

3D Modeling for Everyone

Hsueh-Ti Derek Liu

University of Toronto

3D modeling



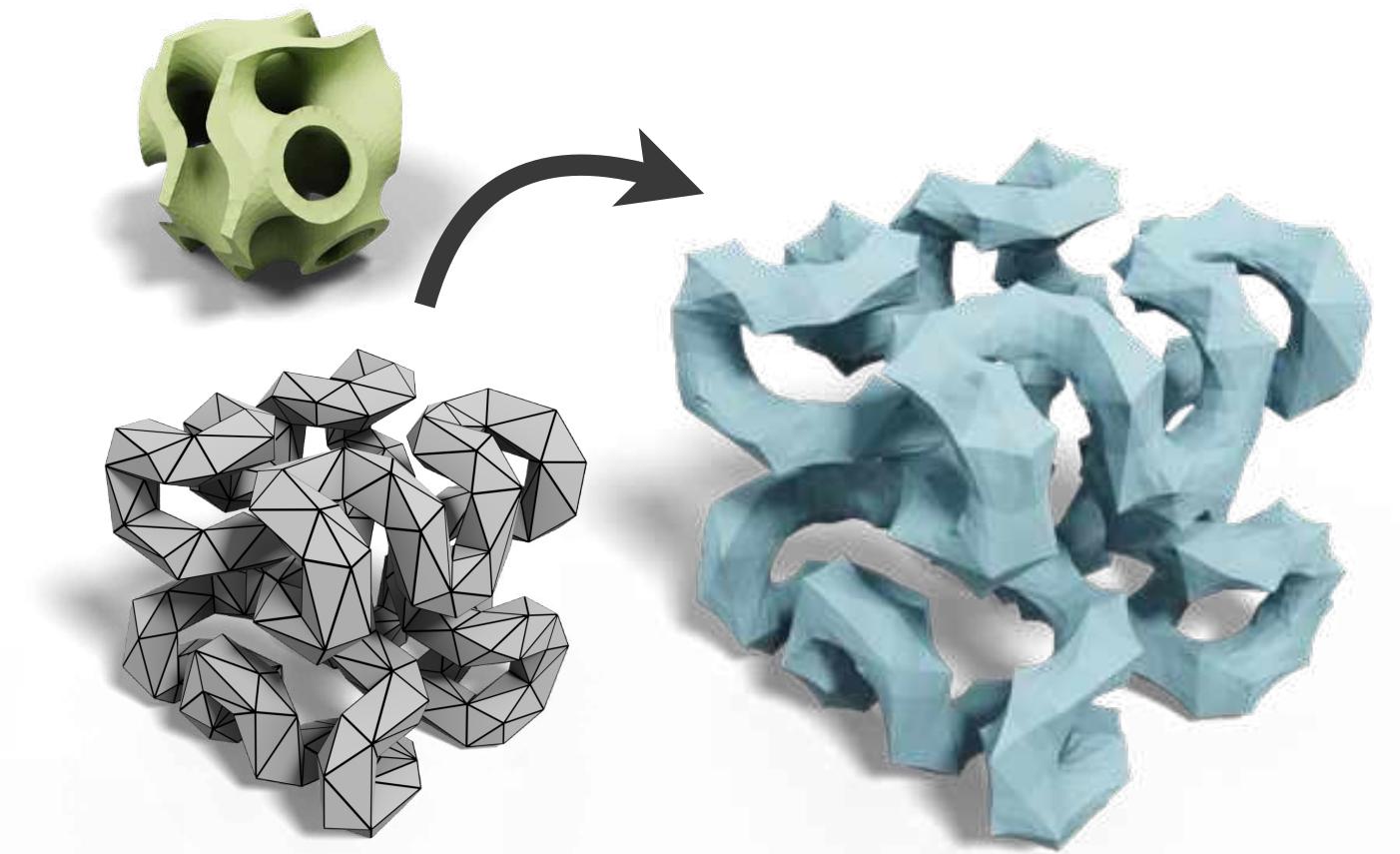
SIGGRAPH Asia 2018



ICLR 2019



SIGGRAPH Asia 2019

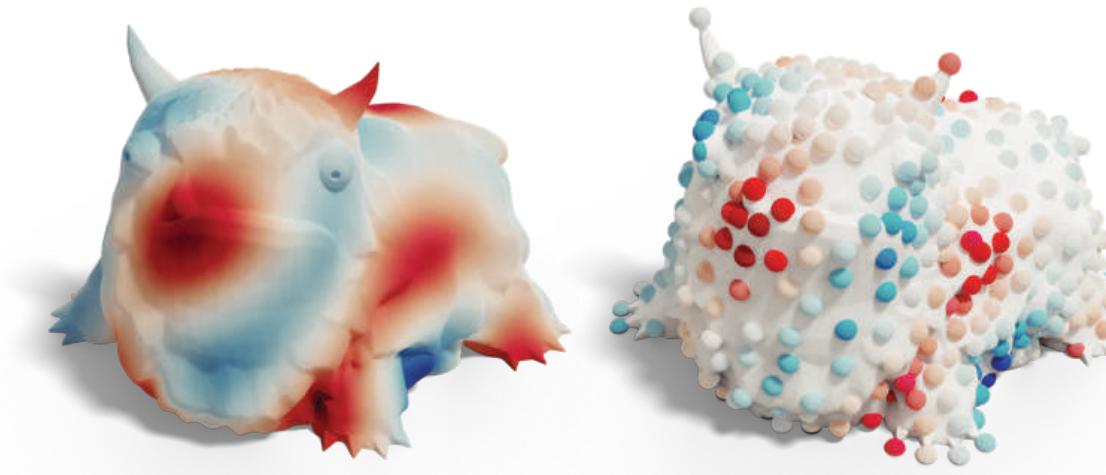


SIGGRAPH 2020

Spectral geometry



