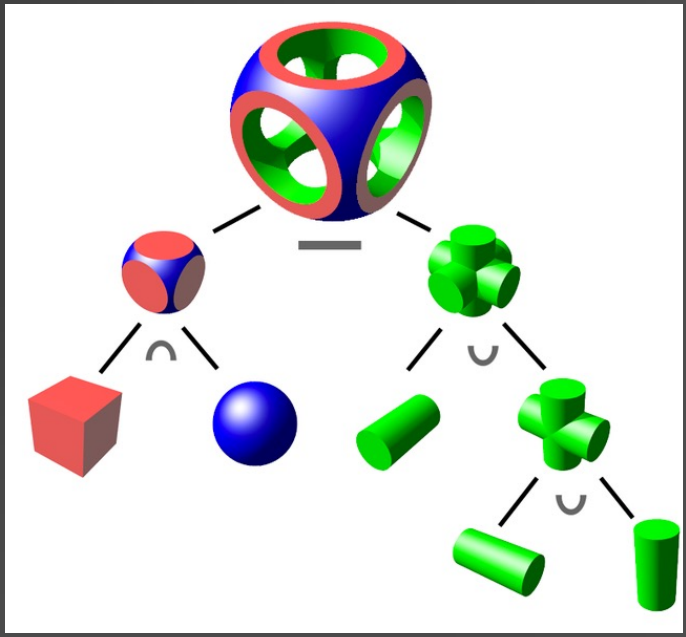

Difference


Intersection

Images taken from Wikipedia

CSG: fundamental shape representation

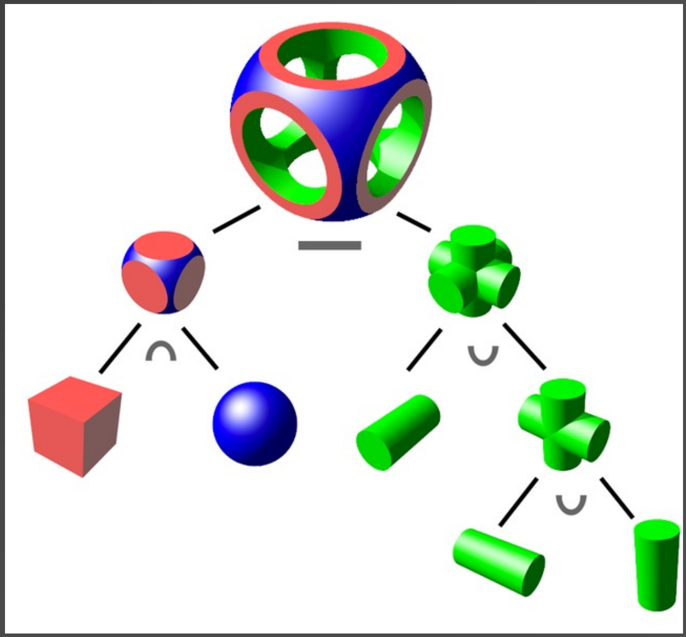
- Recursion results in a tree representation: CSG tree
- Interesting question: inverse CSG
  - How to recover the tree?



Images taken from Wikipedia

CSG: fundamental shape representation

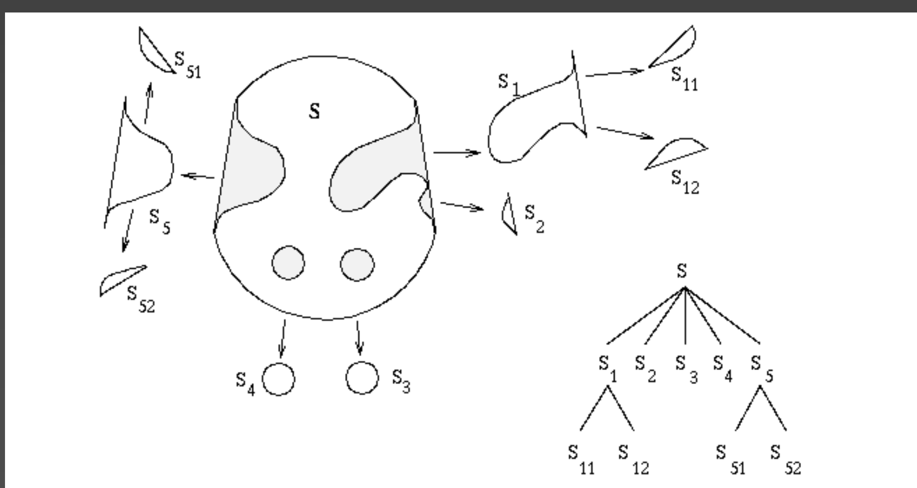
- Recursion results in a tree representation: CSG tree
- Interesting question: inverse CSG
  - How to recover the tree?
  - Use of deep learning in 2020

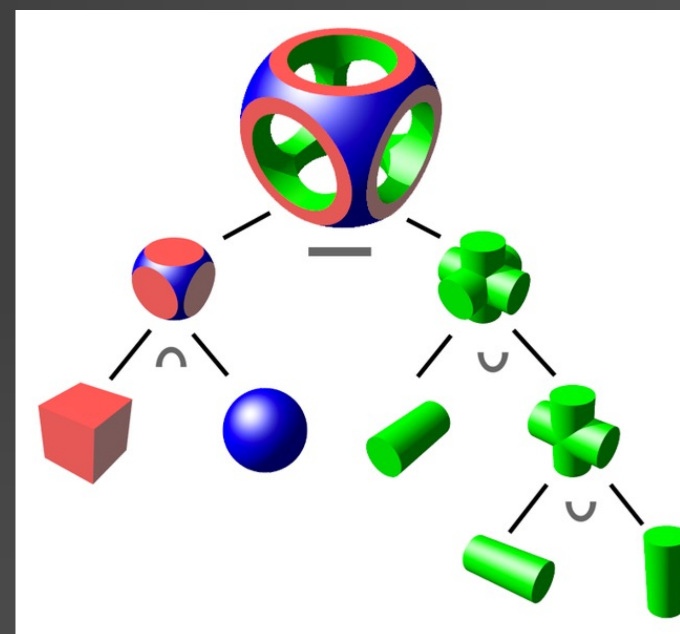


Images taken from Wikipedia

CSG: fundamental shape representation

- Recursion results in a tree representation: CSG tree
- Interesting question: inverse CSG
  - How to recover the tree?
- Related: concavity trees



