Multi-View and Structured 3D Shape Representations

Richard (Hao) Zhang

CMPT 464/764: Geometric Modeling in Computer Graphics

Lecture 6

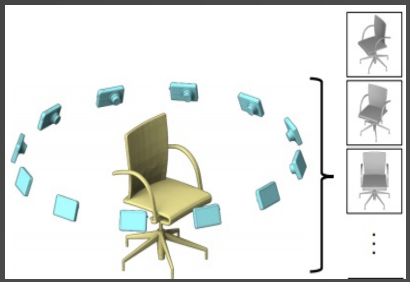
Outline on 3D representations

- Implicit reps
- Parametric reps
- Meshes (subdivision)
- Point clouds
- Volumes
- Projective reps
- Structured reps

Smooth curves and surfaces

Discrete representations

3D → 2D



Parts + relations = structures
Encompasses all low-level reps

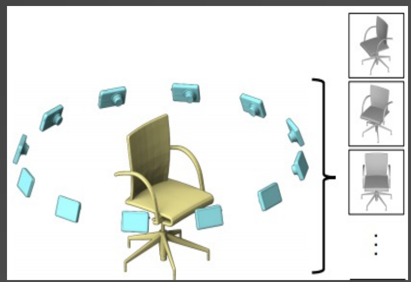
Today

- Implicit reps
- Parametric reps
- Meshes (subdivision)
- Point clouds
- Volumes
- Projective reps
- Structured reps

Smooth curves and surfaces

Discrete representations

3D → 2D

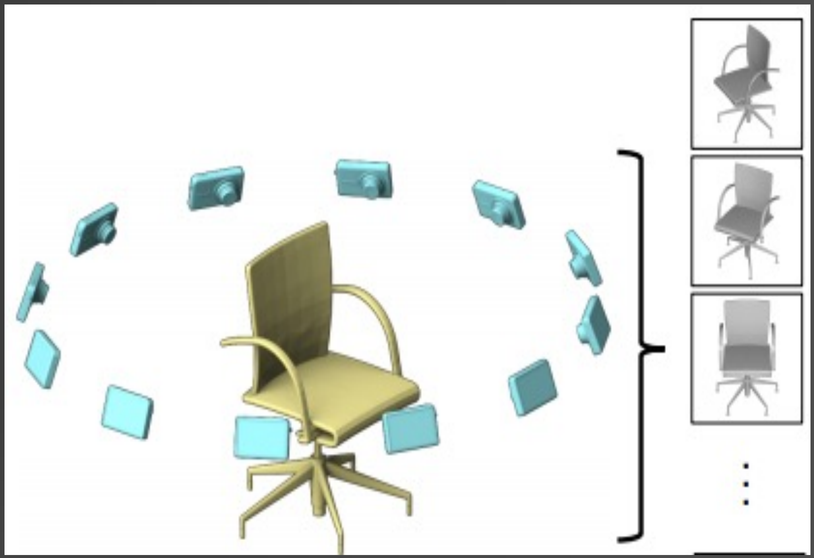


Parts + relations = structures

Encompasses all low-level reps

Multi-view representation of 3D shapes

- Represent a 3D shape as a set of **images from one or more views**
- Technically not a 3D representation
- Why would this be a good idea?
  - Images are easy to acquire



Multi-view representation of 3D shape

- Represent a 3D shape as a
  images from one or more
- Technically not a 3D represe
- Why would this be a good re

  - Images are easy to acquire
  - For visualization, all we care about is to view the 3D shape from another angle — novel view synthesis
  - Do we still need 3D then? ☺

Projective shape processing and analysis

[Wang et al. SIGGRAPH Asia 2013]

Projective Analysis for 3D Shape Segmentation

Yunhai Wang[⋆∗] Minglun Gong[†⋆] Tianhua Wang[‡⋆] Daniel Cohen-Or[※] Hao Zhang[§] Baoquan Chen[⋆♯∗]
[⋆]Shenzhen VisuCA Key Lab/SIAT [†]Memorial University of Newfoundland [‡]Jilin University
[※]Tel-Aviv University [§]Simon Fraser University [♯]Shangdong University

- Perform analysis in image space and then “back-project” into 3D

Projective shape processing and analysis

[Wang et al. SIGGRAPH Asia 2013]

Projective Analysis for 3D Shape Segmentation

Yunhai Wang[⋆∗] Minglun Gong[†⋆] Tianhua Wang[‡⋆] Daniel Cohen-Or[⁕] Hao Zhang[§] Baoquan Chen[⋆♯∗]
[⋆]Shenzhen VisuCA Key Lab/SIAT [†]Memorial University of Newfoundland [‡]Jilin University
[⁕]Tel-Aviv University [§]Simon Fraser University [♯]Shangdong University

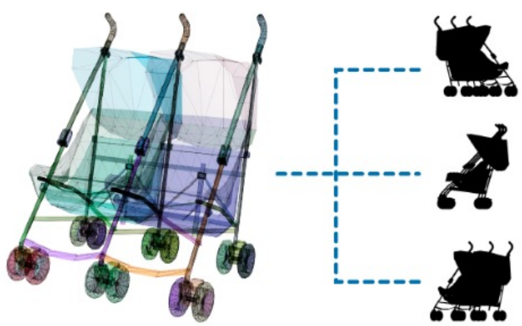
- Perform analysis in image space and then “back-project” into 3D

Projective shape processing and analysis

[Wang et al. SIGGRAPH Asia 2013]

Projective Analysis for 3D Shape Segmentation

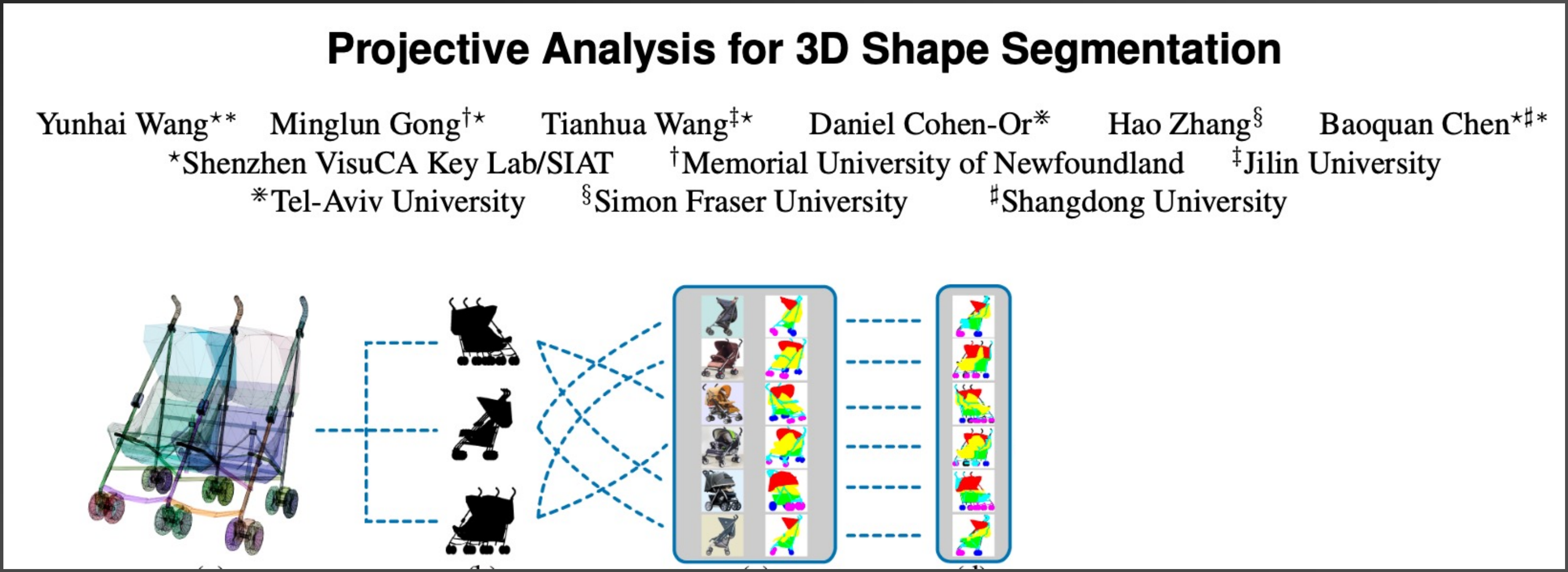
Yunhai Wang[⋆∗] Minglun Gong[†⋆] Tianhua Wang[‡⋆] Daniel Cohen-Or[⁎] Hao Zhang[§] Baoquan Chen[⋆♯∗]
[⋆]Shenzhen VisuCA Key Lab/SIAT [†]Memorial University of Newfoundland [‡]Jilin University
[⁎]Tel-Aviv University [§]Simon Fraser University [♯]Shangdong University