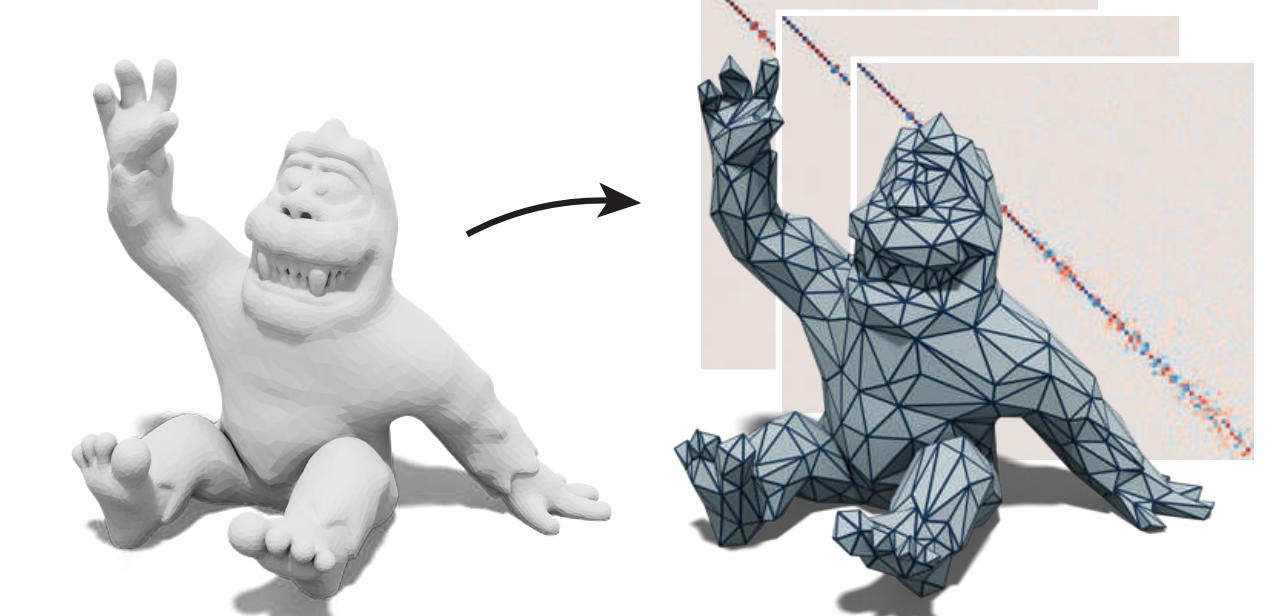
SGP 2017



SIGGRAPH 2019

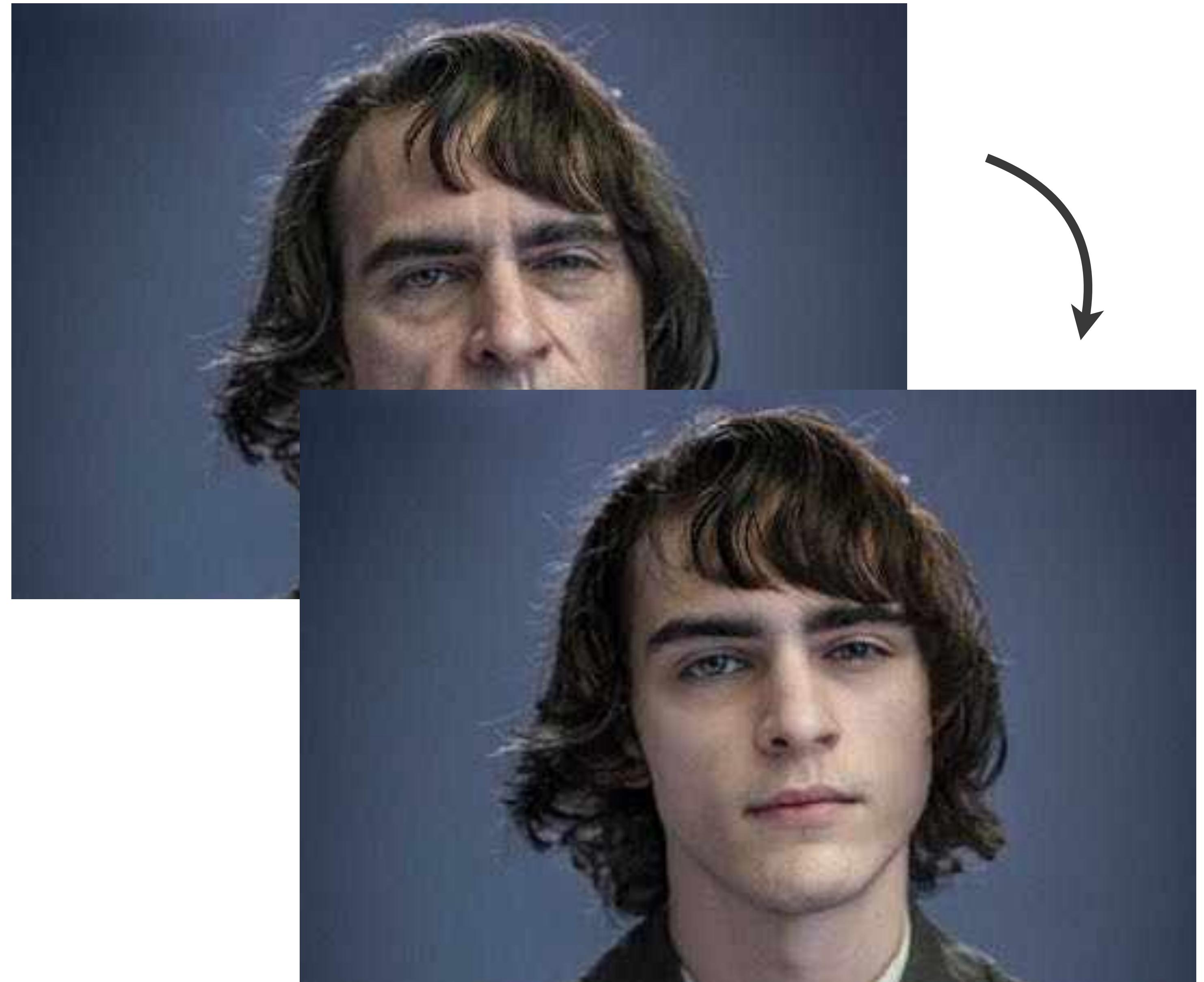


Eurographics 2019



SIGGRAPH Asia 2020

Image Editing



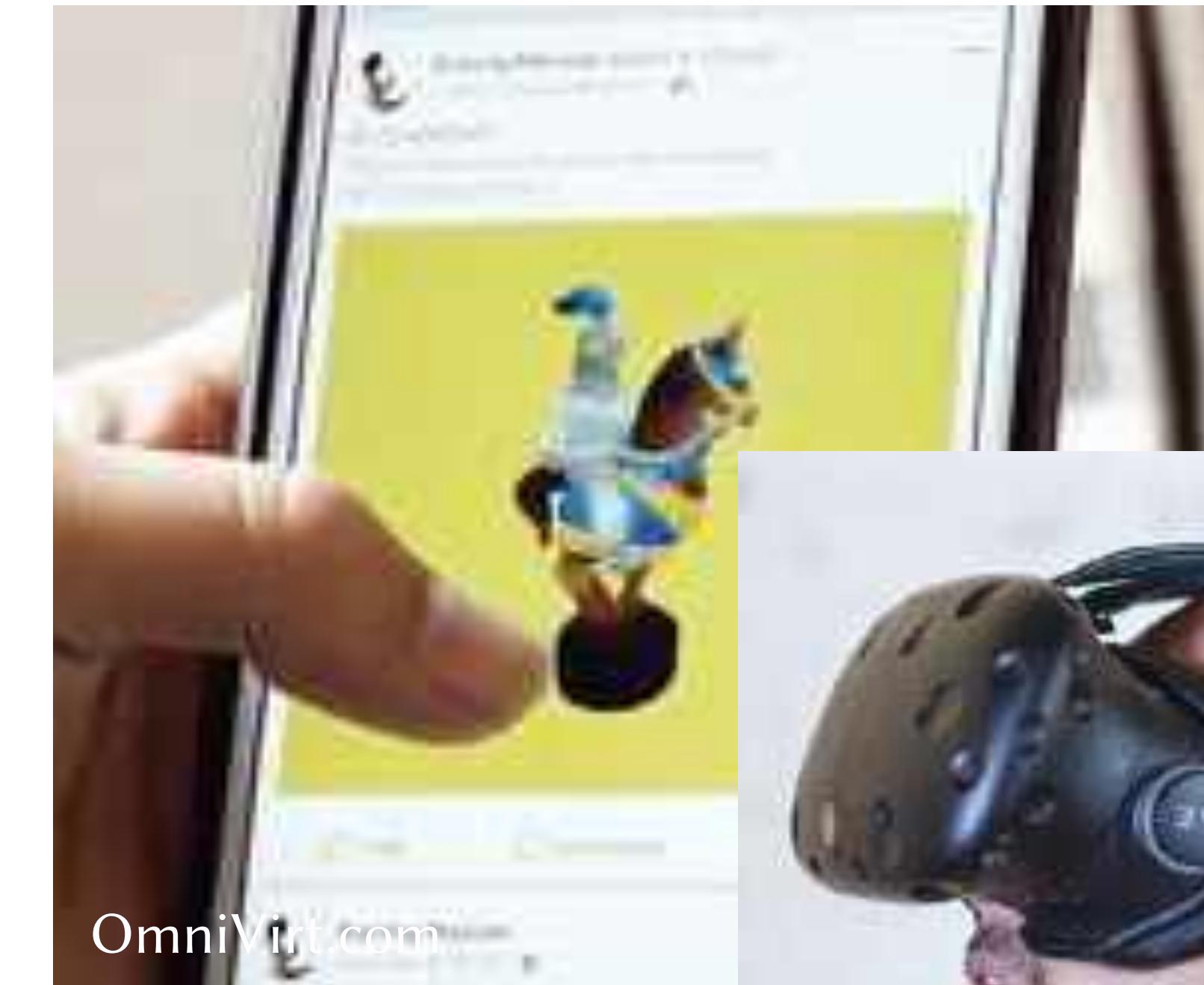
Advances in 3D Technology