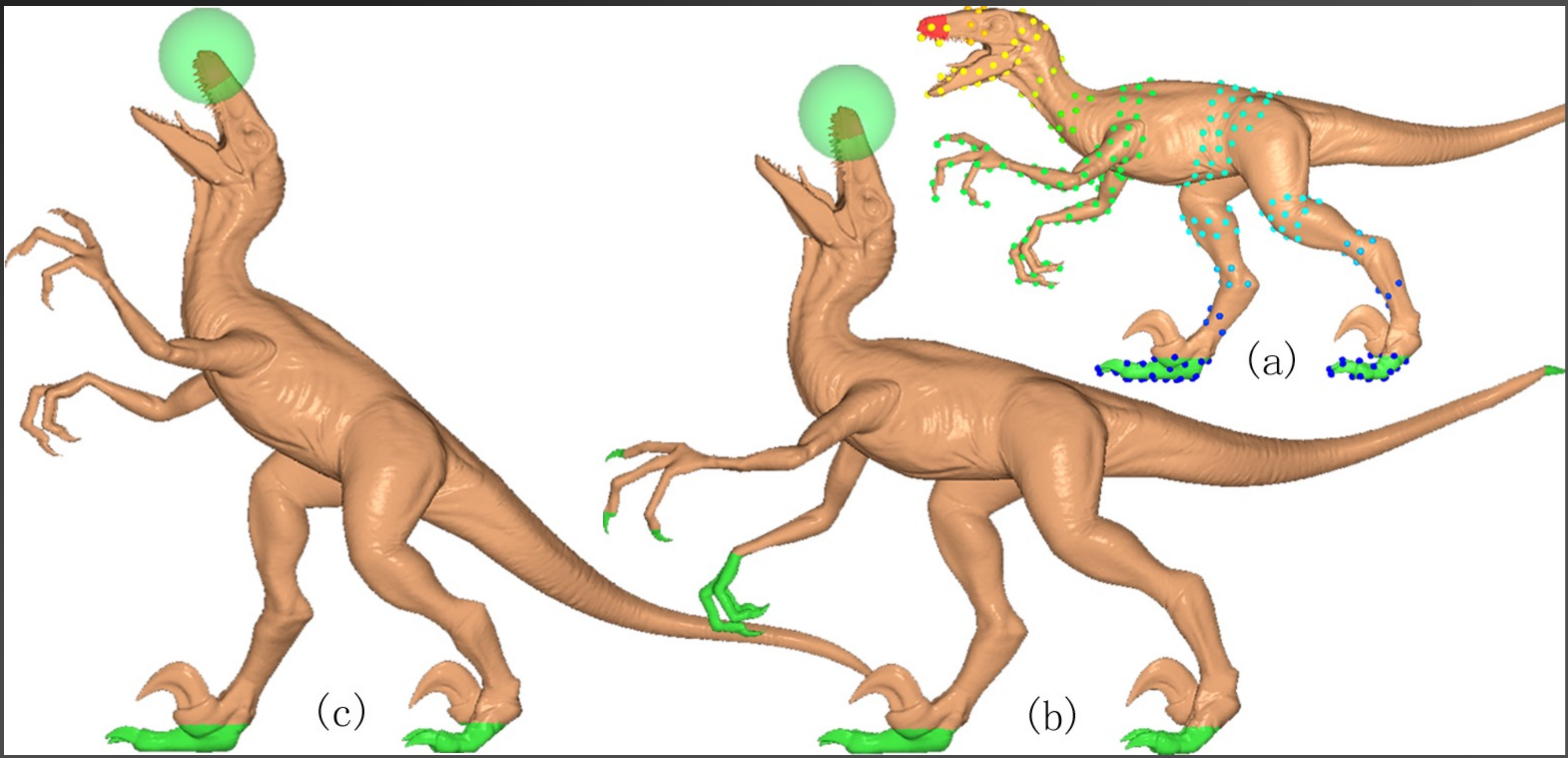


Images taken from Wikipedia

Free-form deformation

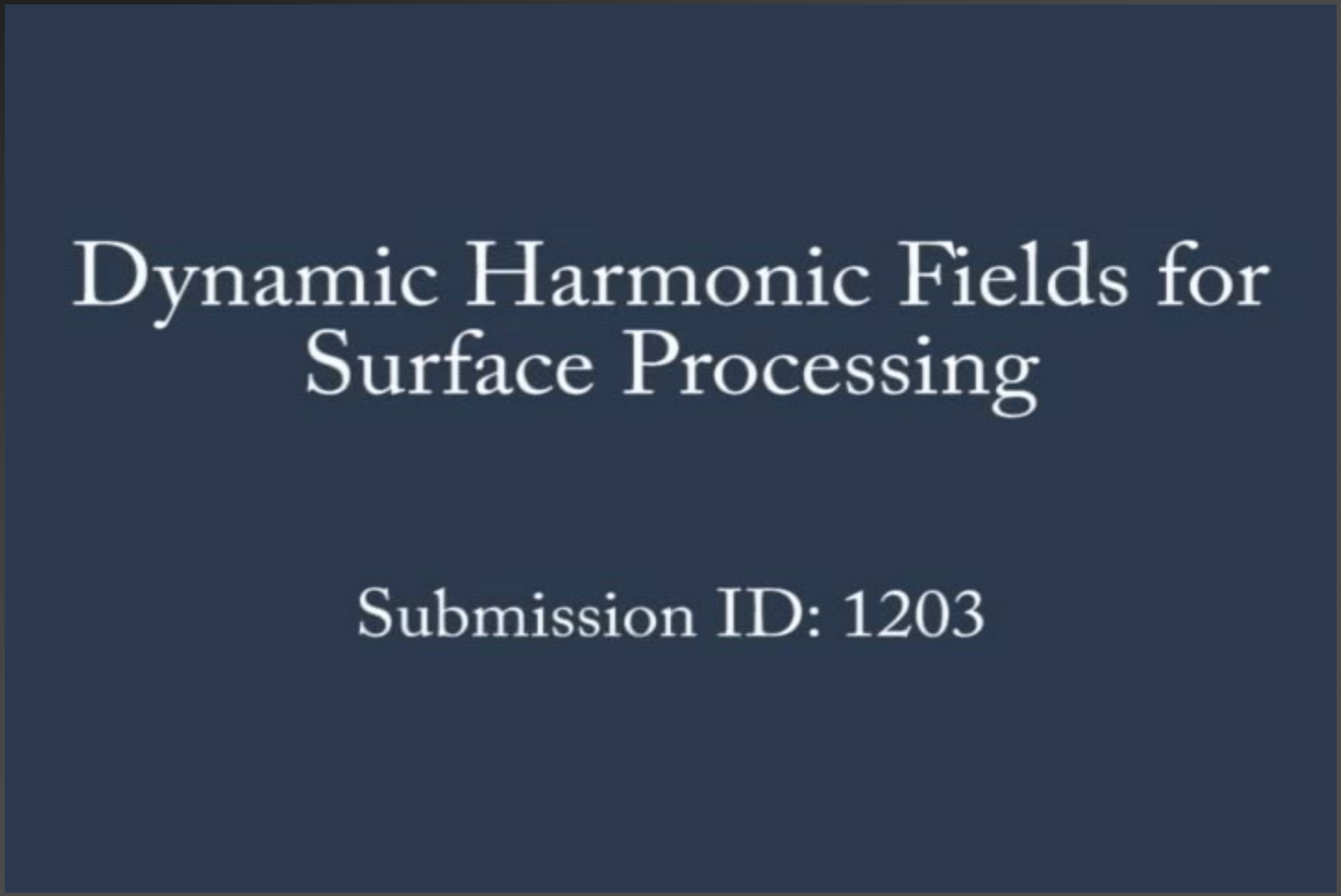
- What are the deformation handles?
  - Anchor points
  - Skeletons
  - Wires
  - Cages
- What needs to be preserved away from handles?
  - Surface details
  - Shape structures, e.g., symmetry

Mesh deformation by anchor points



K. Xu, H. Zhang, and D. Cohen-Or, "Dynamic Harmonic Fields for Shape Processing", *IEEE Shape Modeling International 2009.*