- Perform analysis in multi-view image space …

Projective shape processing and analysis

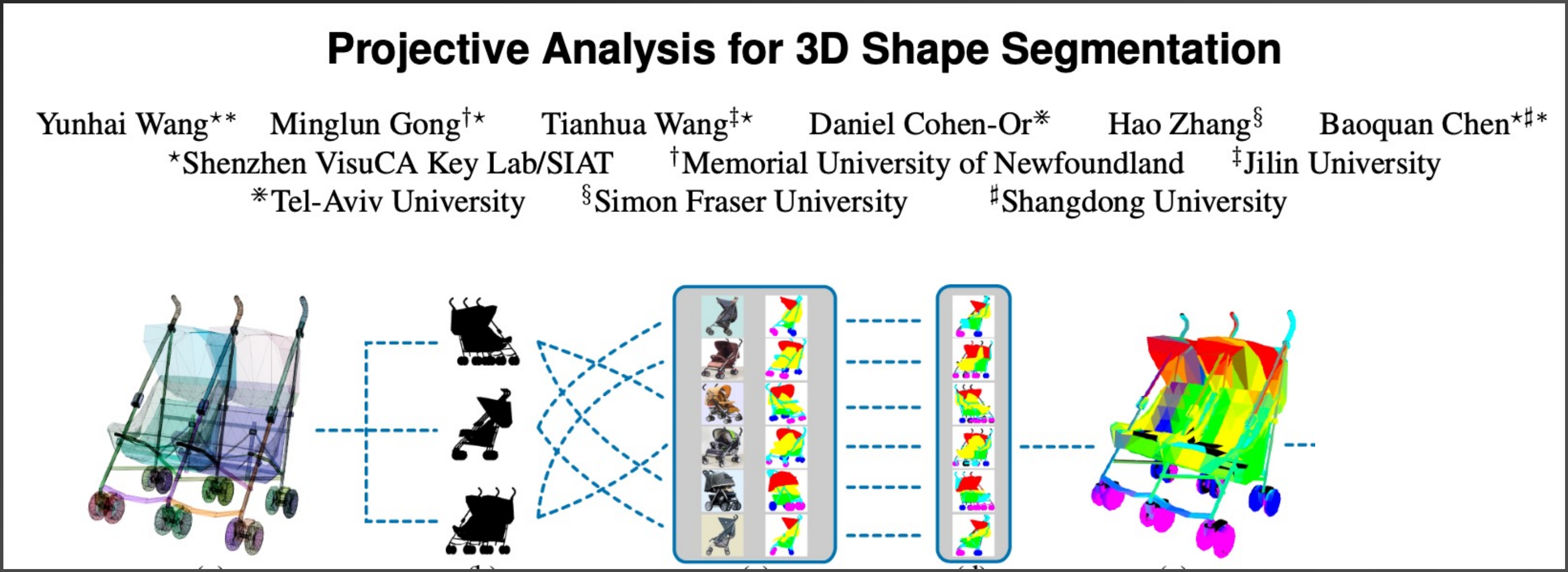
[Wang et al. SIGGRAPH Asia 2013]



- Perform analysis in image space, leveraging **large volumes of 2D segmentation training data**

Projective shape processing and analysis

[Wang et al. SIGGRAPH Asia 2013]

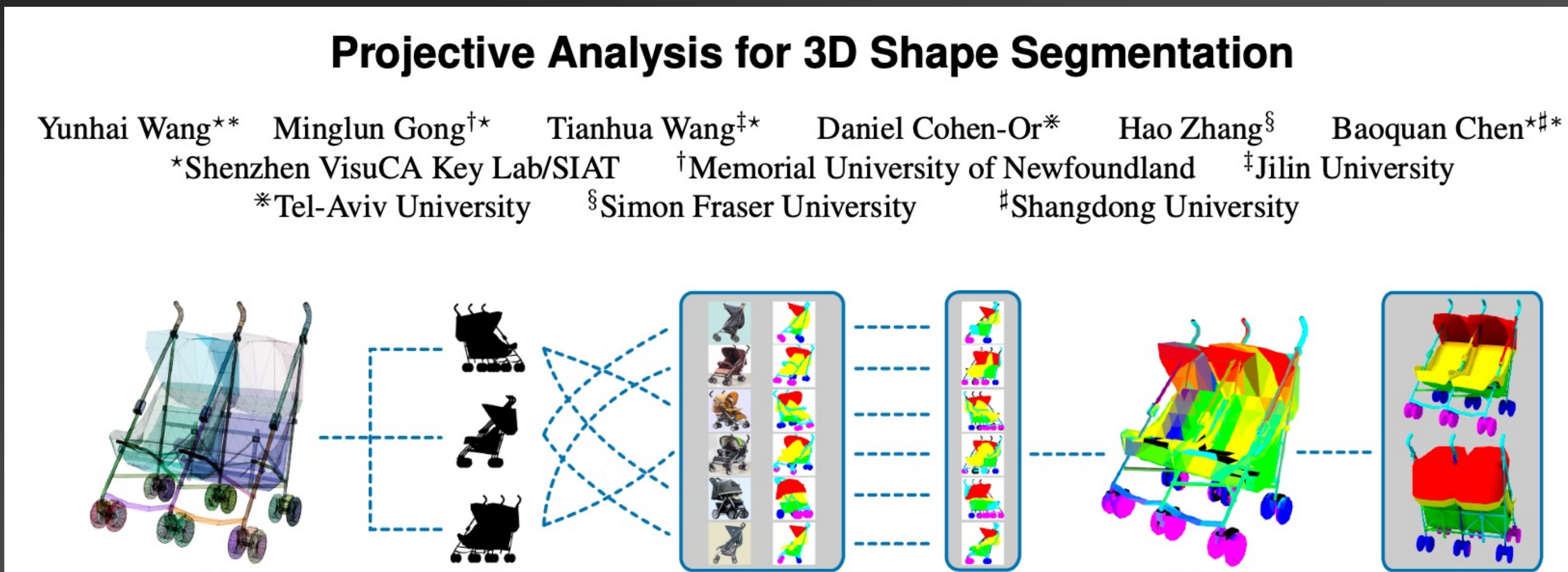


- “Back-project” multi-view segmented images into 3D: non-trivial task

Paper: https://www.yunhaiwang.net/public_html/psa.html

Projective shape processing and analysis

[Wang et al. SIGGRAPH Asia 2013]


Projective Analysis for 3D Shape Segmentation

Yunhai Wang[⋆∗] Minglun Gong[†⋆] Tianhua Wang[‡⋆] Daniel Cohen-Or[⁎] Hao Zhang[§] Baoquan Chen[⋆♯∗]

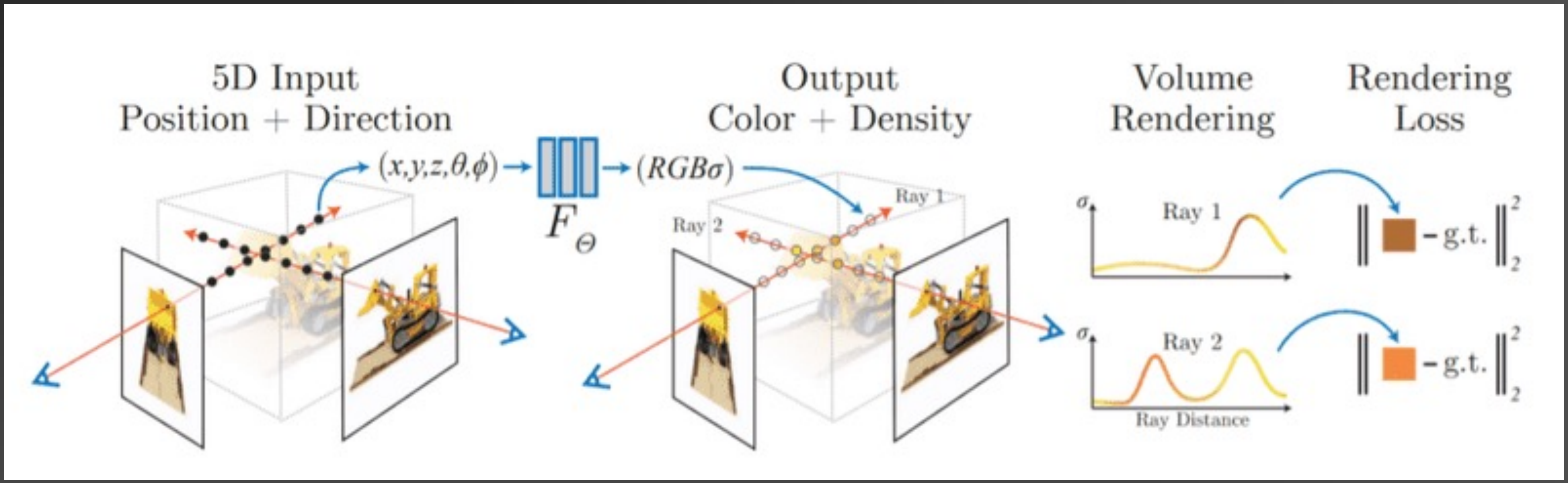
[⋆]Shenzhen VisuCA Key Lab/SIAT [†]Memorial University of Newfoundland [‡]Jilin University