medium.com



Macworld.com



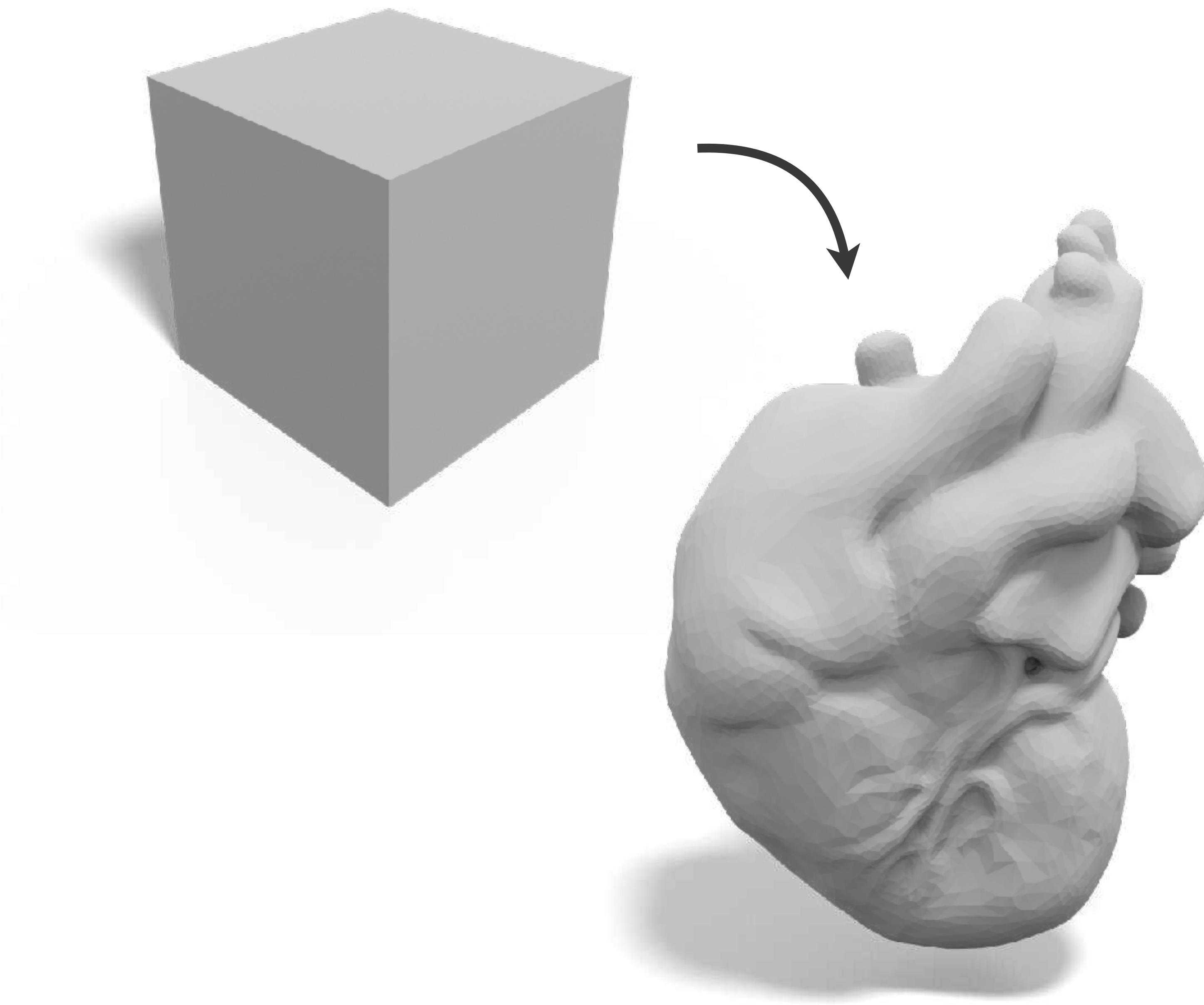
OmniVirt.com



3D display

3D scanning

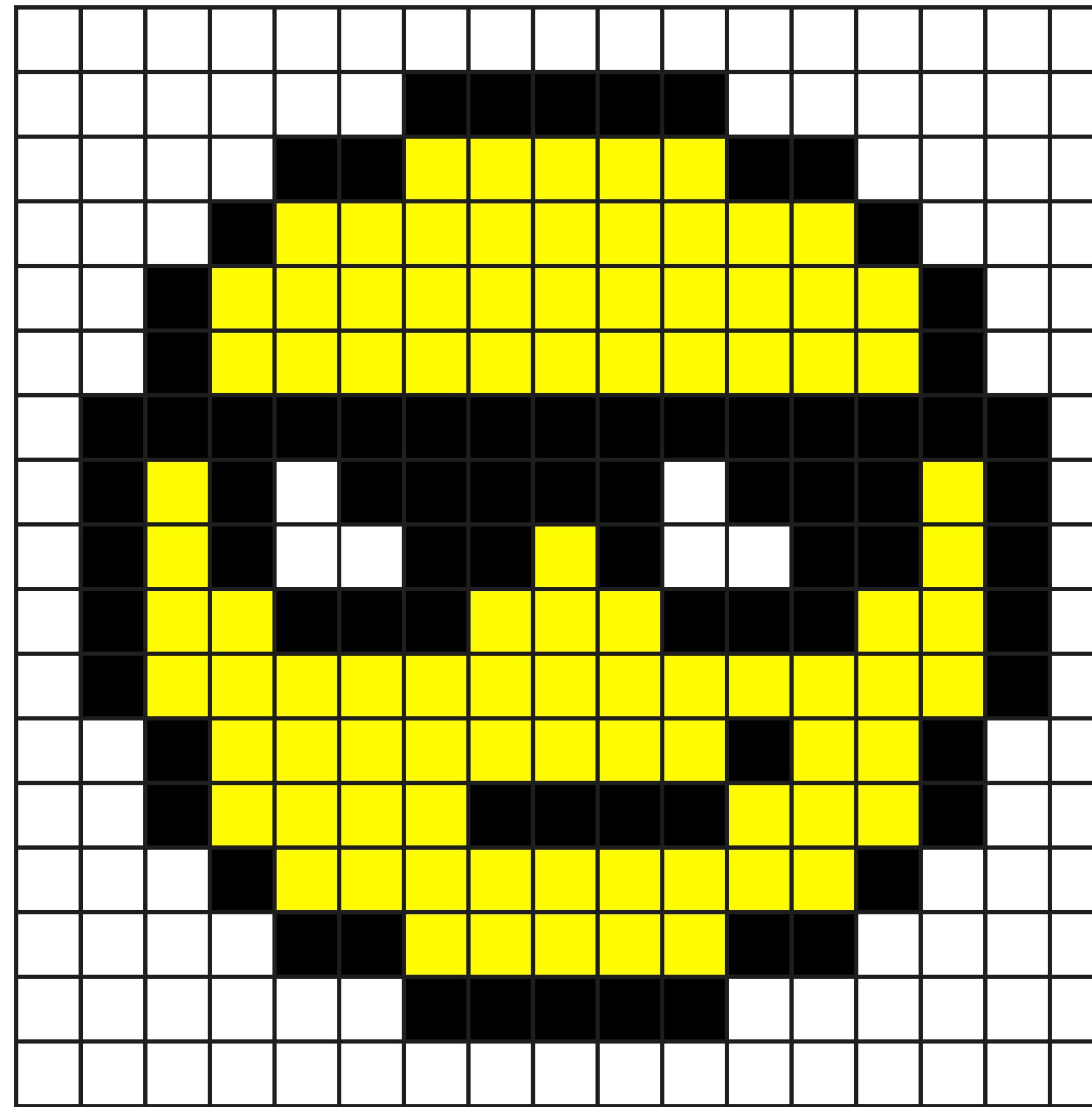
3D Modeling



3D Modeling
is Hard



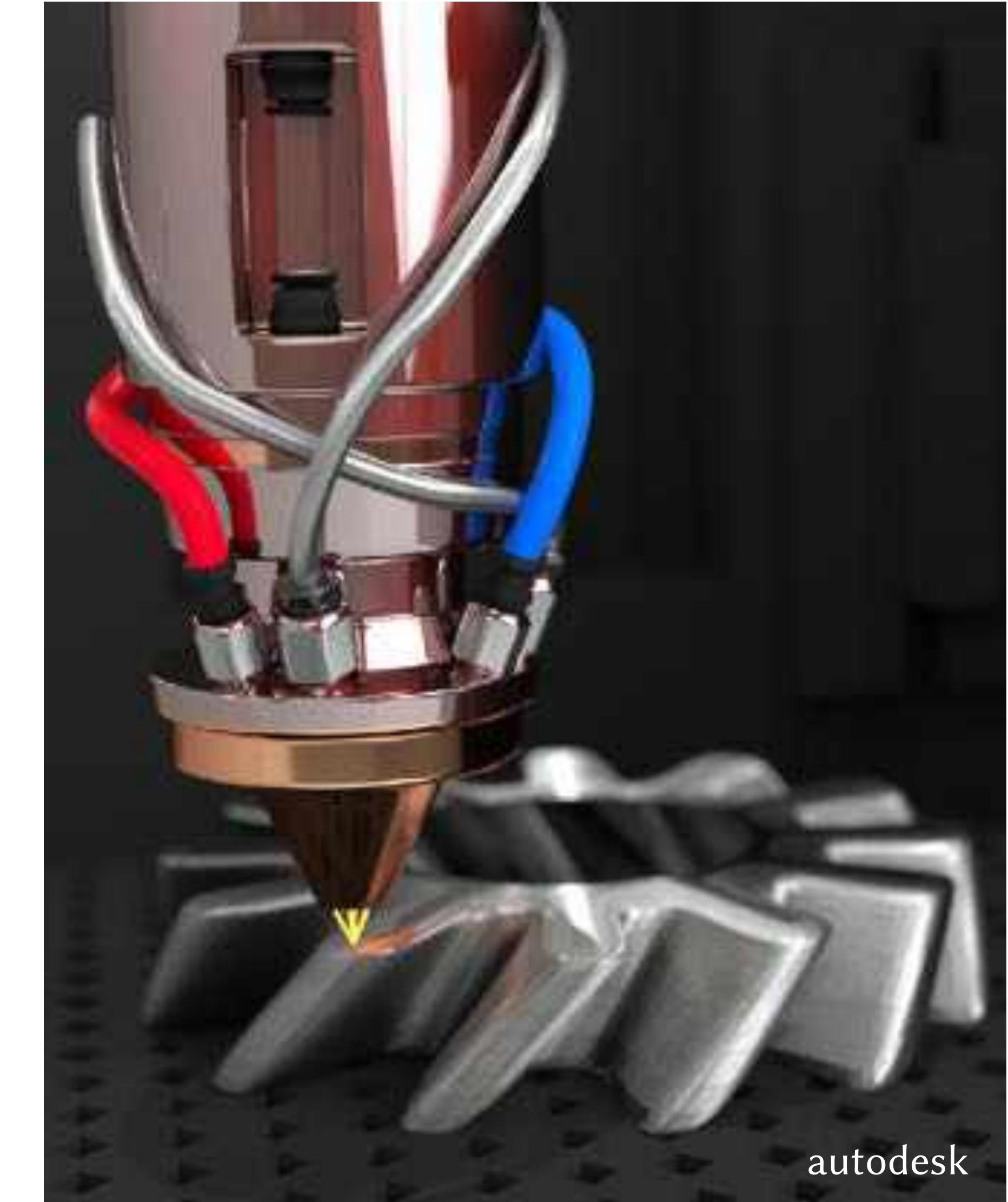
2D Analogue