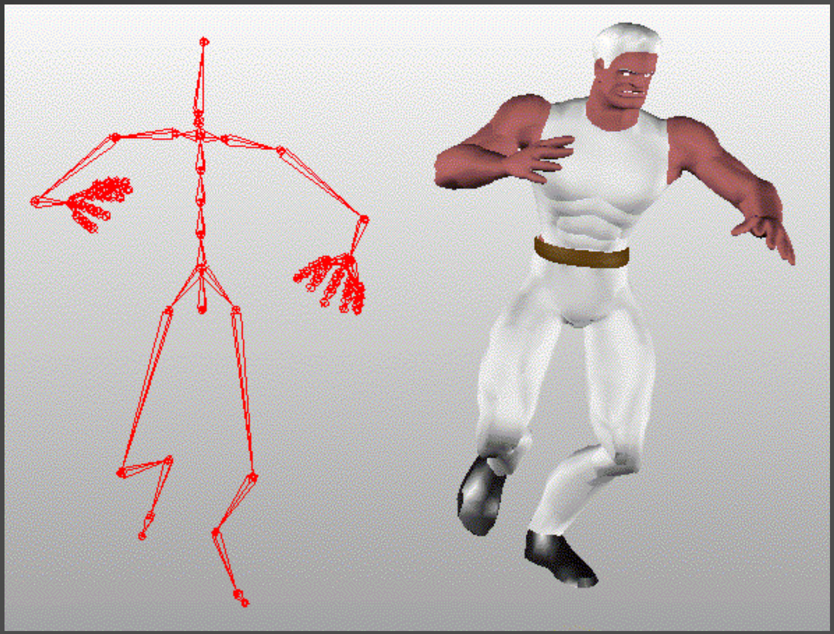
Video demo [2:04:00 -]



K. Xu, H. Zhang, and D. Cohen-Or, “Dynamic Harmonic Fields for Shape Processing”, *IEEE Shape Modeling International 2009.*

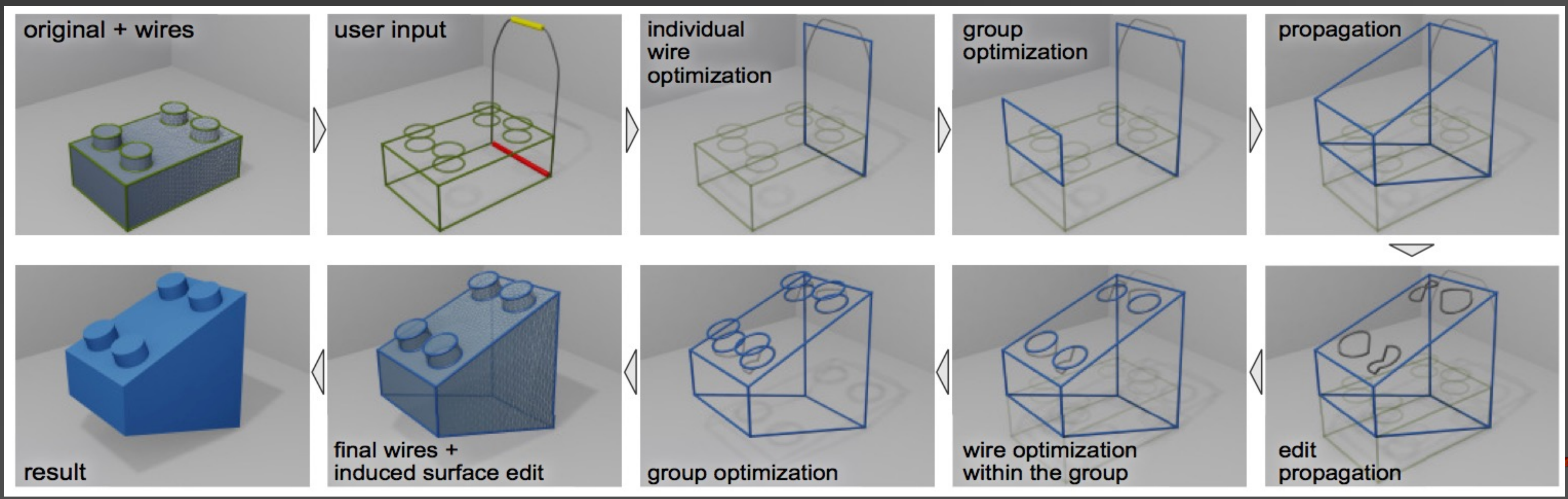
Skeletons and mesh skinning

- Shape characterized by a set of bones (the skeleton)
- The boundary surface acts as "skins"
- Boundary vertices specified as weighted combinations of bone vertices
- Editing done on the bones and the skins will follow



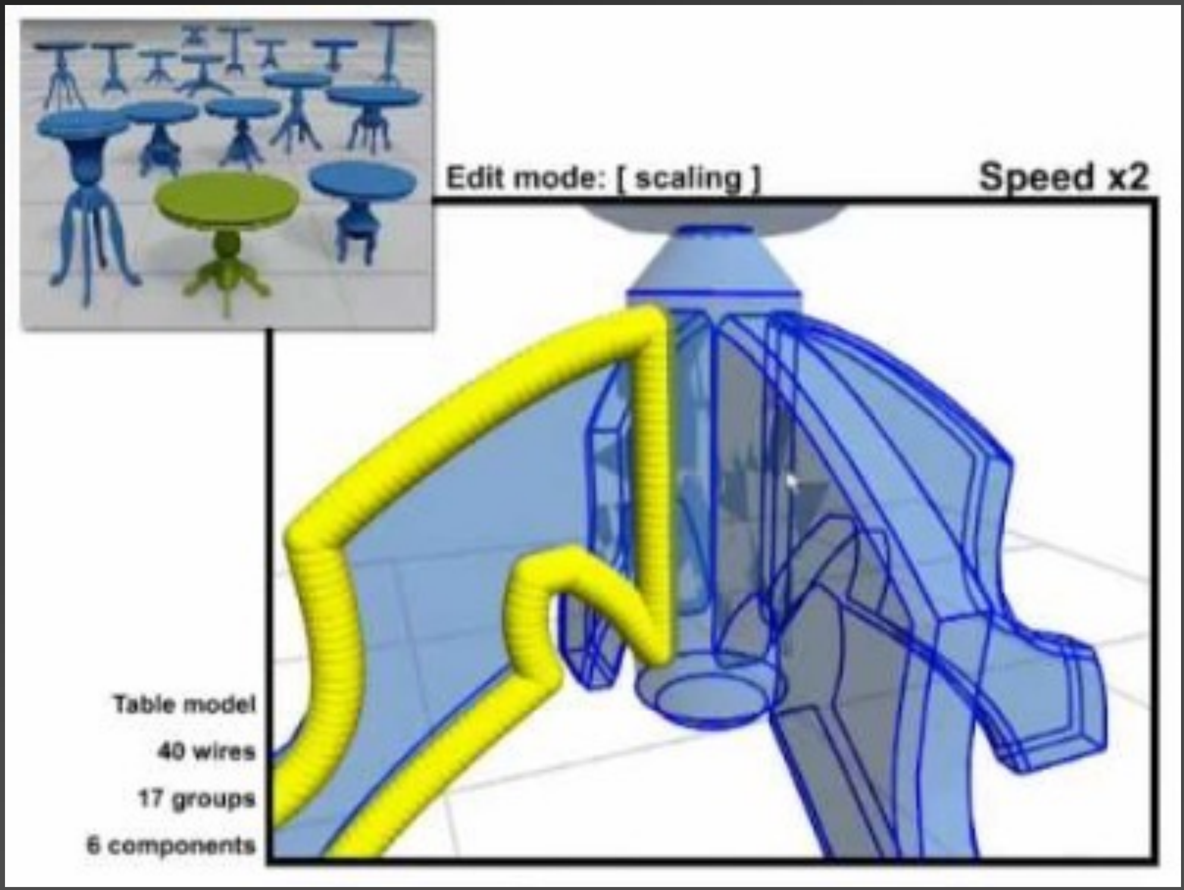
iWires: analyze-and-edit

- Wires as control handles [Singh & Fiume 1999]
- Edits must preserve structural relations among wires
  - E.g., symmetry, co-planarity, perpendicularity, etc.