[⁎]Tel-Aviv University [§]Simon Fraser University [♯]Shangdong University

- Perform analysis in image space and then "back-project" into 3D

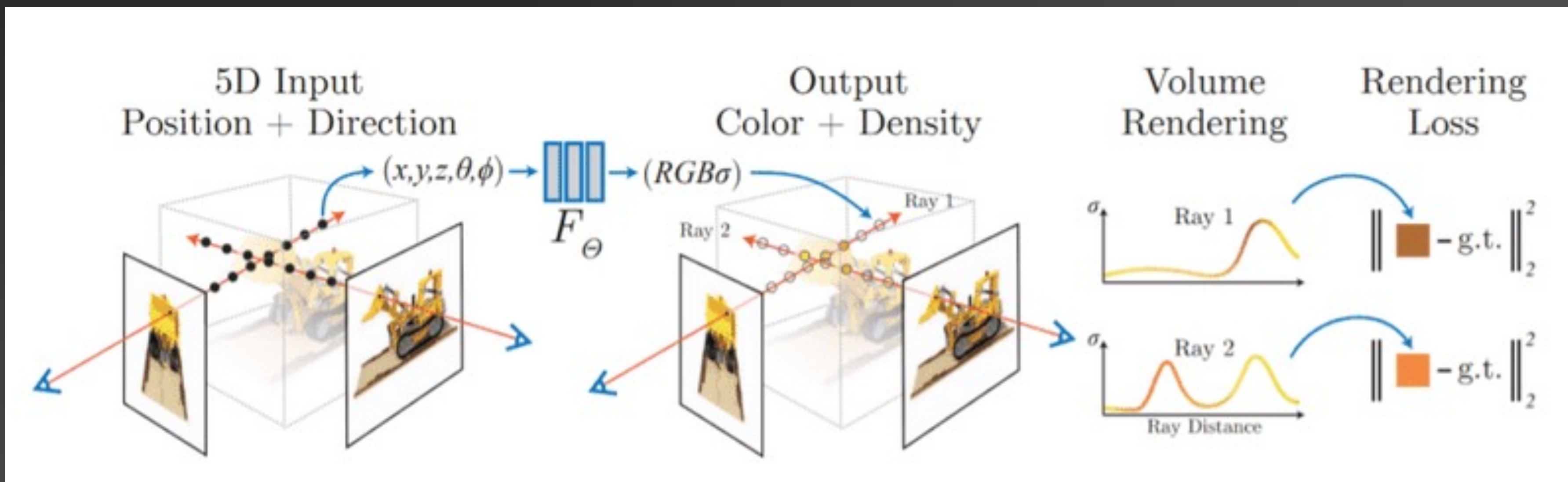
Novel view synthesis via neural networks

- **Neural Radiance Field (NeRF)** [Mildenhall et al. ECCV 2020]



Novel view synthesis via neural networks

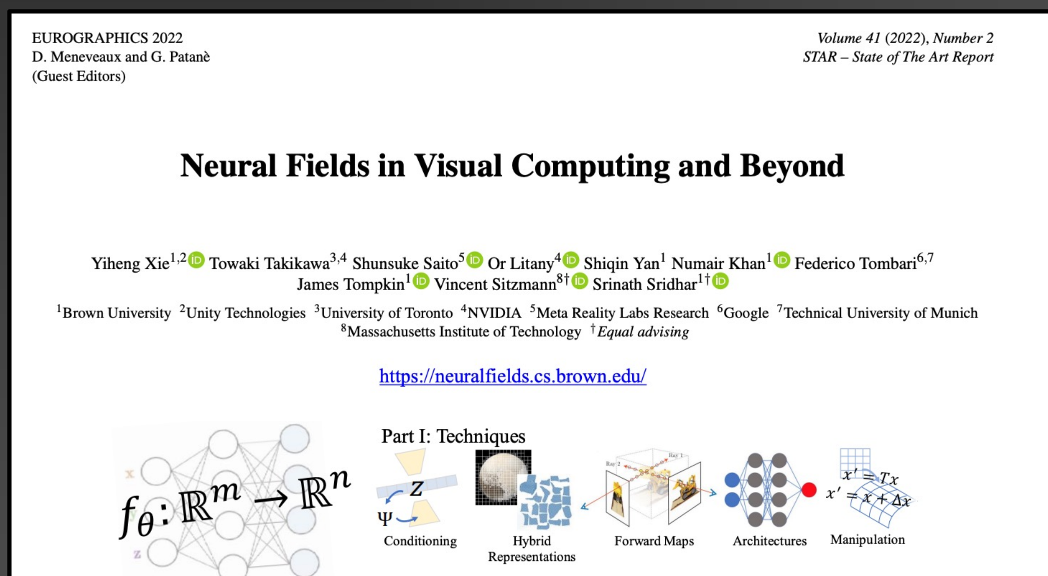
- **Neural Radiance Field (NeRF)** [Mildenhall et al. ECCV 2020]



- Input: multi-view images (a **dense** set in the initial NeRF paper)
- Output: a neural network trained via **differentiable rendering**, that can be “queried” for novel view synthesis through **volume rendering**

What is a neural field?

- A field of values (typically in voxel grids for 3D), e.g., occupancy or signed distance function (SDF), predicted using neural networks



EUROGRAPHICS 2022
D. Meneveaux and G. Patanè
(Guest Editors)

Volume 41 (2022), Number 2
STAR – State of The Art Report

Neural Fields in Visual Computing and Beyond

Yiheng Xie[1,2] Towaki Takikawa[3,4] Shunsuke Saito[5] Or Litany[4] Shiqin Yan[1] Numair Khan[1] Federico Tombari[6,7]
James Tompkin[1] Vincent Sitzmann[8†] Srinath Sridhar[1†]

[1]Brown University [2]Unity Technologies [3]University of Toronto [4]NVIDIA [5]Meta Reality Labs Research [6]Google [7]Technical University of Munich
[8]Massachusetts Institute of Technology †*Equal advising*

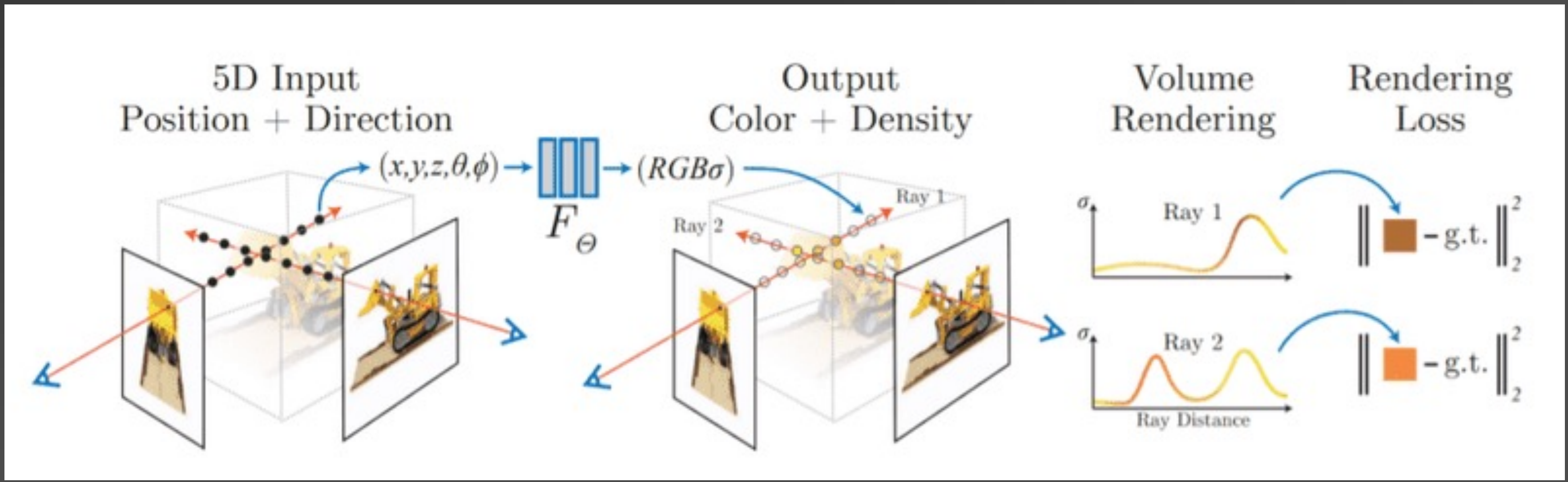
https://neuralfields.cs.brown.edu/

Survey: https://arxiv.org/abs/2111.11426

What is a neural field?

- A field of values (typically in voxel grids for 3D), e.g., occupancy or signed distance function (SDF), predicted using neural networks

- NeRF: **color** and **density** per voxel for **volume rendering**



What is a neural field?