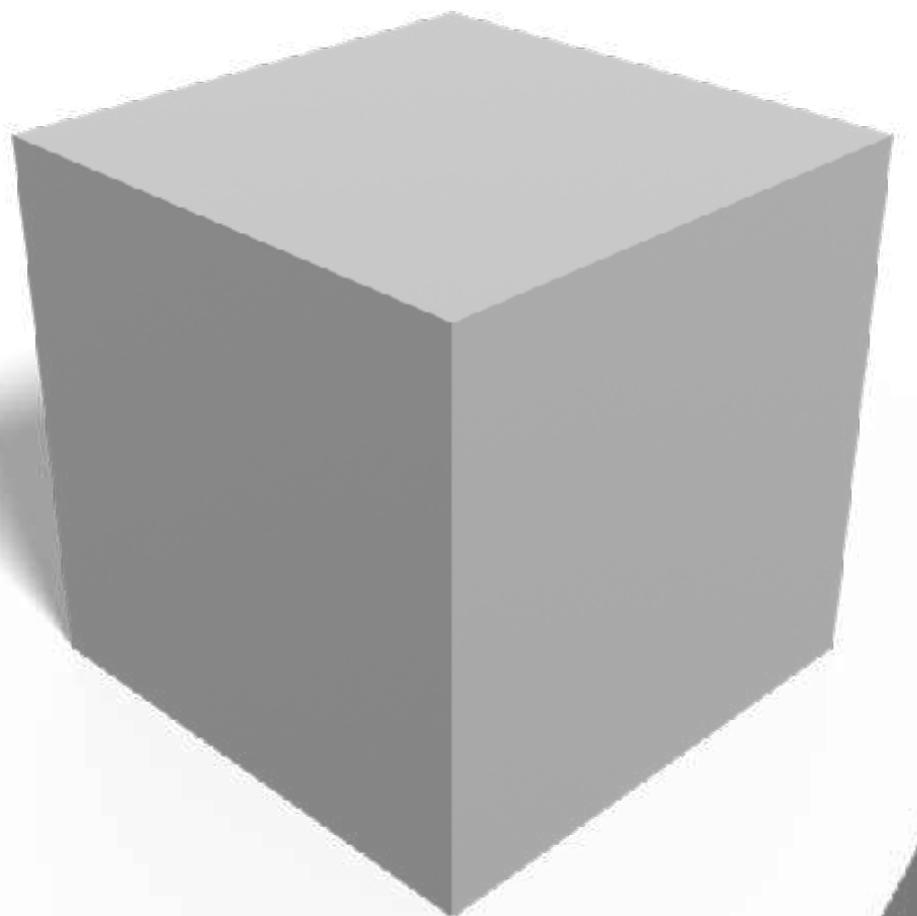
Make 3D modeling available for everyone



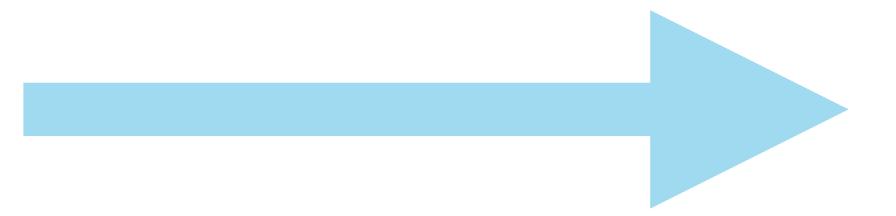
3D modeling is a long-standing problem



Classic 3D Modeling



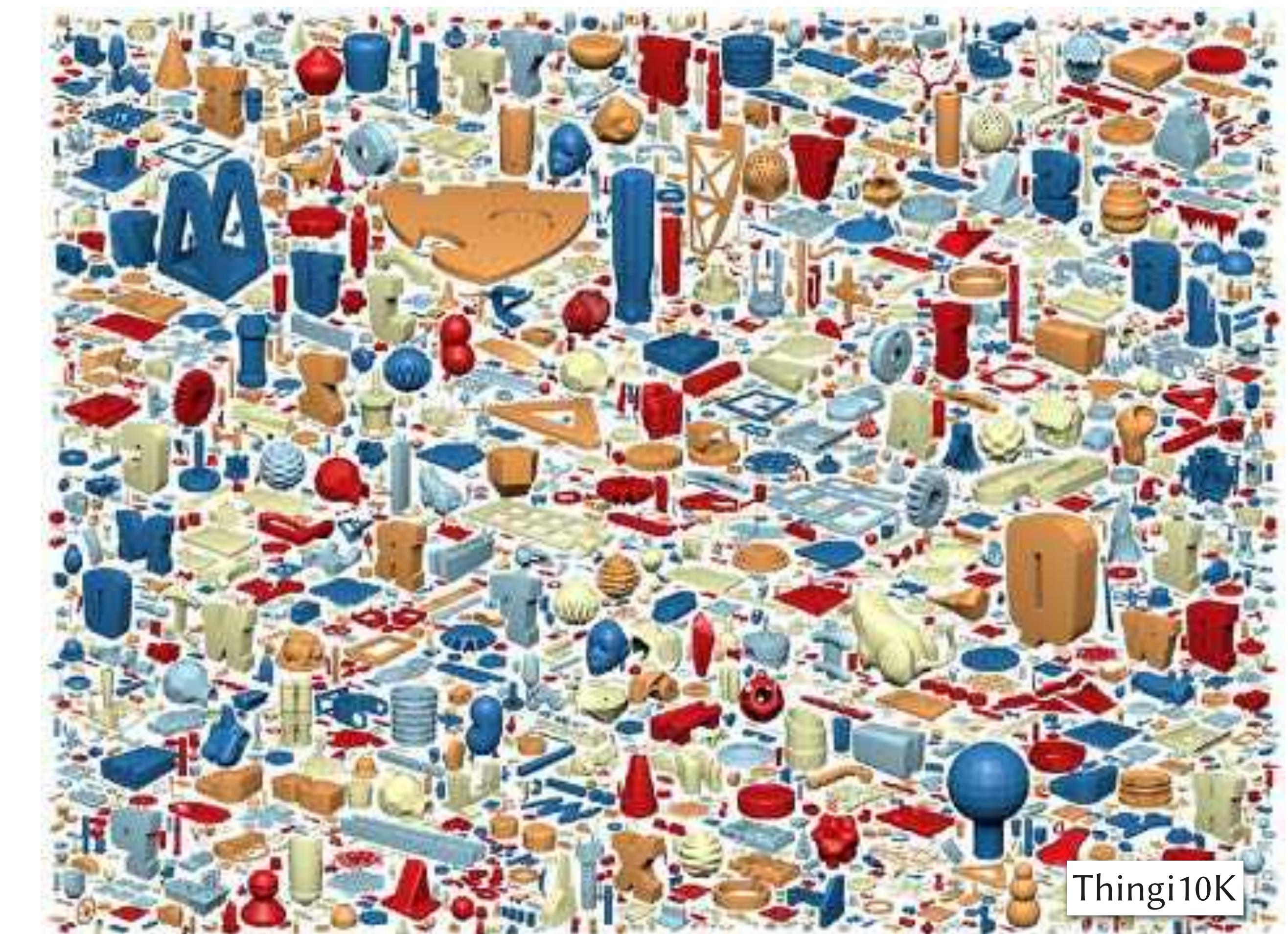
primitive shape