[Gal et al. 2009]

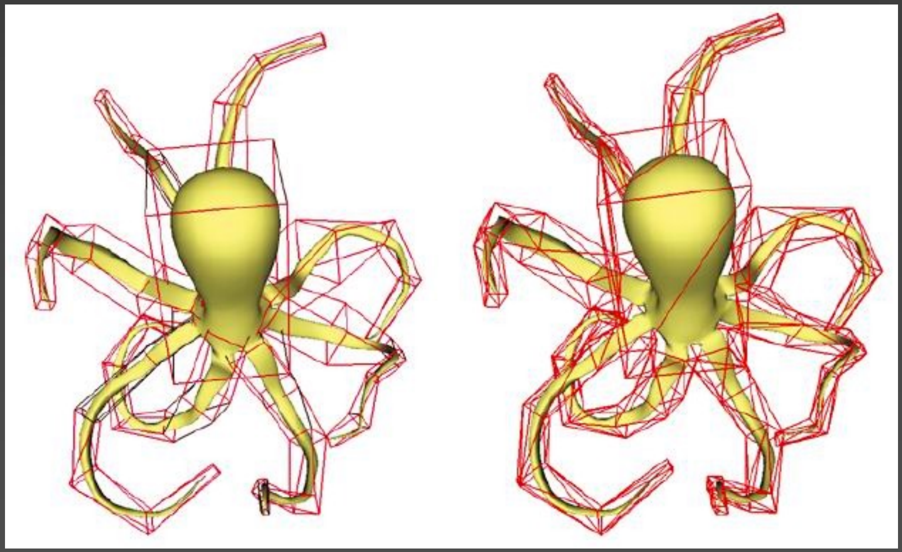
iWires video



[Gal et al. 2009]

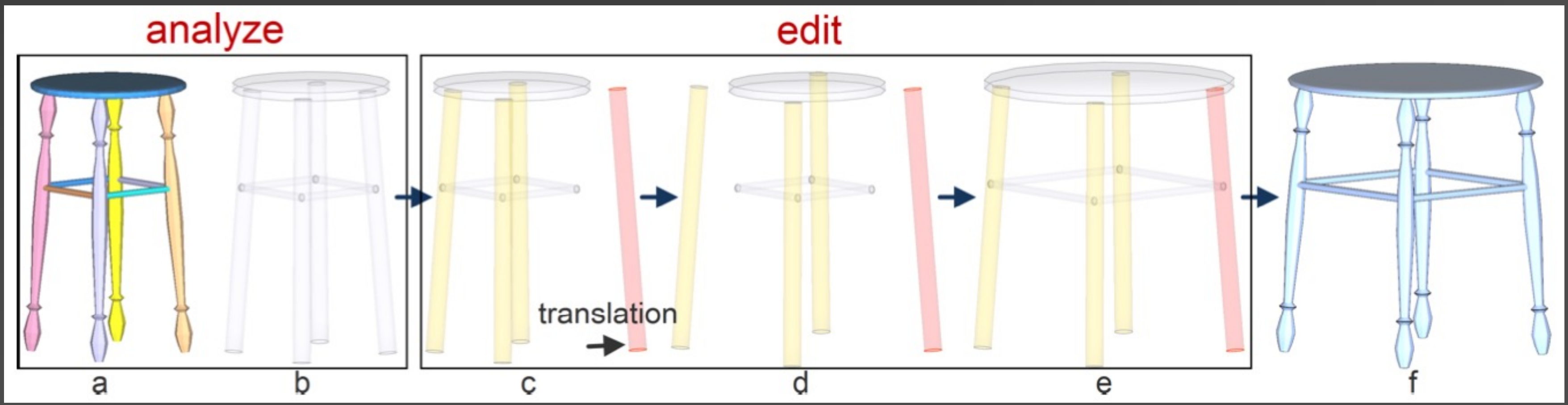
Cage-based mesh deformation

- Cages are simple enclosing primitives, e.g., cuboids
- Edits done on cages with shape properties preserved
  - E.g., coordinates of interior points as weights on cage vertices



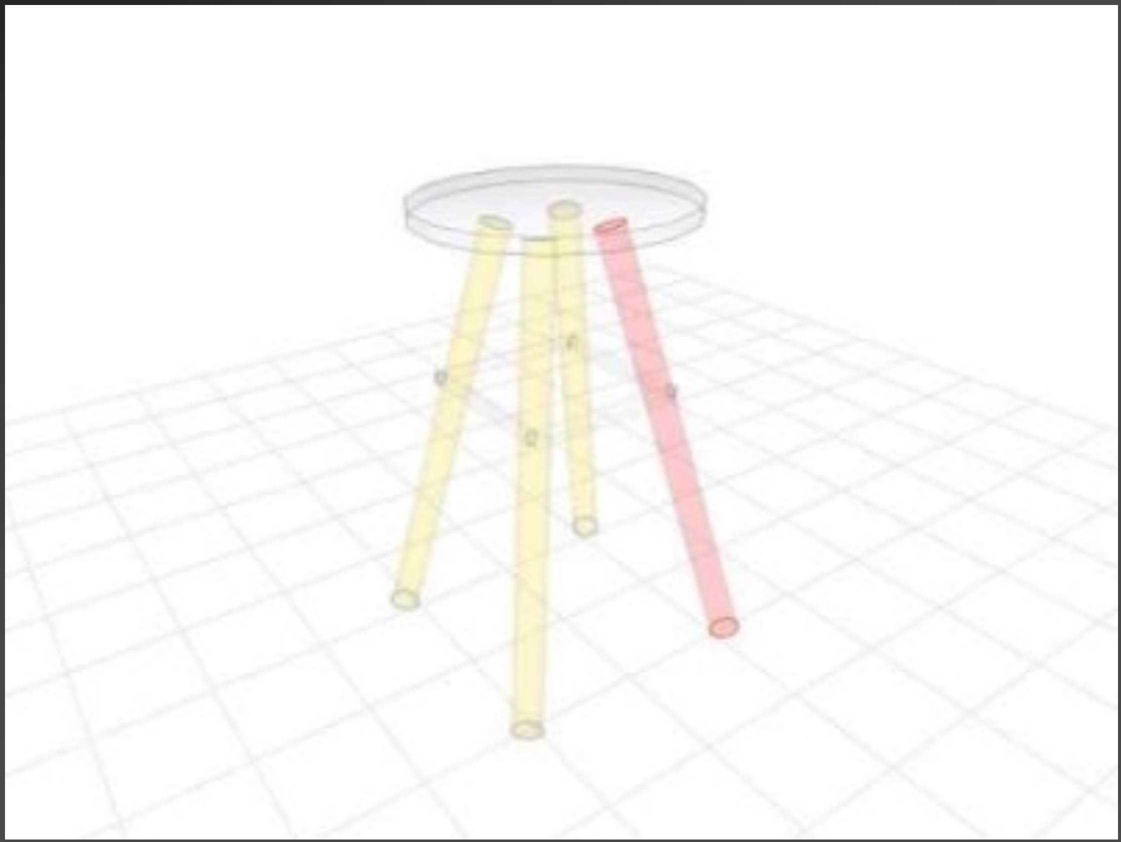
Component-wise controllers

- Cuboids and generalized cylinders as control handles
- These simple primitives bound shape parts, like cages
- Edits preserve structures among controllers, like iWire