- A field of values (typically in voxel grids for 3D), e.g., occupancy or signed distance function (SDF), predicted using neural networks
- NeRF: **color and density** per voxel for **volume rendering**
- The 3D representation **is the neural network** that is trained on multi-view, or even single-view, images: an **"overfit"** network

What is a neural field?

- A field of values (typically in voxel grids for 3D), e.g., occupancy or signed distance function (SDF), predicted using neural networks
- NeRF: **color and density** per voxel for **volume rendering**
- The 3D representation is the **neural network** that is trained on multi-view, or even single-view, images: an **"overfit"** network
- A **compressed** representation that can be **quickly queried**

3D Gaussian splatting (3DGS)

- 3DGS [Kerbl et al. SIG 2023] surpassing NeRF (2000) in speed
- Also from multi-view images as input and produces a PBR

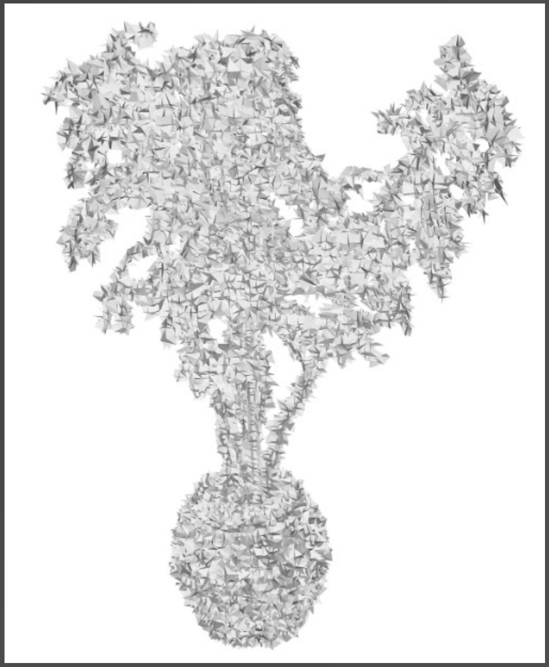
Critical issue #1

- 🙁 NeRF and 3DGS: **rendering, not modeling** primitives

Images from MobileNeRF [Chen et al. CVPR 2023]

Critical issue #1

- 🙁 NeRF and 3DGS: **rendering, not modeling** primitives
  - Unstructured: not how human reasons about 3D
  - Not editable/reusable
  - Not functional



Images from MobileNeRF [Chen et al. CVPR 2023]

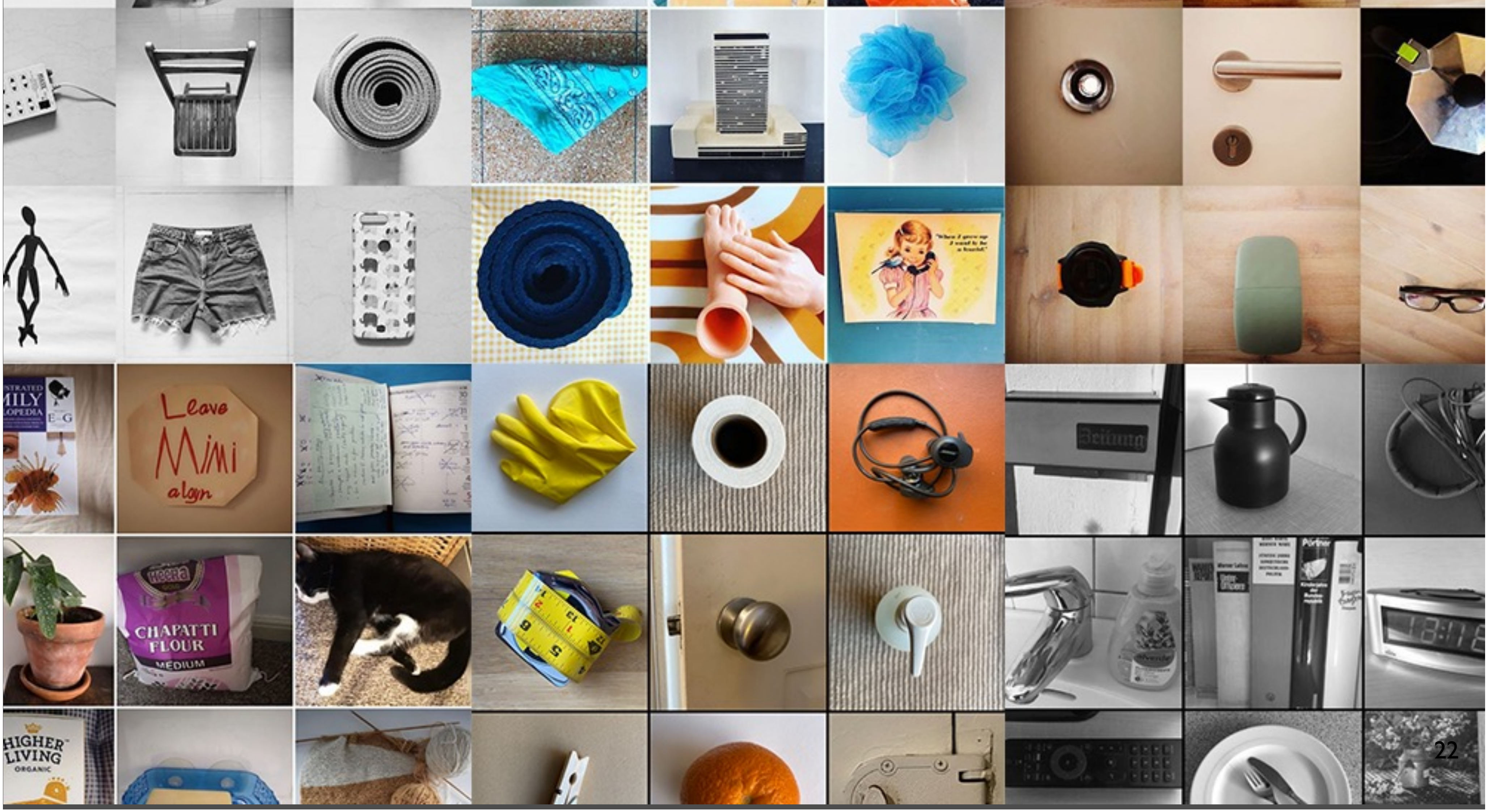
Critical issue #2

- 🙁 NeRF and 3DGS: **rendering, not modeling** primitives
- 🙁 Avoiding 3D supervision is **unnatural**

vs.

Learning by interacting in 3D

Think of an object class, e.g., cars, lamps, chairs, buildings, etc., and what properties or attributes you are most sure of about it.



Predictability of object attributes from class

Which object attribute is more predicable from class label, e.g., chairs?

- Shape?
- Topology?

Image taken from dreamstimes.com

Predictability of object attributes from class

Which object attribute is more predicable from class label, e.g., chairs?

- Color?
- Texture?
- Material?

Image taken from pinterest.ca

Is this a chair or not?

Why is this not a chair? Or is it?

What makes a chair a chair?

CVPR 2011

What Makes a Chair a Chair?

Helmut Grabner[1] Juergen Gall[1] Luc Van Gool[1,2]

[1]Computer Vision Laboratory
ETH Zurich
{grabner,gall,vangool}@vision.ee.ethz.ch

[2]ESAT - PSI / IBBT
K.U. Leuven
luc.vangool@esat.kuleuven.be

Abstract

Many object classes are primarily defined by their functions. However, this fact has been left largely unexploited by visual object categorization or detection systems. We propose a method to learn an affordance detector. *It identifies locations in the 3d space which "support" the particular function. Our novel approach "imagines" an actor performing an action typical for the target object class, instead of relying purely on the visual object appearance. So, function is handled as a cue complementary to appearance, rather than being a consideration after appearance-based detection. Experimental results are given for the functional category "sitting". Such affordance is tested on a 3d representation of the scene, as can be realistically obtained through SfM or depth cameras. In contrast to appearance-based object detectors, affordance detection requires only very few training examples and generalizes very well to other sittable objects like benches or sofas when trained on a few chairs.*

1. Introduction

"An object is first identified as having important functional relations, [...] perceptual analysis is derived of the functional concept [...]."

Nelson, 1974, [17]

"Affordances relate the utility of things, events, and places to the needs of animals and their actions in fulfilling them [...]. Affordances themselves are perceived and, in fact, are the essence of what we perceive."

Gibson, 1982, [8, p. 60]

"There's little we can find in common to all chairs – except for their intended use."

Minsky, 1986, [16, p. 123]

"[...] objects like coffee cups are artifacts that were created to fulfill a function. The function of an object plays a critical role in processing that object [... for] categorization and naming."

Carlson-Radvansky *et al.*, 1999, [4]

Figure 1. The "chair-challenge" by I. and H. Bülthoff [3] (reprint with the author's permission).