Acquiring 3D data is easy nowadays

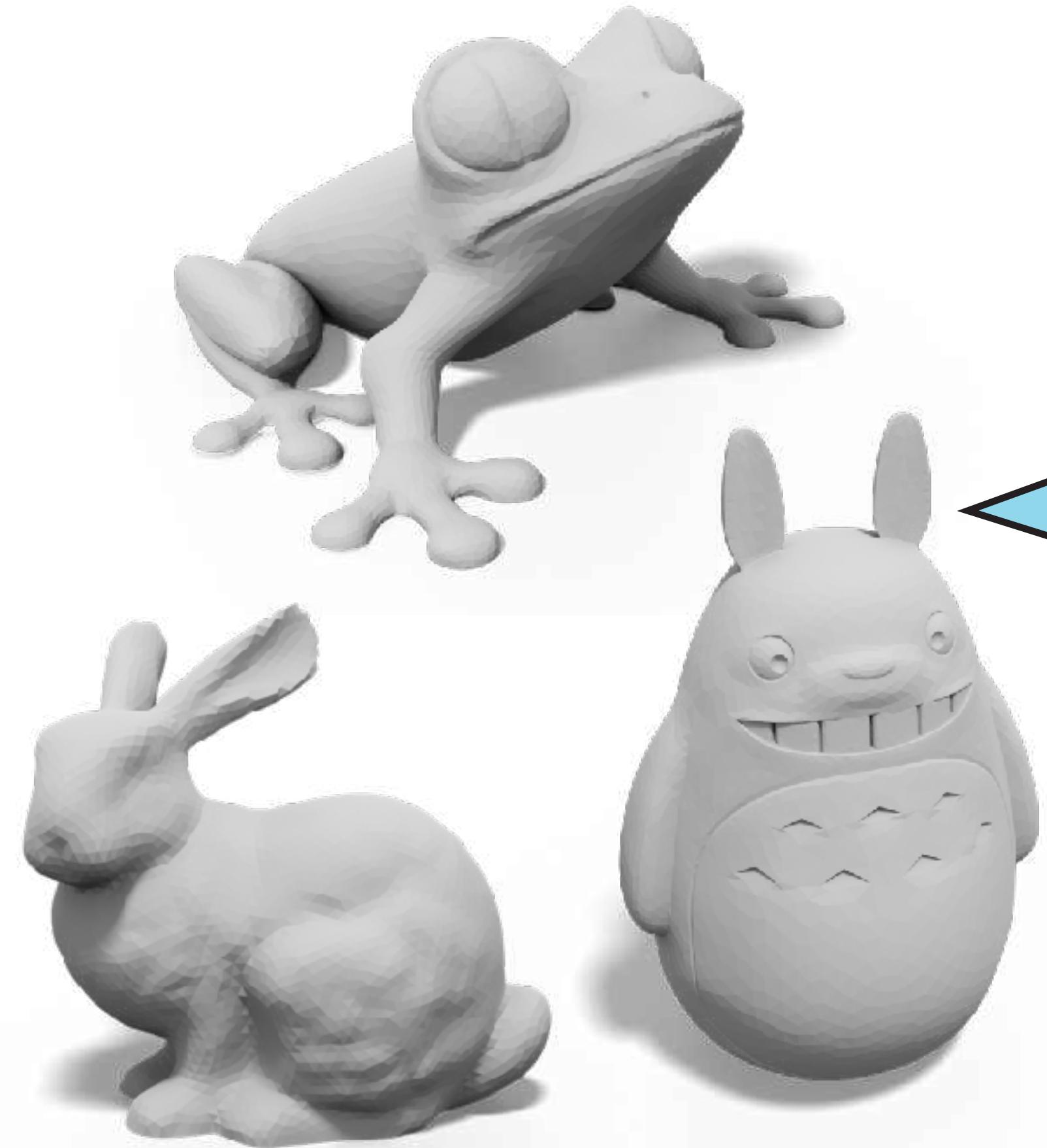


3D scanning



3D datasets

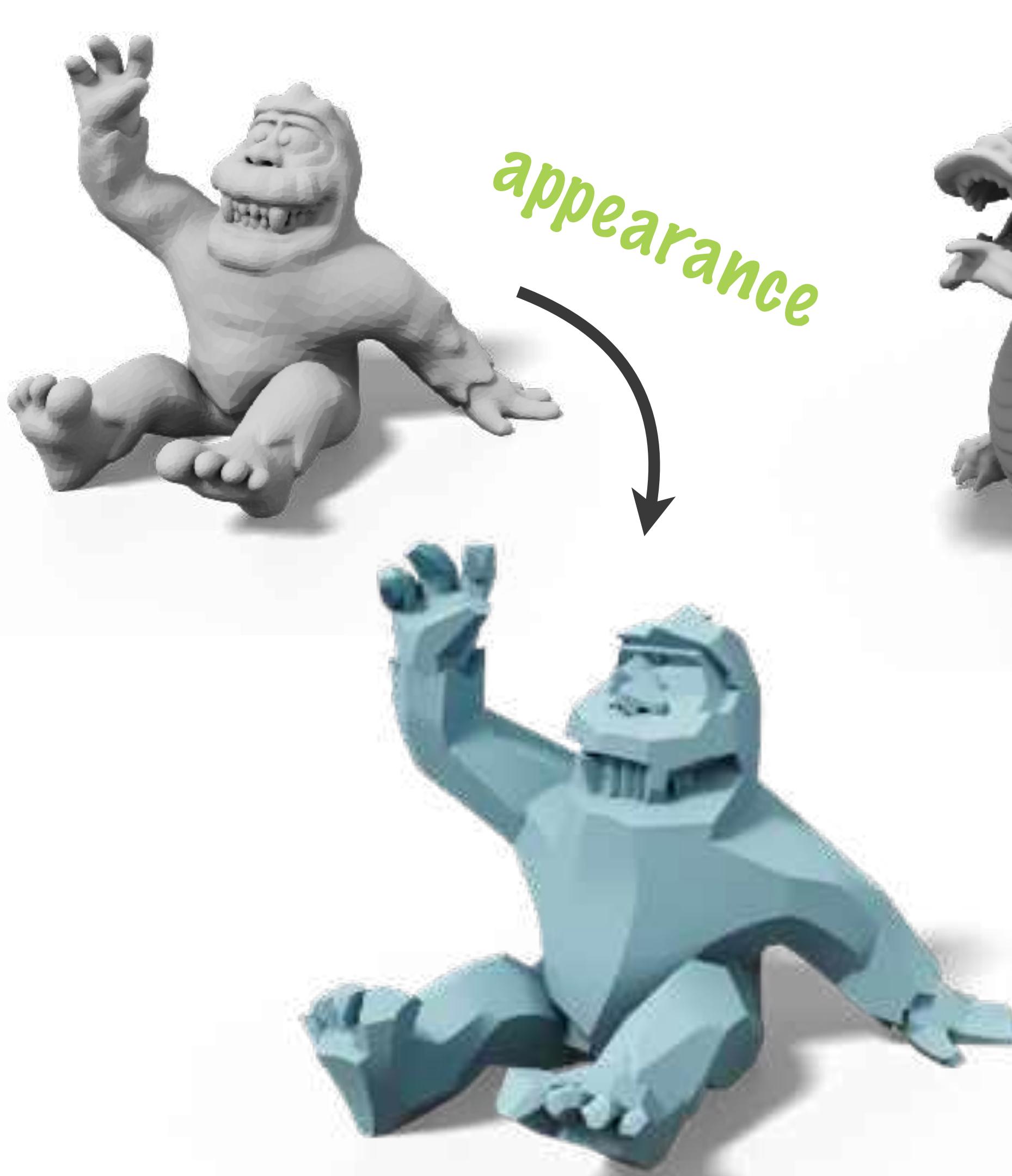
Future 3D Modeling



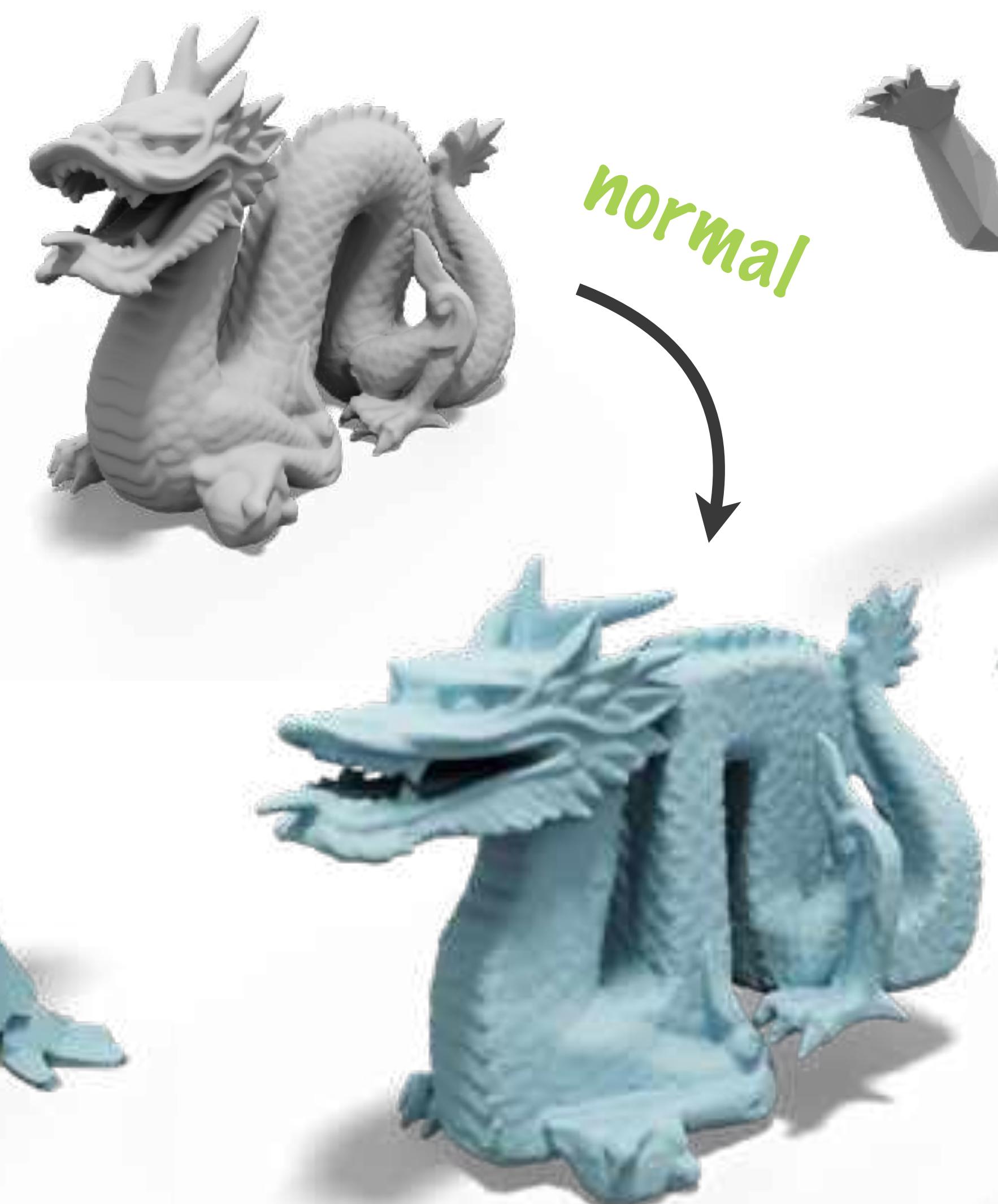
~~primitive shape~~
complex shape



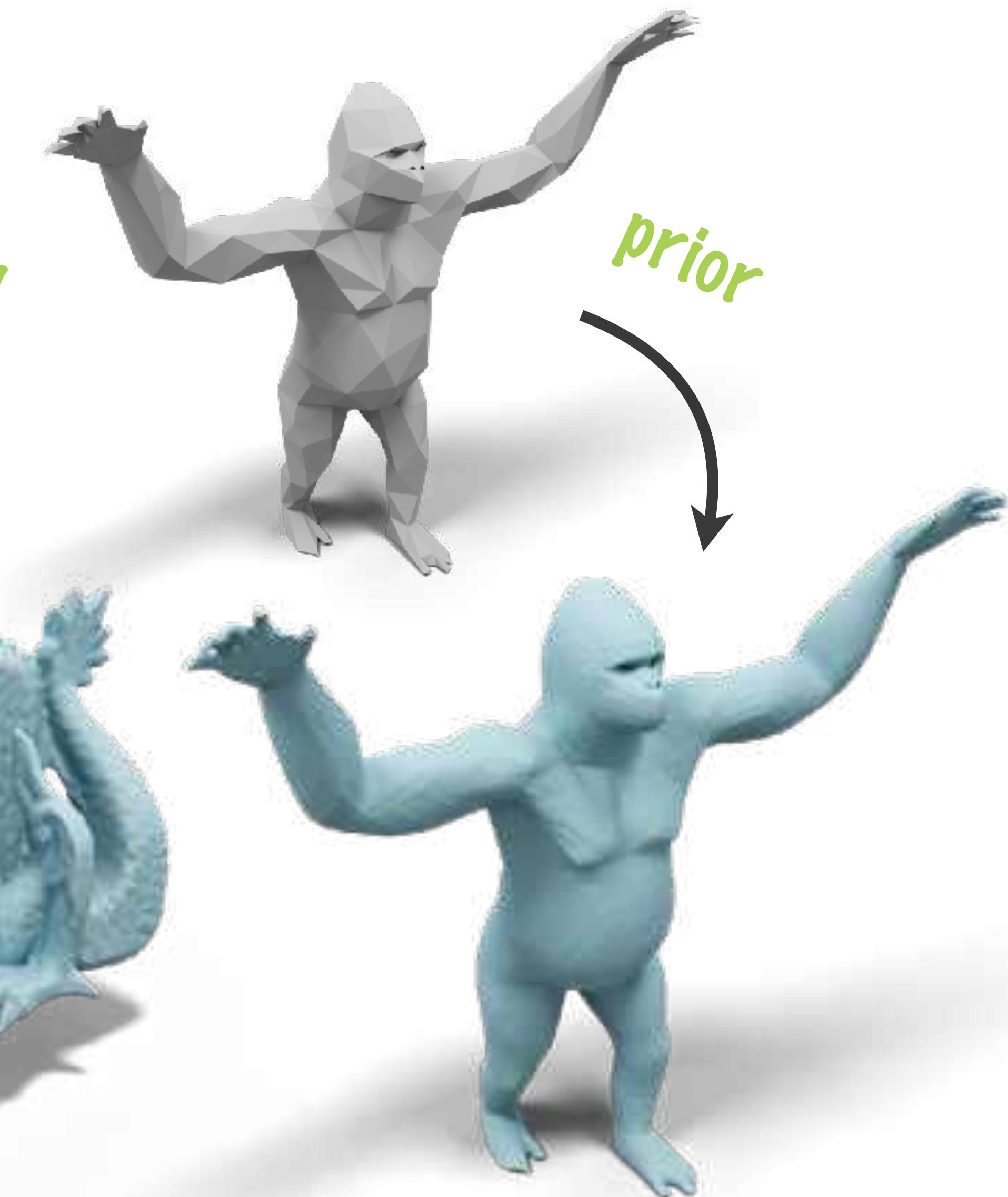