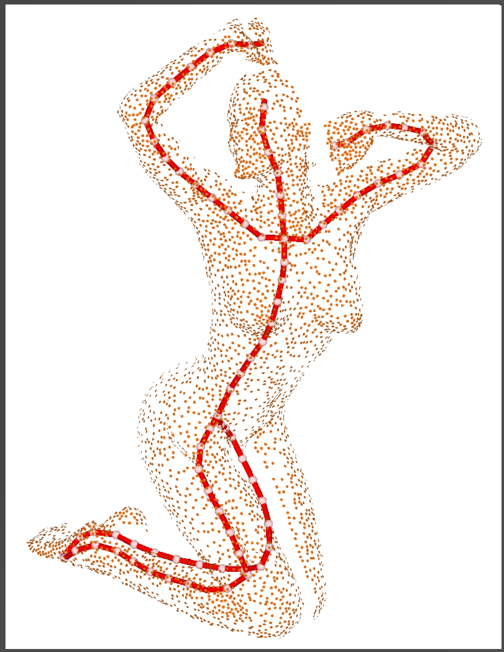
[Zheng et al. 2010]

Component-wise controller video



Commonalities

- Handles are all simple primitives
  - Simple to specify and manipulate
  - But provide good shape abstraction
- Getting good abstractions non-trivial
  - E.g., curve skeleton or cage extraction
- Important to derive properties to preserve
  - Encoding surface details
  - Structure (e.g., symmetry) detection



[Tagliassachi et al. SIG 2009]

Modeling from parts

Modeling via part re-assembly [Funkhouser et al., SIG 2004]

Key: segmentation, shape understanding

back

seat

armrest

legs

Automatic synthesis

- No longer editing of an existing 3D model
- But exploit existing, possibly pre-analyzed, models for automatic or semi-automatic model synthesis
- Mostly a data-driven approach

Modeling as variations

- New models = variations of existing 3D models
- Paradigm I: variation as modification of an existing model
- Paradigm II: via part composition, from multiple models

Photo-inspired modeling

Inspiration: a single photograph [Xu et al., SIG 2011]

From single photo to 3D

- Not editing; modeling inspired by a single photo
- Warp an existing 3D model to fit object silhouette
- Structure preservation ensures a coherent 3D model