These quotes emphasize that functional properties or affordances[1] are essential for forming concepts and learning object categories. Experiments (*e.g.* [18, 4]) have demonstrated that both appearance and function are strong cues for learning by infants. Initially they attend only to the form of an object. Later they use form and function and finally (by the age of 18 months) they attend to the relationships between form and function. Furthermore, Booth and Waxman [2] have identified two salient cues that facilitate categorization in infancy, namely (i) object functions and (ii) object names. Moreover, names of objects most often evolve on the basis of function[2].

Whereas all this is well known for a long time, it has been left mostly unused for object detection in computer vision. Taking a look at the results of the recent Pascal VOC Challenge [5], the performance still strongly depends

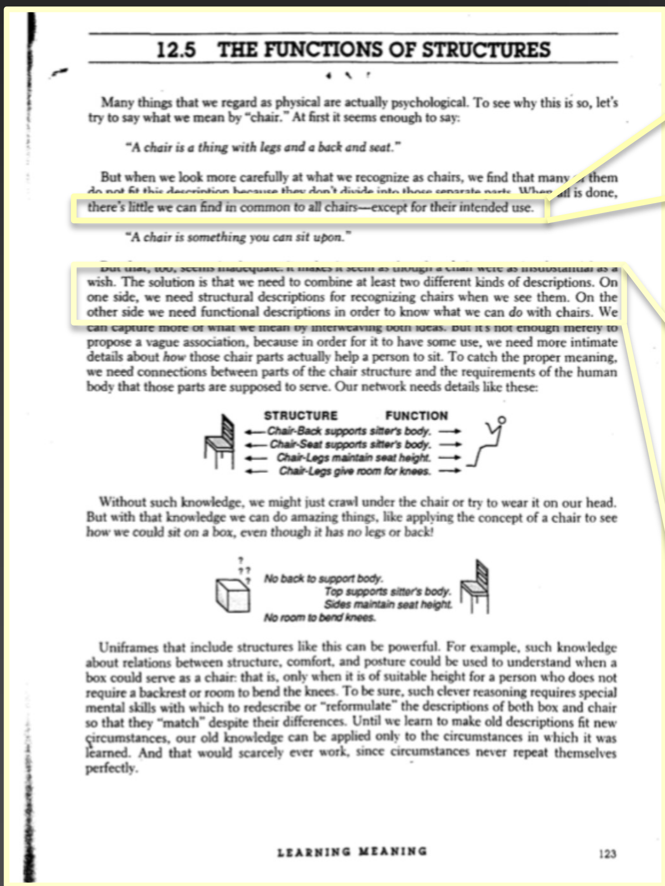
[1]"*Affordance:* A situation where an object's sensory characteristics intuitively imply its functionality and use. [...] A chair, by its size, its curvature, its balance, and its position, suggests sitting on it.", `http://www.usabilityfirst.com/glossary/affordance`, 2010/07/28. Introduced in 1979 by Gibson [9, p. 127] based on the verb *afford*.

[2]When considering the evolution of a word for an object, most of the time it is based on its function. For example the word "chair": PIE base **sed-* (to sit) → Latin *sedentarius* (sitting, remaining in one place) → *sedentary* (meaning "not in the habit of exercise") → *cathedral* → *chair*. `http://www.etymonline.com`, 2010/10/02.

"There's little we can find in common to all chairs – except for their intended use."

Marvin Minsky: "The Society of Mind" [1986]

Marvin Minsky's "The Society of Mind"



12.5 THE FUNCTIONS OF STRUCTURES

Many things that we regard as physical are actually psychological. To see why this is so, let's try to say what we mean by "chair." At first it seems enough to say:

"*A chair is a thing with legs and a back and seat.*"

But when we look more carefully at what we recognize as chairs, we find that many of them do not fit this description because they don't divide into those separate parts. When all is done, there's little we can find in common to all chairs—except for their intended use.

"*A chair is something you can sit upon.*"

But that, too, seems inadequate: it makes it seem as though a chair were as insubstantial as a wish. The solution is that we need to combine at least two different kinds of descriptions. On one side, we need structural descriptions for recognizing chairs when we see them. On the other side we need functional descriptions in order to know what we can *do* with chairs. We can capture more of what we mean by interweaving both ideas. But it's not enough merely to propose a vague association, because in order for it to have some use, we need more intimate details about *how* those chair parts actually help a person to sit. To catch the proper meaning, we need connections between parts of the chair structure and the requirements of the human body that those parts are supposed to serve. Our network needs details like these:

Without such knowledge, we might just crawl under the chair or try to wear it on our head. But with that knowledge we can do amazing things, like applying the concept of a chair to see how we could sit on a box, even though it has no legs or back!

Uniframes that include structures like this can be powerful. For example, such knowledge about relations between structure, comfort, and posture could be used to understand when a box could serve as a chair: that is, only when it is of suitable height for a person who does not require a backrest or room to bend the knees. To be sure, such clever reasoning requires special mental skills with which to redescribe or "reformulate" the descriptions of both box and chair so that they "match" despite their differences. Until we learn to make old descriptions fit new circumstances, our old knowledge can be applied only to the circumstances in which it was learned. And that would scarcely ever work, since circumstances never repeat themselves perfectly.

"There's little we can find in common to all chairs – except for their intended use."

"… we need to combine at least two different kinds of descriptions (of objects). On one side, we need structural descriptions for recognizing chairs when we see them. On the other side, we need functional descriptions in order to know what we can *do* with chairs."

Predictability of shape structures

- More predictability in the structure of objects from the same class, than other attributes such as color, shape, etc.
- This is due to the common functionality of the objects

Predictability of shape structures

- More predictability in the structure of objects from the same class, than other attributes such as color, shape, etc.
- This is due to the common functionality of the objects
- Functionality is characterized by object parts + relations,

Dictionary

Definitions from Oxford Languages · Learn more

struc·ture

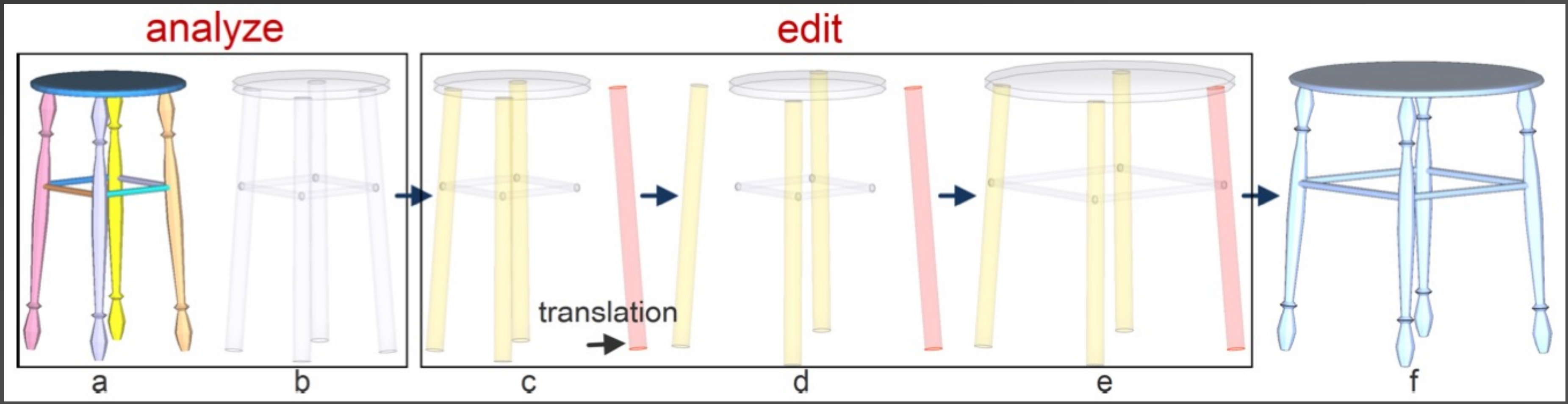
/ˈstrək(t)SHər/

noun

the arrangement of and relations between the parts or elements of something complex.