Geometric filtering

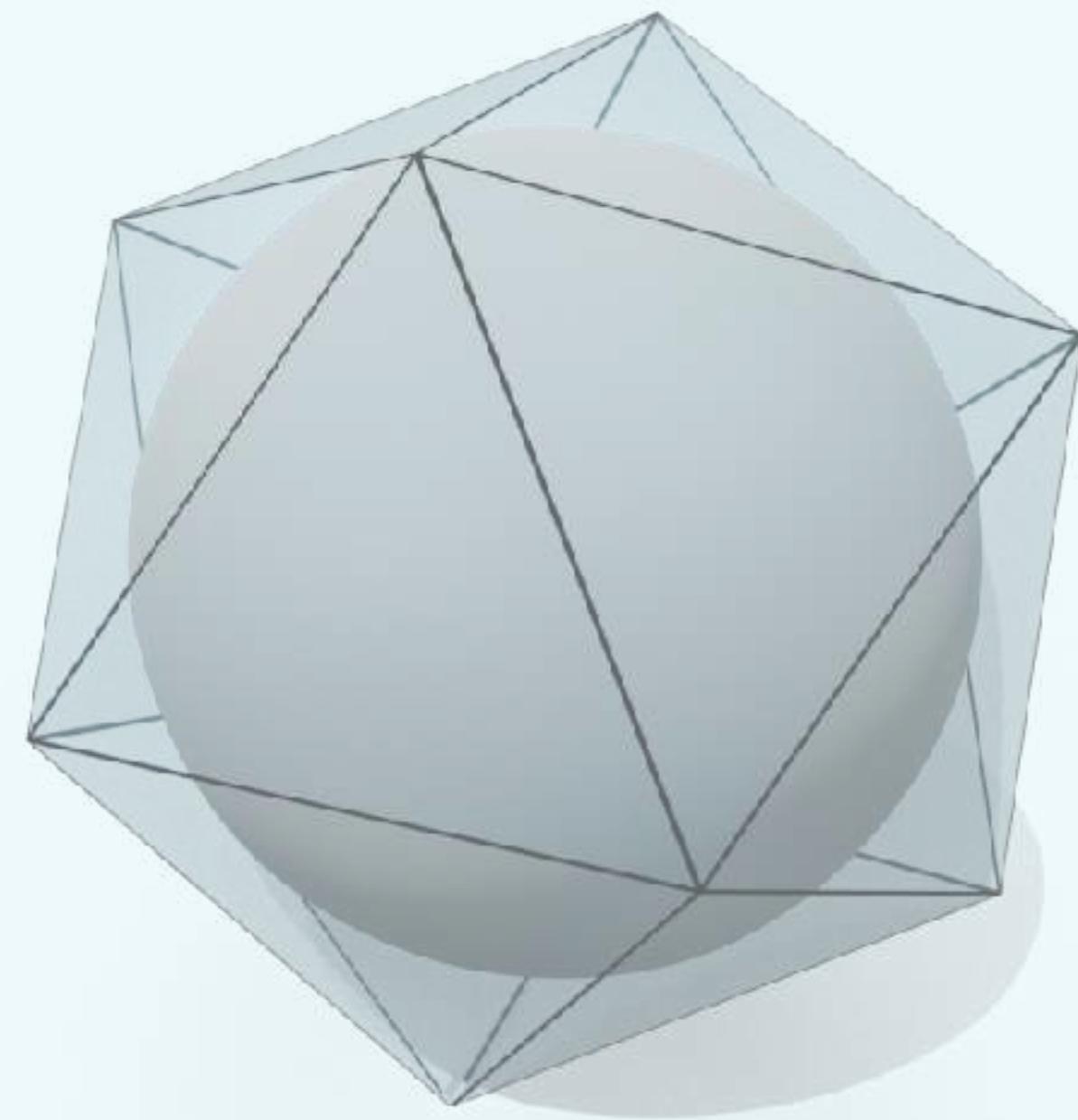


3D Stylization



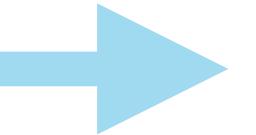
Data-driven subdivision



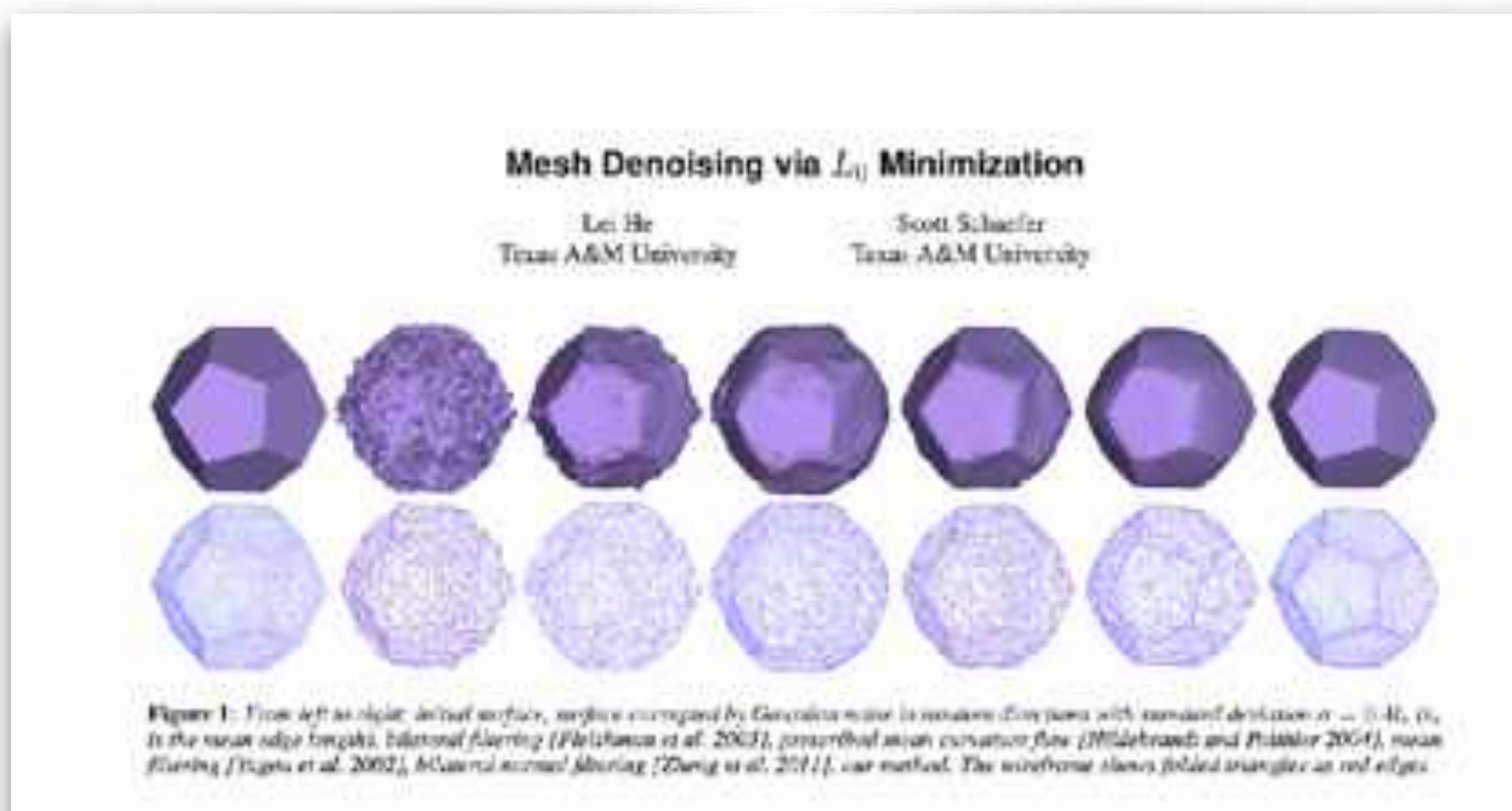
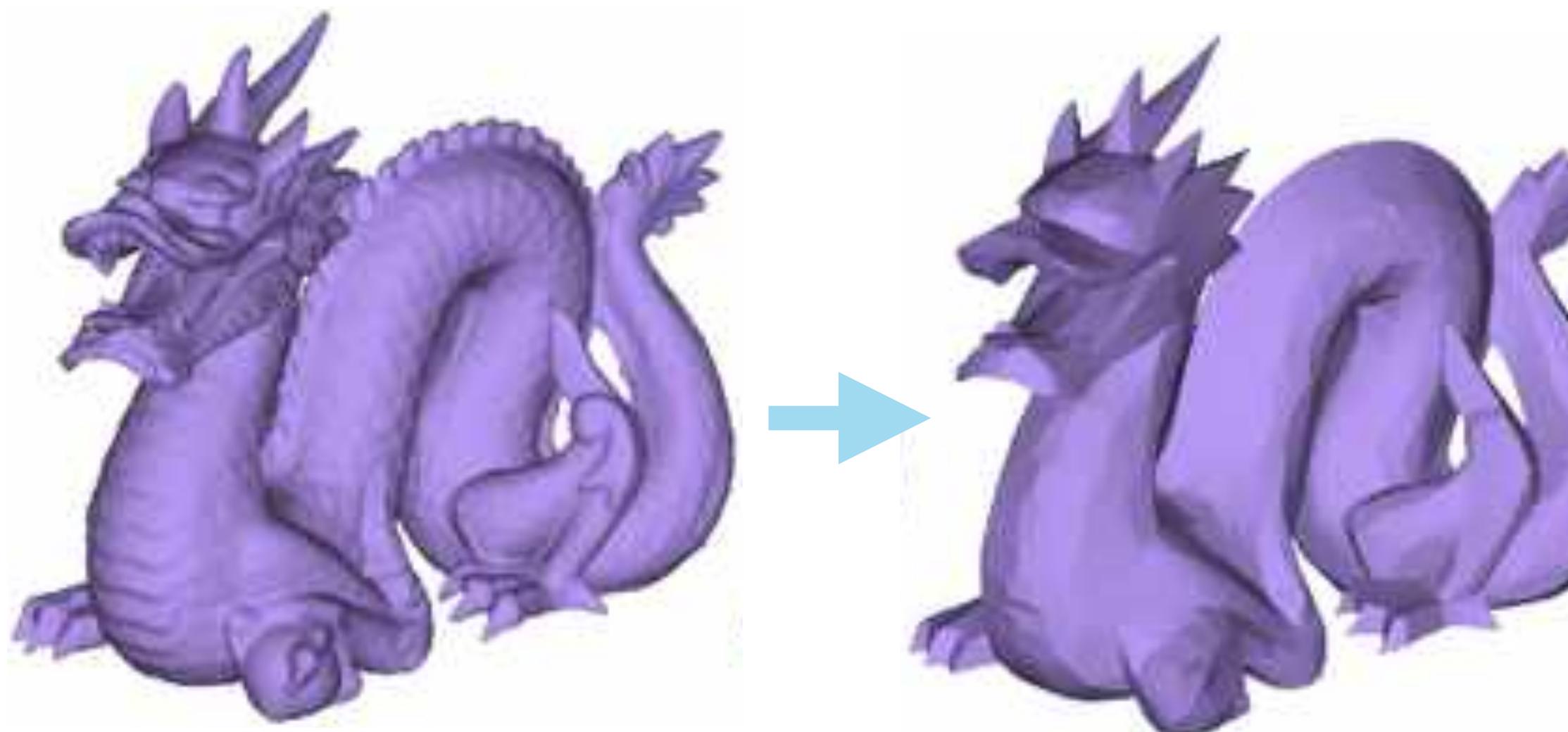