Photo-inspired modeling

photo

Photo-inspired modeling

photo

Retrieved
candidate model

Photo-inspired modeling

photo

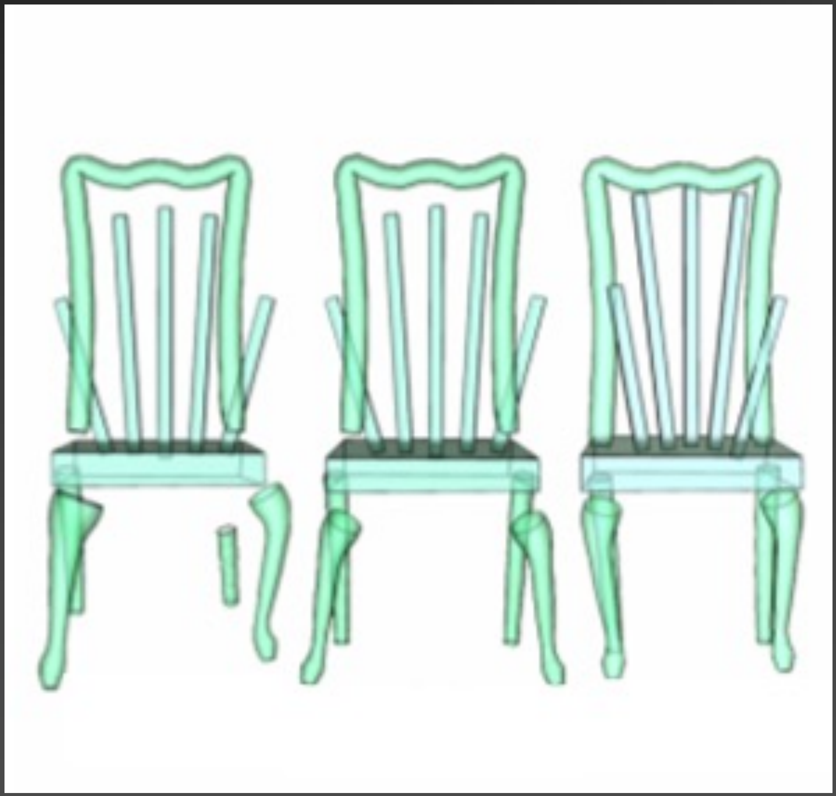
Retrieved candidate model

Result of deformation to fit silhouette

Structure preservation at work

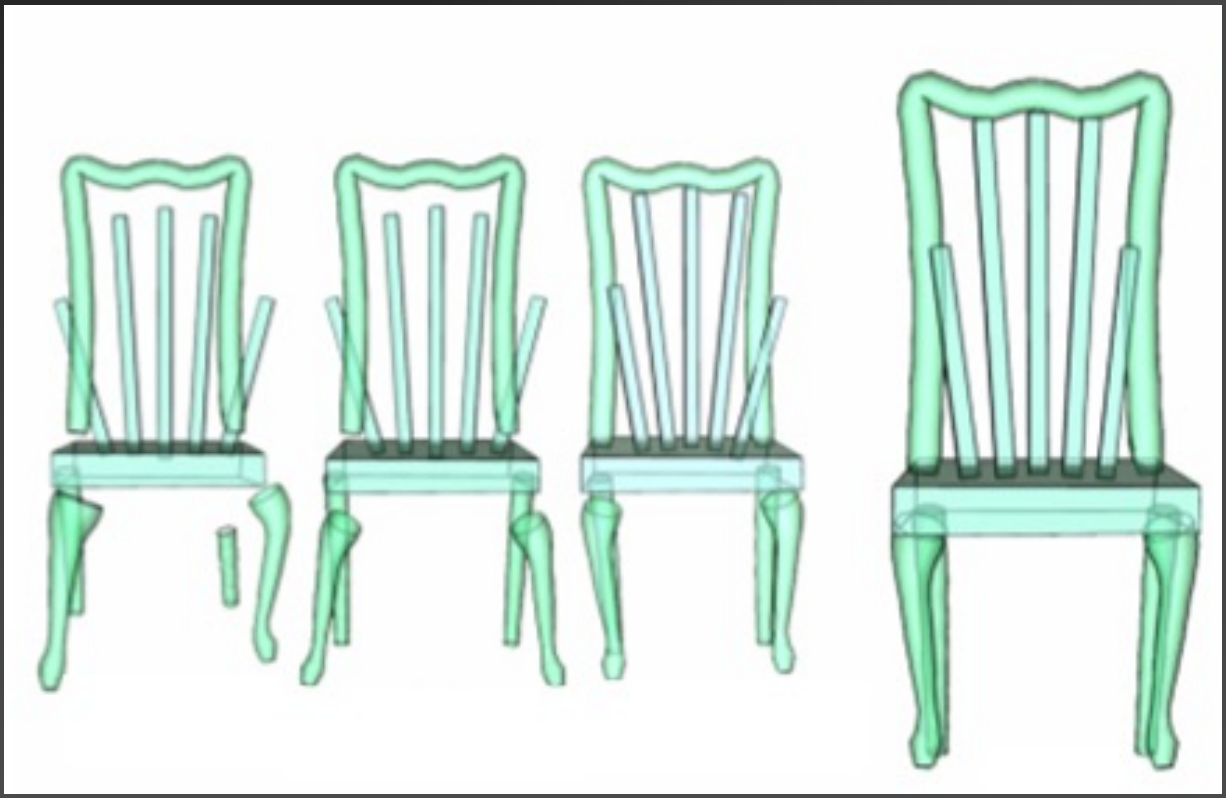
by symmetry

Structure preservation at work



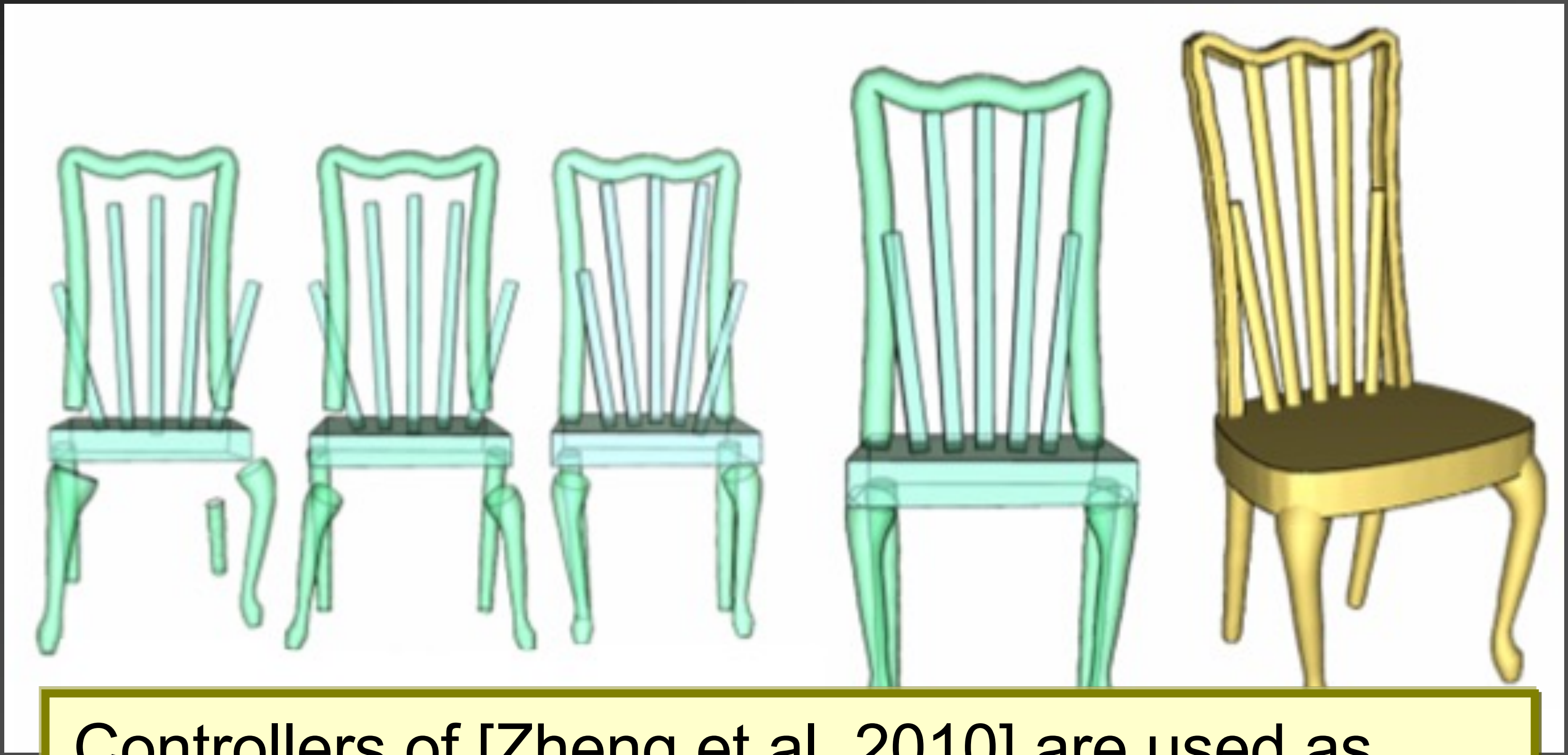
by proximity

Structure preservation at work



additional optimization

Structure preservation at work



Controllers of [Zheng et al. 2010] are used as deformation/warp model

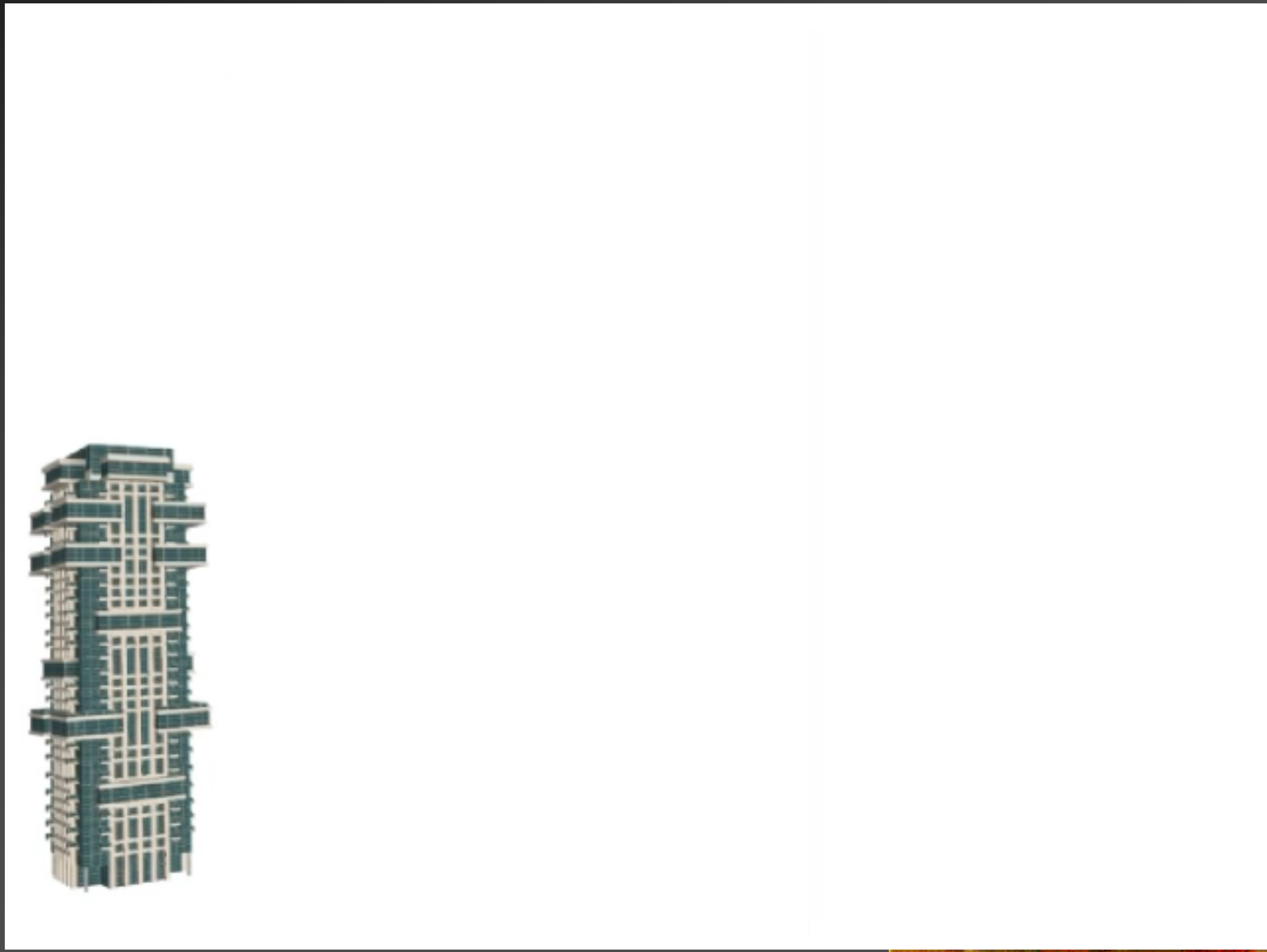
Structure retargeting

Key words: analyze-and-stretch; increases or decrease pattern repetitions instead of geometric stretch