Structure relations and application

- Part symmetry, repetitions, proximity, angles, relative positioning (of parts) such as co-planarity, etc.
- Application: structure-aware editing

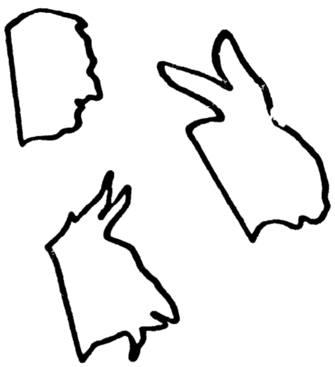


[Zheng et al. 2010]

Mimicking human perception

Cognition, 18 (1984) 65–96

2



Parts of recognition*

D.D. HOFFMAN
University of California, Irvine

W.A. RICHARDS
Massachusetts Institute of Technology

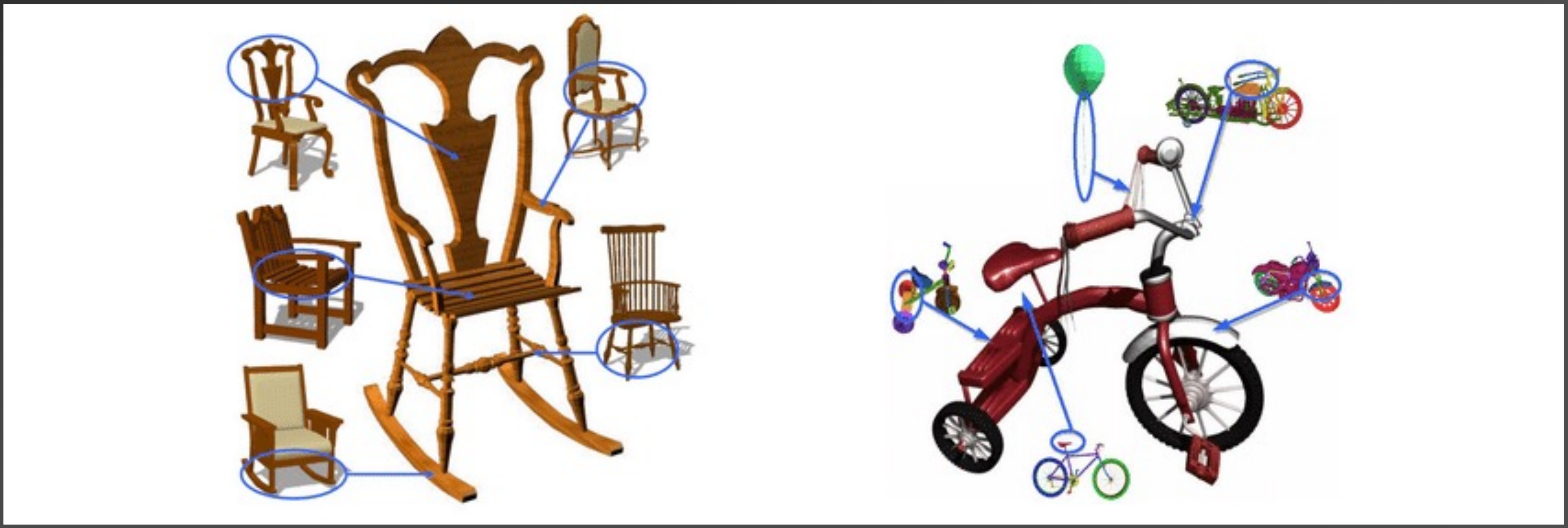
Abstract

We propose that, for the task of object recognition, the visual system decomposes shapes into parts, that it does so using a rule defining part boundaries rather than part shapes, that the rule exploits a uniformity of nature—transversality, and that parts with their descriptions and spatial relations provide a first index into a memory of shapes. This rule allows an explanation of several visual illusions. We stress the role inductive inference in our theory and conclude with a précis of unsolved problems.

"For the task of object recognition, the visual system decomposes shapes into parts, . . ., parts with their descriptions and spatial relations provide a first index into a memory of shapes …

Another benefit of structured models

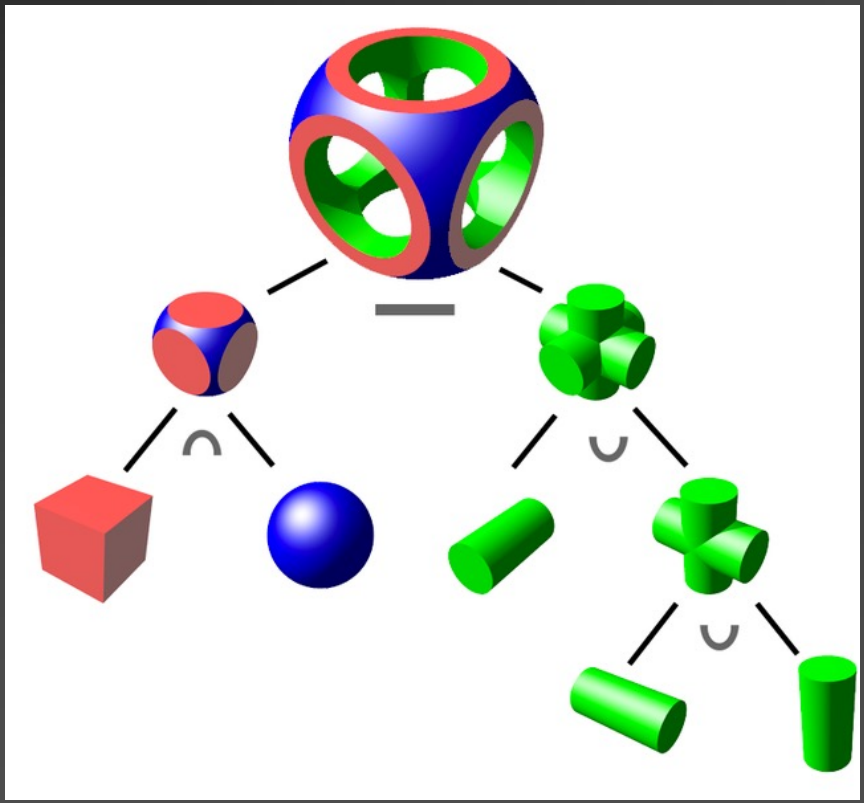
- Reusability, e.g., part re-assembly for 3D model creation



"Modeling by Example" [Funkhouser et al. SIGGRAPH 2004]

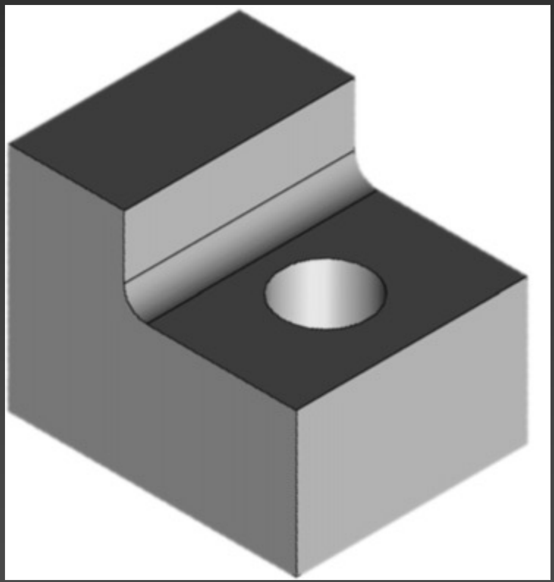
Structured representation: CSG

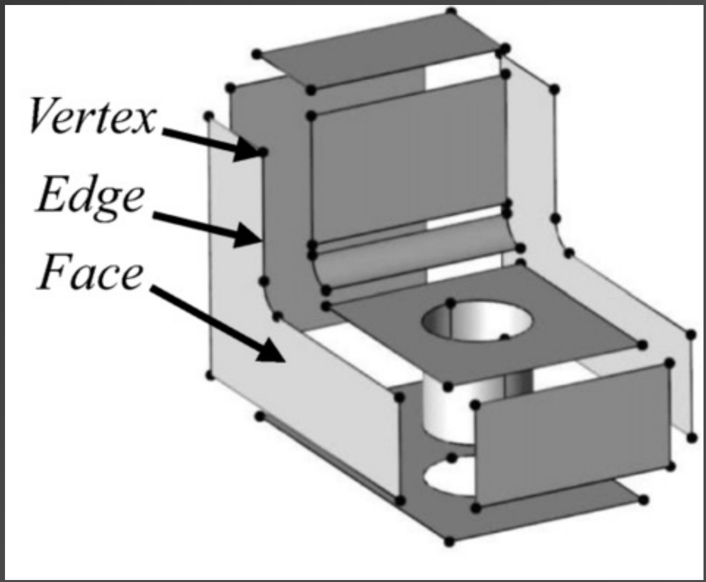
- Constructed solid geometry (CSG) for CAD models