Geometric Filtering

Image Filters



Geometric Filters

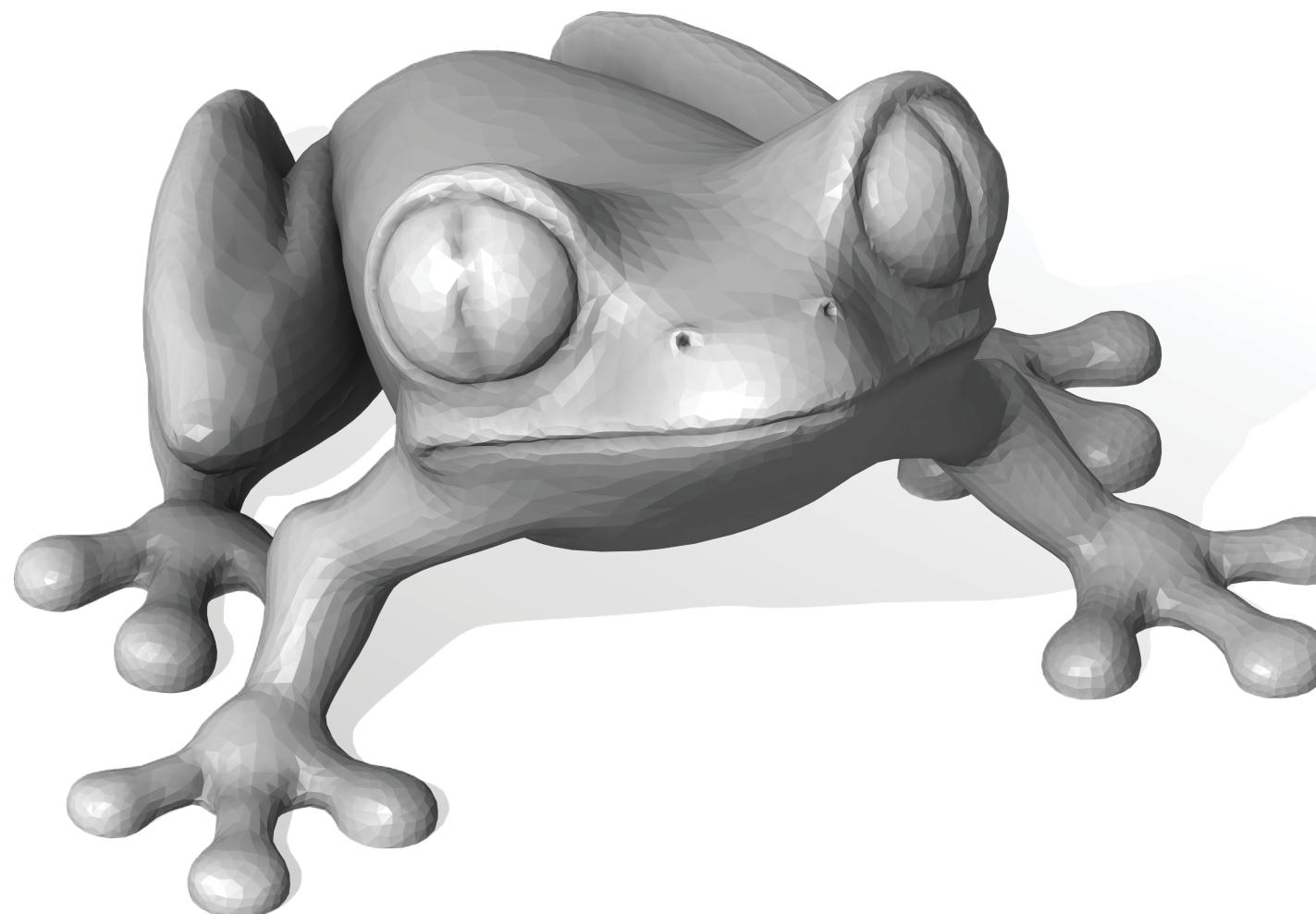


$$\min_{p, \delta} \|p - \delta\|^2 + \beta|D(p) - \delta|^2 + \lambda|\delta|_0$$

$$\Delta_{2,3,4} = \frac{1}{2} |p_2 - p_3| / n$$

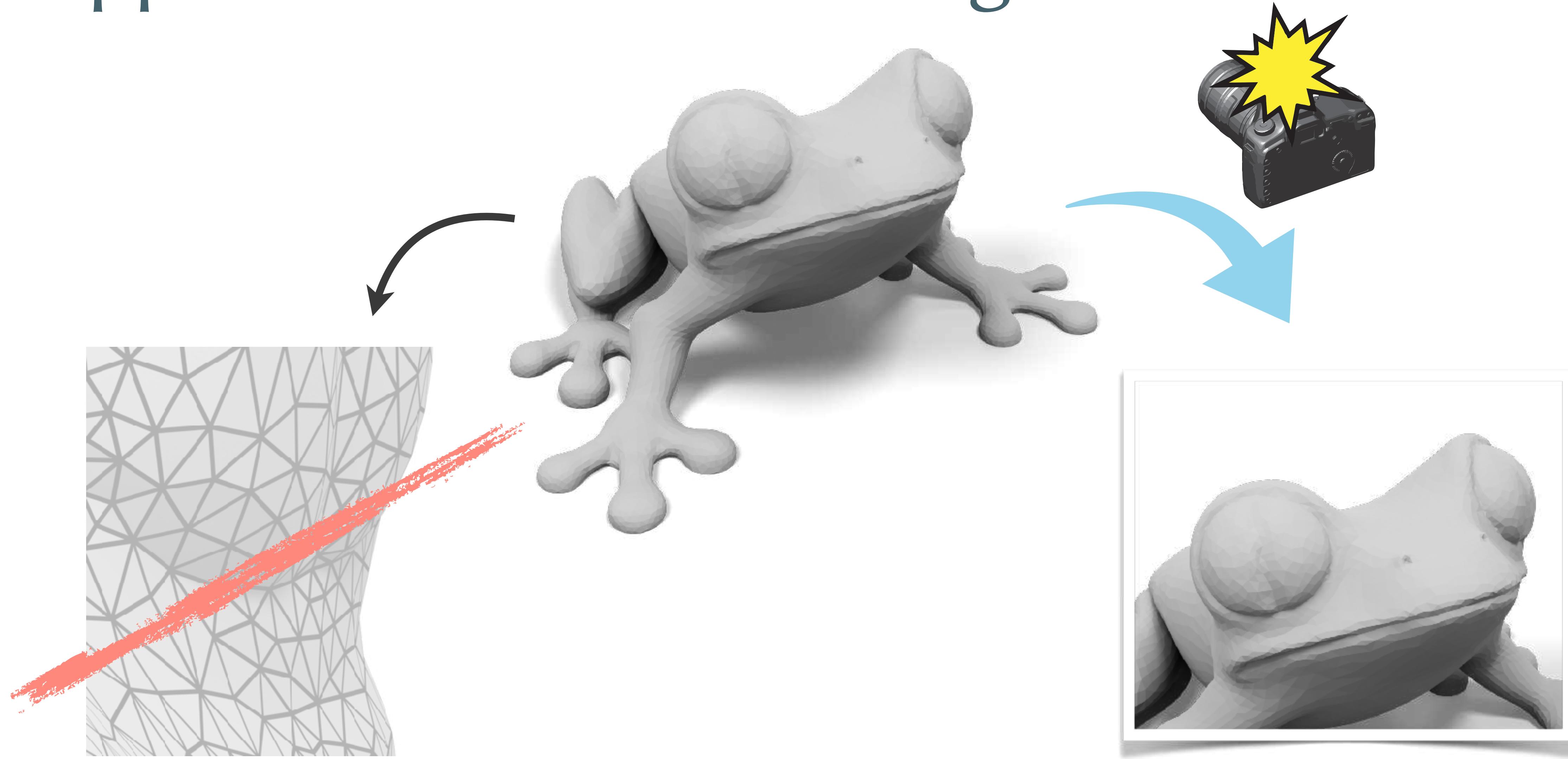
$$D(e) = \begin{bmatrix} \Delta_{1,2,3}((p_4-p_3)\cdot(p_3-p_1)) + \Delta_{1,3,4}((p_1-p_3)\cdot(p_3-p_2)) \\ |p_3-p_1|^2(\Delta_{1,2,3} + \Delta_{1,3,4}) \\ \Delta_{1,3,4} \\ \Delta_{1,2,3} + \Delta_{1,3,4} \\ \Delta_{1,2,3}((p_2-p_1)\cdot(p_1-p_3)) \\ |p_3-p_1|^2(\Delta_{1,2,3} + \Delta_{1,3,4}) \\ \Delta_{1,2,3} \\ \Delta_{1,2,3} + \Delta_{1,3,4} \end{bmatrix}^T \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix}$$

Complex Filters?