Structure retargeting

Structure-preserving retargeting of irregular 3D architecture
[Lin et al., SIG Asia 2011]

Structure retargeting video



Variation as part composition

- Modeling by example [Funkhouser et al. 2004]
- Fit & diverse [Xu et al. 2012]
- Structure recovery by part assembly [Shen et al. 2012]

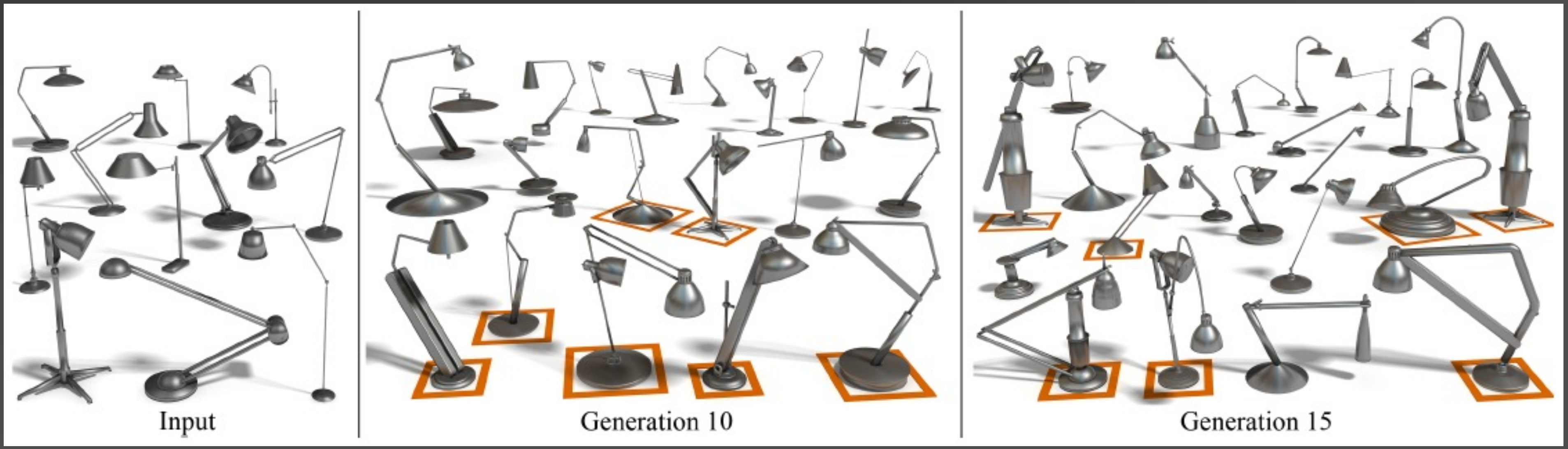
Modeling by example

- New models composed by parts retrieved from an existing database
- Key: retrieve relevant parts by geometric similarity of parts
- Many variants to date

[Funkhouser et al. 2004]

Fit & diverse

- Different from previous works: instead of generating one model at a time, evolve sets of shapes together
- Inspired by the biological process of evolution



[Xu et al. 2012]

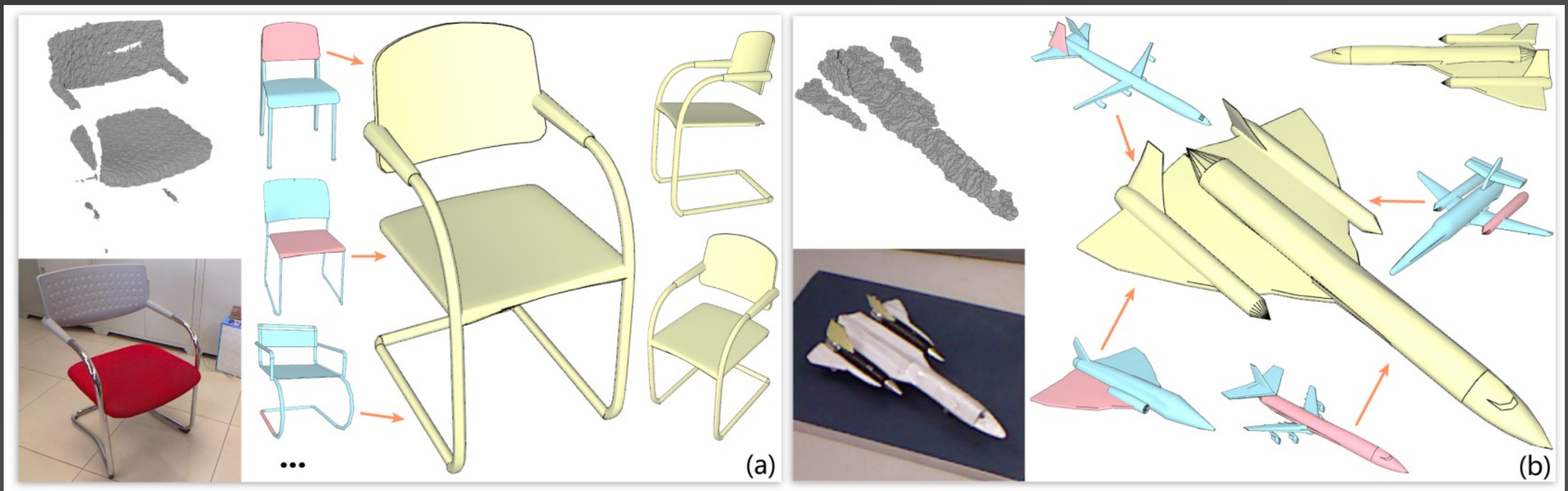
Fit & diverse

- Off-springs by part mutation (warp) and cross-over (part reassembly): leads to diversity in shape creations
- A “design gallery”: user specifies “likes” or “dislikes”; defines fitness function
- Potential for **creativity**!

[Xu et al. 2012]

Structure recovery by part assembly

- Modeling from single Kinect depth scan + RGB image
- Unlike [Xu et al. 2011], model is built by part assembly from multiple shapes, more versatile than just warp



[Shen et al. 2012]