Structured representation: B-Rep

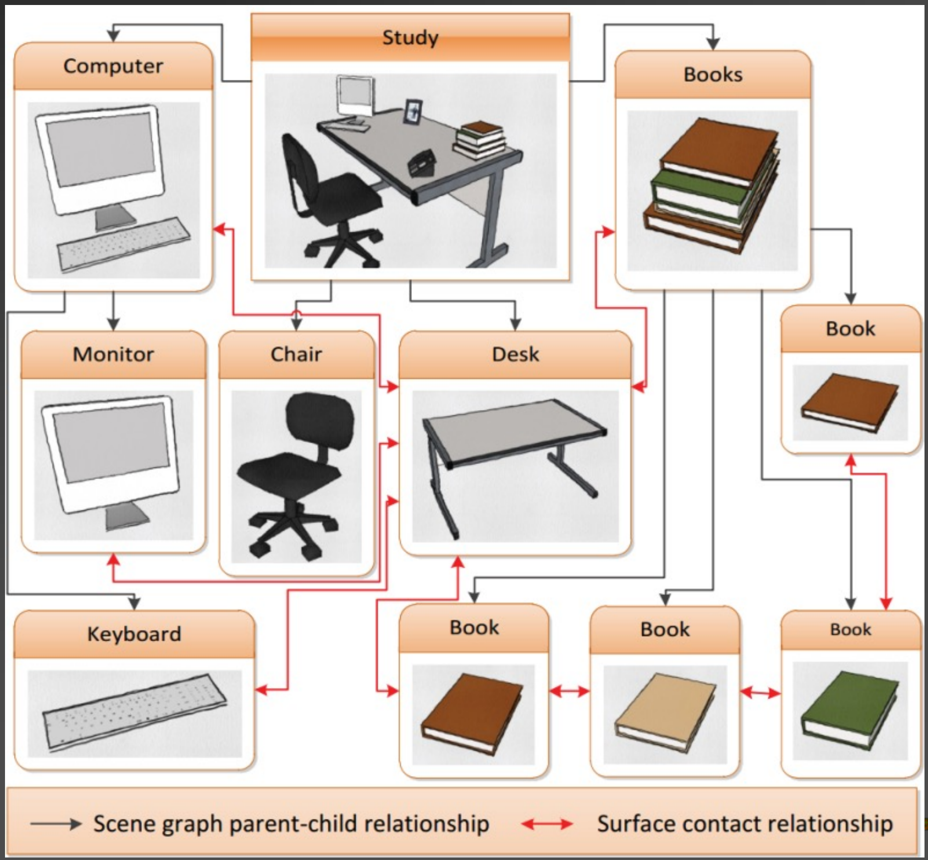
- Boundary representation (B-Rep) for CAD models





Structured representation: scene graphs

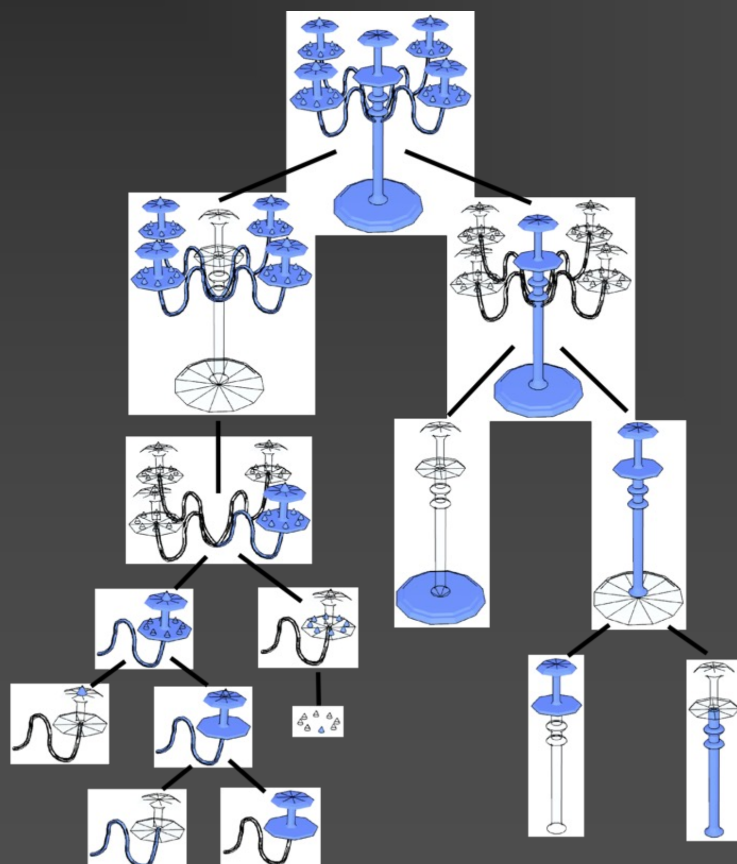
- For indoor scenes and object layouts/arrangements



Structured representation: symmetry hierarchy

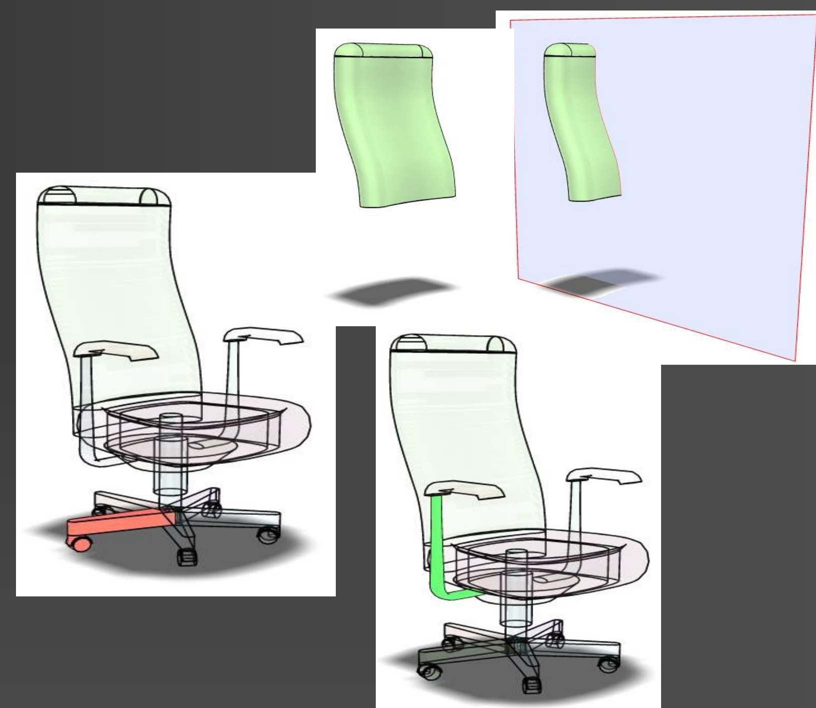
Symmetry hierarchy: symmetry guides grouping and assembly of shape parts to form a meaningful hierarchical part organization.

[Wang et al. 2011]