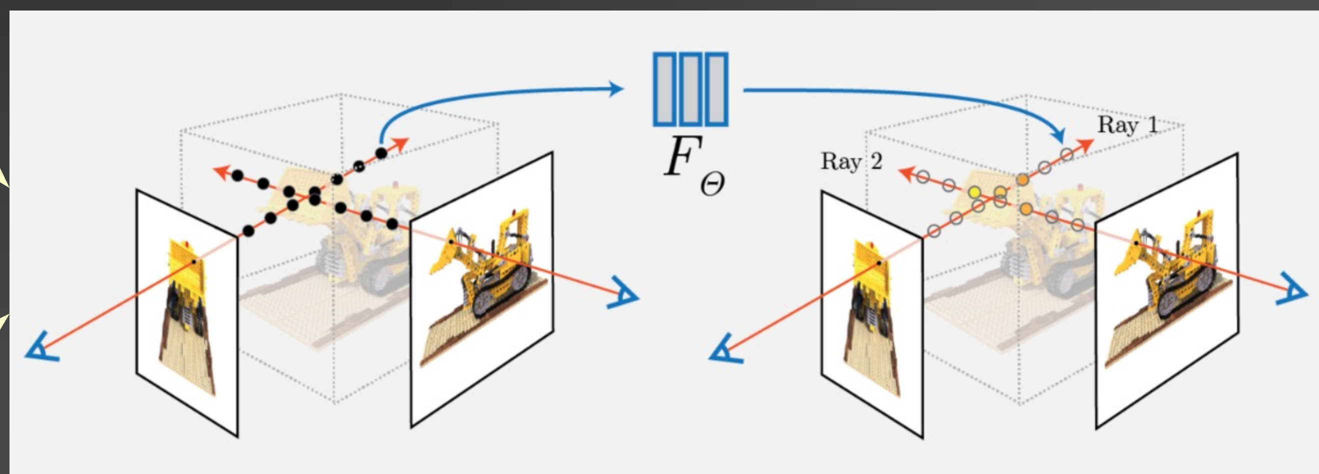
Fast track to 2024: 3D GenAI

- Most popular: 3D Generative AI (GenAI) from scratch
  - – Basically **hallucination** (“dreaming?”) with some level of **conditioning**

Text:
“A Lego front loader”

or

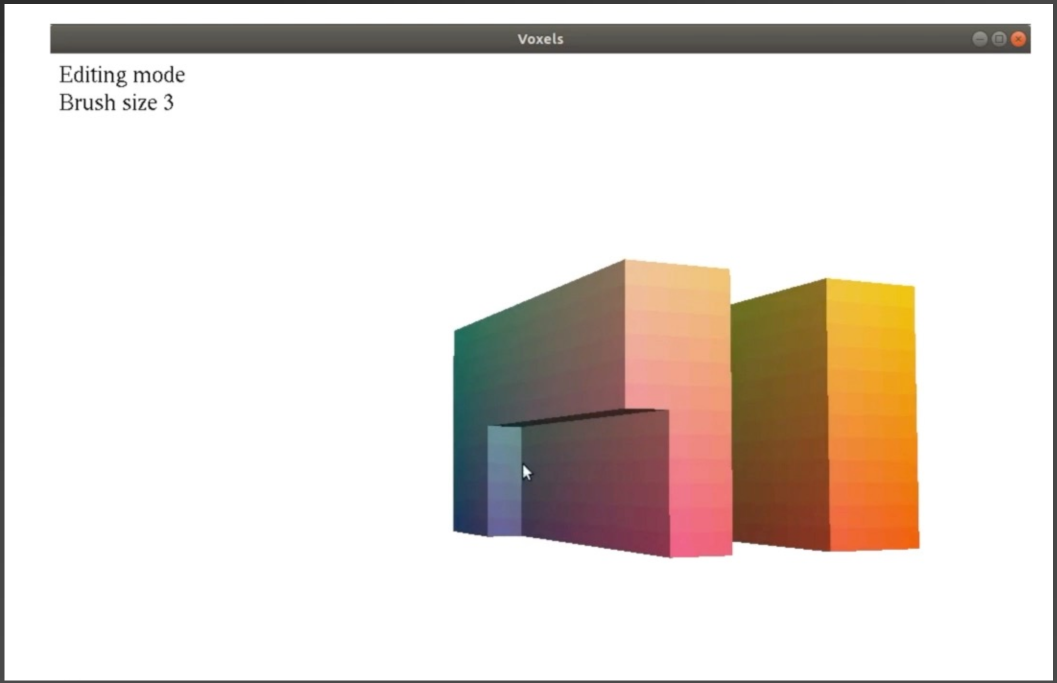
Single-view image



Neural Radiance Field (NeRF) for novel view synthesis
[Mildenhall et al. ECCV 2020]

Controllable 3D generation

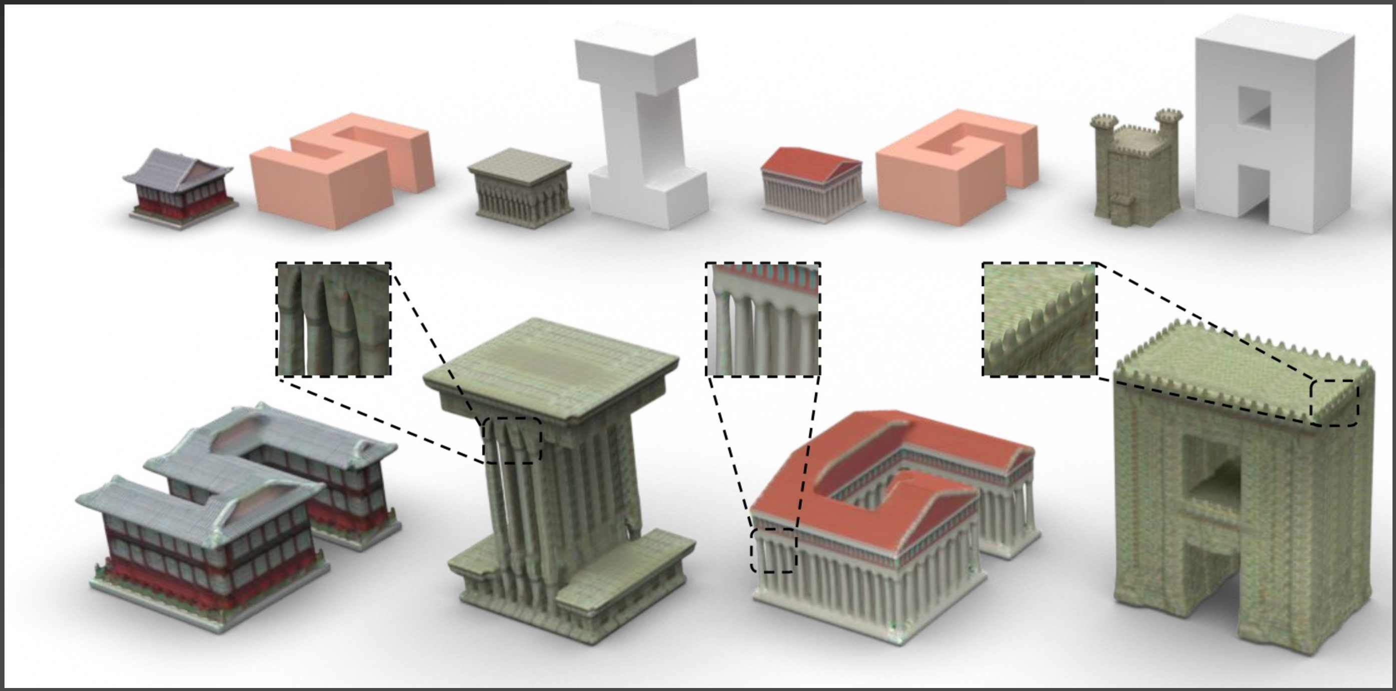
- Give creators **coarse structural control**, then **“detailize”**



ShaDDR: geometry detailization and texture generation
[Chen et al. SIGGRAPH Asia 2023]

Controllable 3D generation

- Give creators **coarse-level structural control**, then **“detailize”**



ShaDDR: geometry detailization and texture generation
[Chen et al. SIGGRAPH Asia 2023]

Text-to-X is fun, but texts are not sufficiently expressive to provide spatial control with locality, dimension, and other 3D attributes

ShaDDR: geometry detailization and texture generation
[Chen et al. SIGGRAPH Asia 2023]