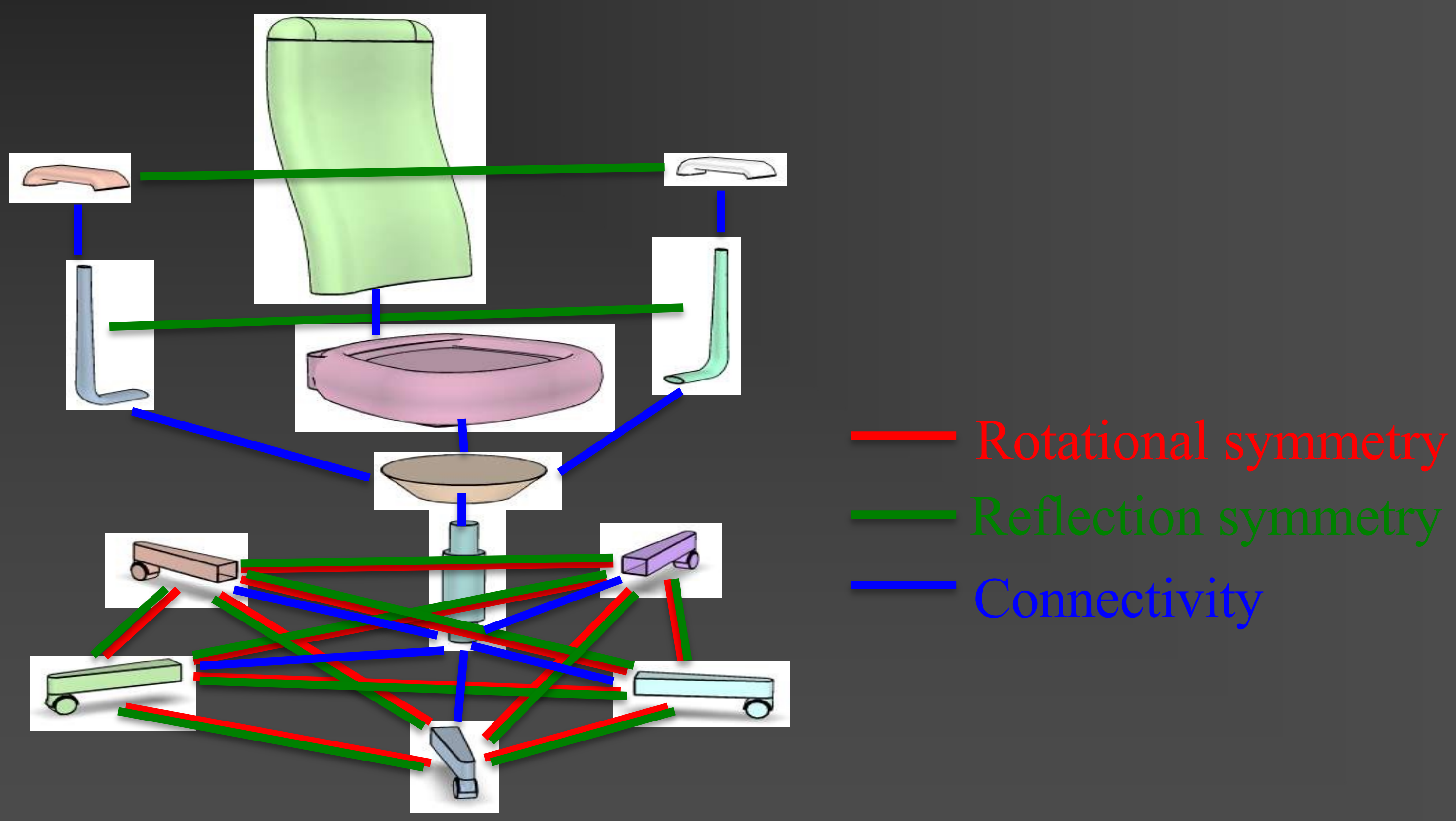
Symmetry hierarchy construction

Part segmentation

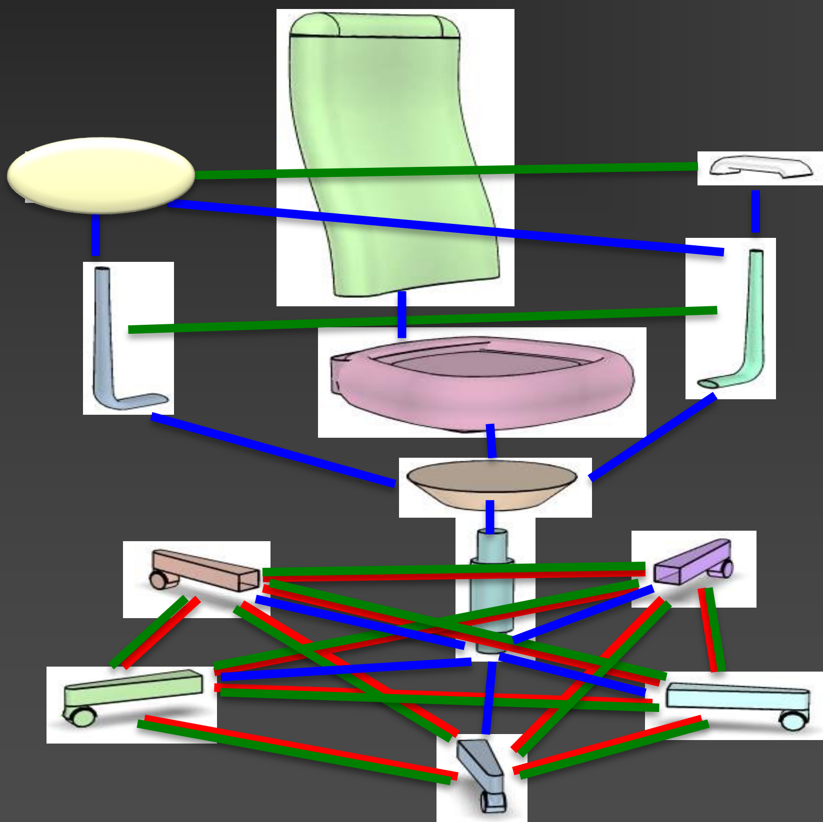


Symmetry detection

Initial graph



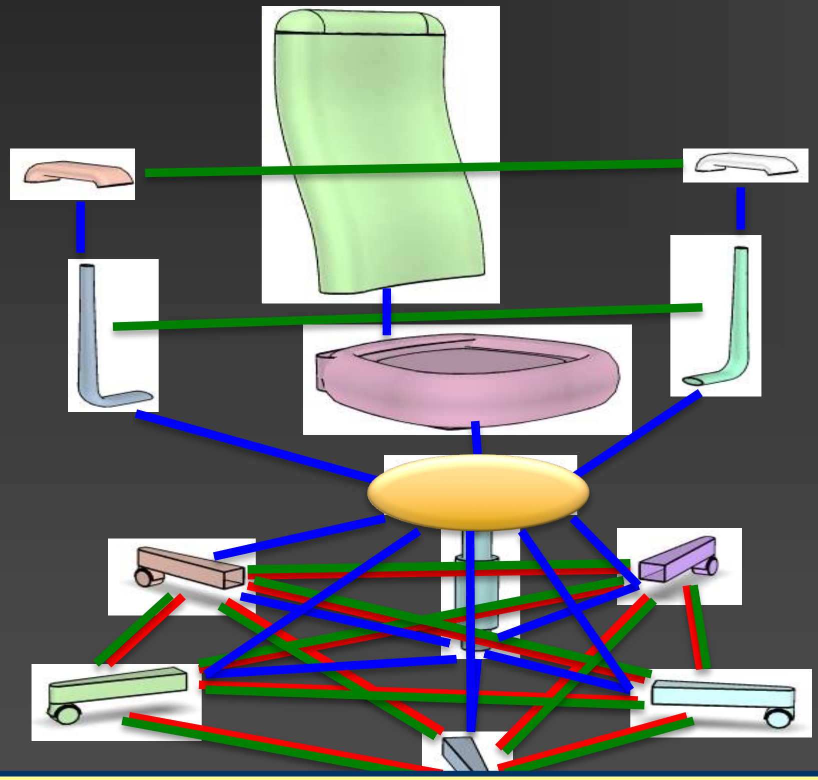
Bottom-up graph construction



Two operations:

Grouping by symmetry

Bottom-up graph construction



Two operations:

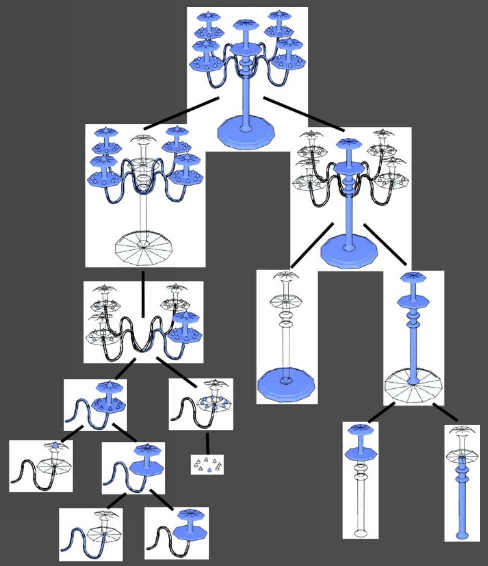
Grouping by symmetry

Assembly by proximity

Key question: in which order should we group the parts?

Symmetry hierarchy properties

- Structure-aware: hierarchical part organization
- Functionality-aware (a bit): symmetric parts tend to perform the same function
- Decouple structure and fine part geometry
  - Tree organization reveals part structure
  - Leaf nodes reveal part geometries



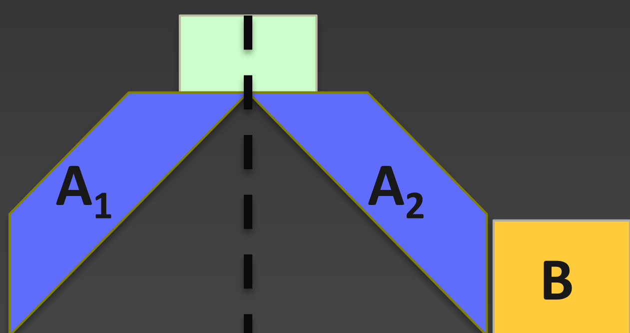
How to order graph contractions

- First attempt (2011): a model- or rule-driven approach
- Two guiding principles
  1. Perceptual grouping based on symmetry
  2. Compactness of representation = fewest grouping operations
- Results in a set of handcrafted “precedence rules”

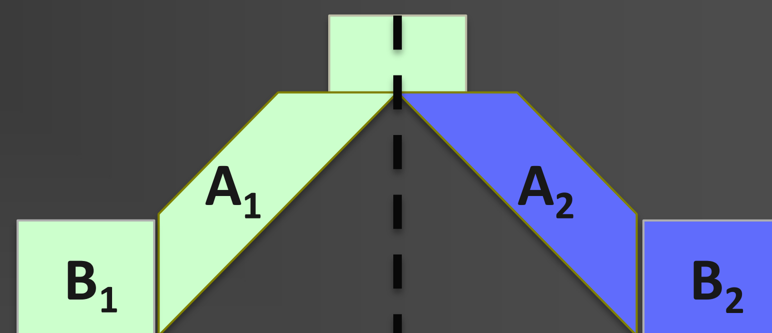
[Wang et al. EG 2011]

Example of handcrafted precedence rules

- Grouping-assembly mixing rules, e.g.,
  - M1: symmetry grouping takes precedence over assembly



unless



M2: Assembly before grouping: 2 ops

Grouping before assembly: 3 ops

Example of handcrafted precedence rules

- Symmetry grouping rules, e.g.,
  - G1: rot-symmetry takes precedence over ref-symmetry