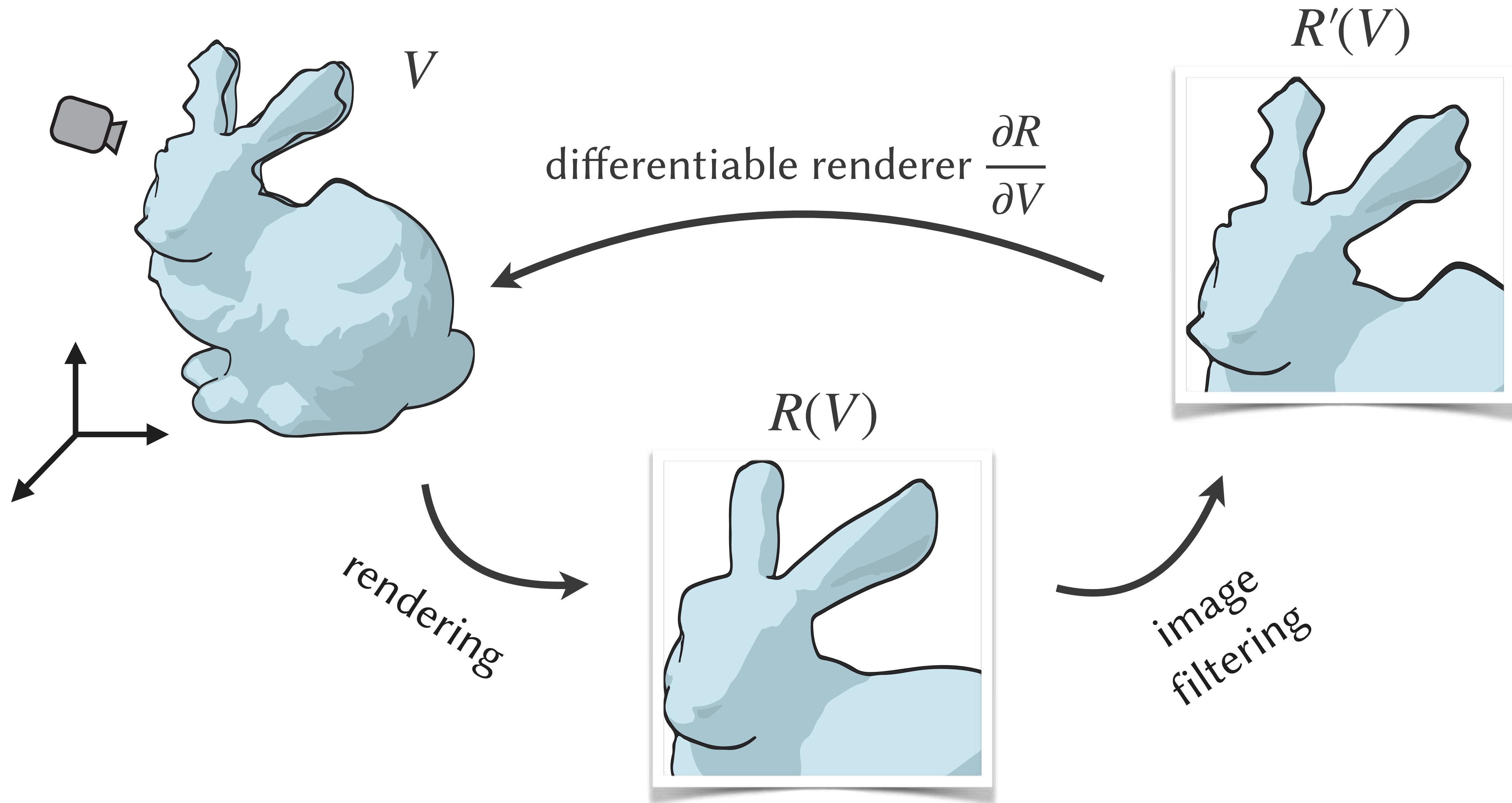


[Gatys et al. 2016]

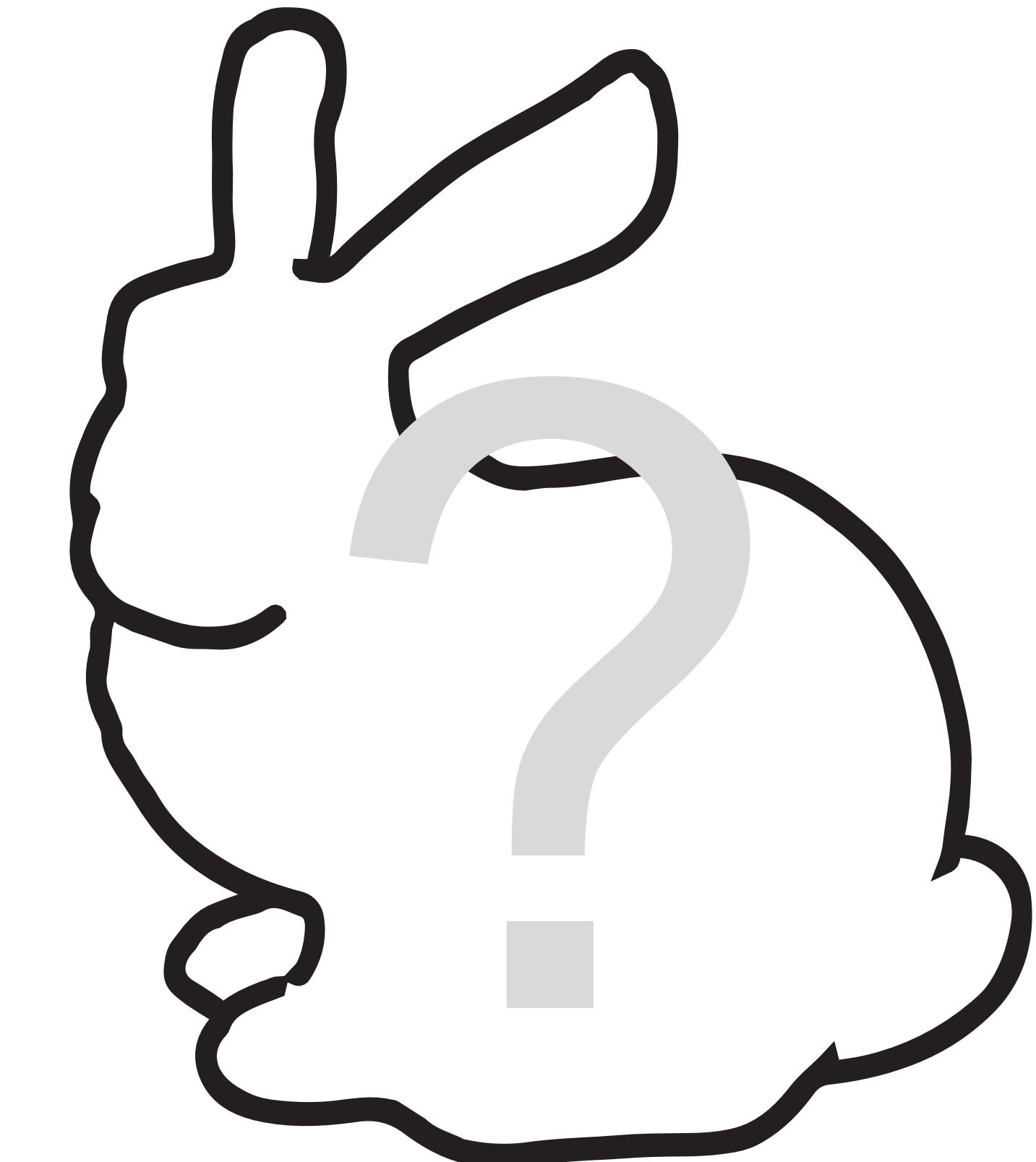
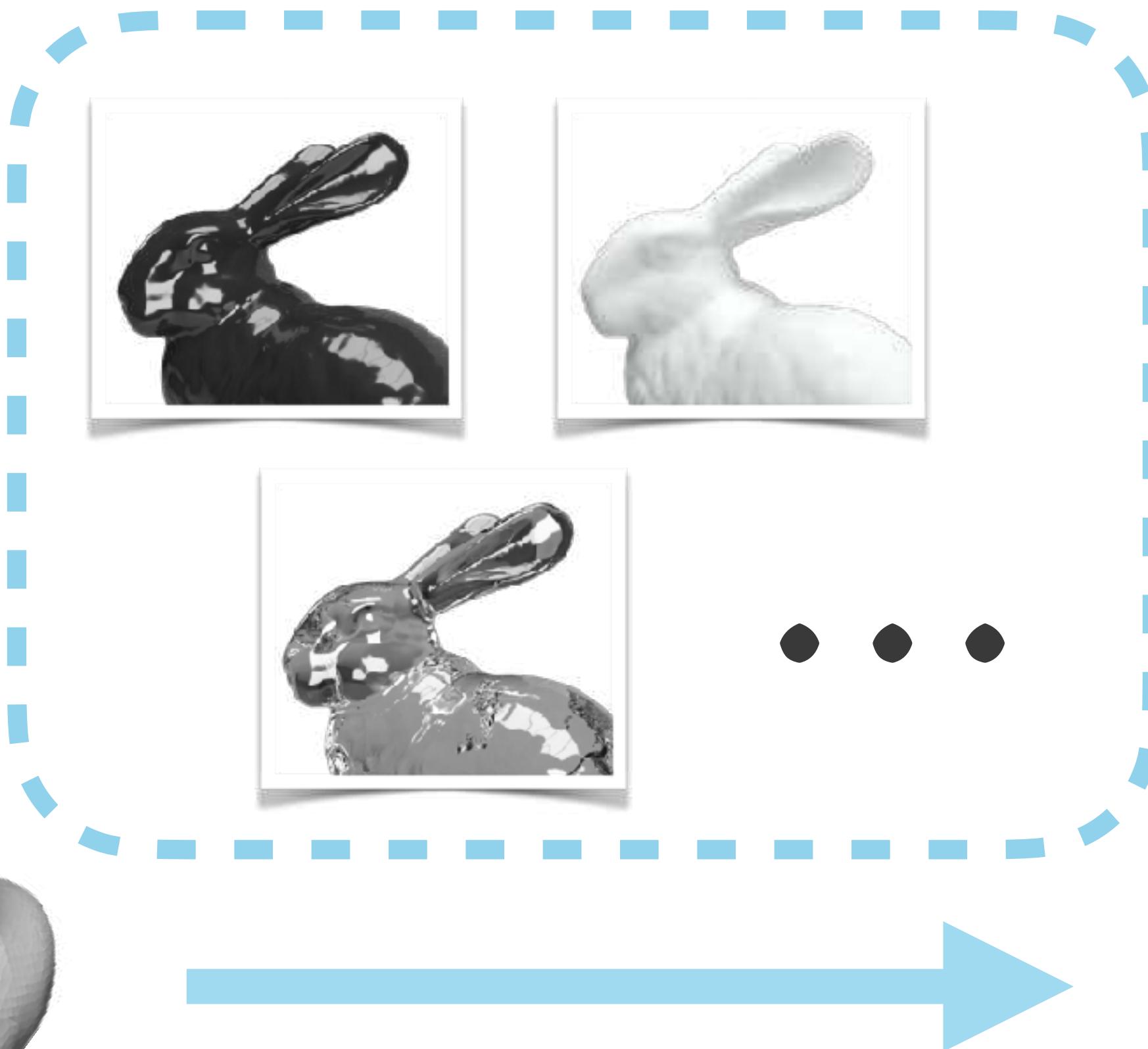
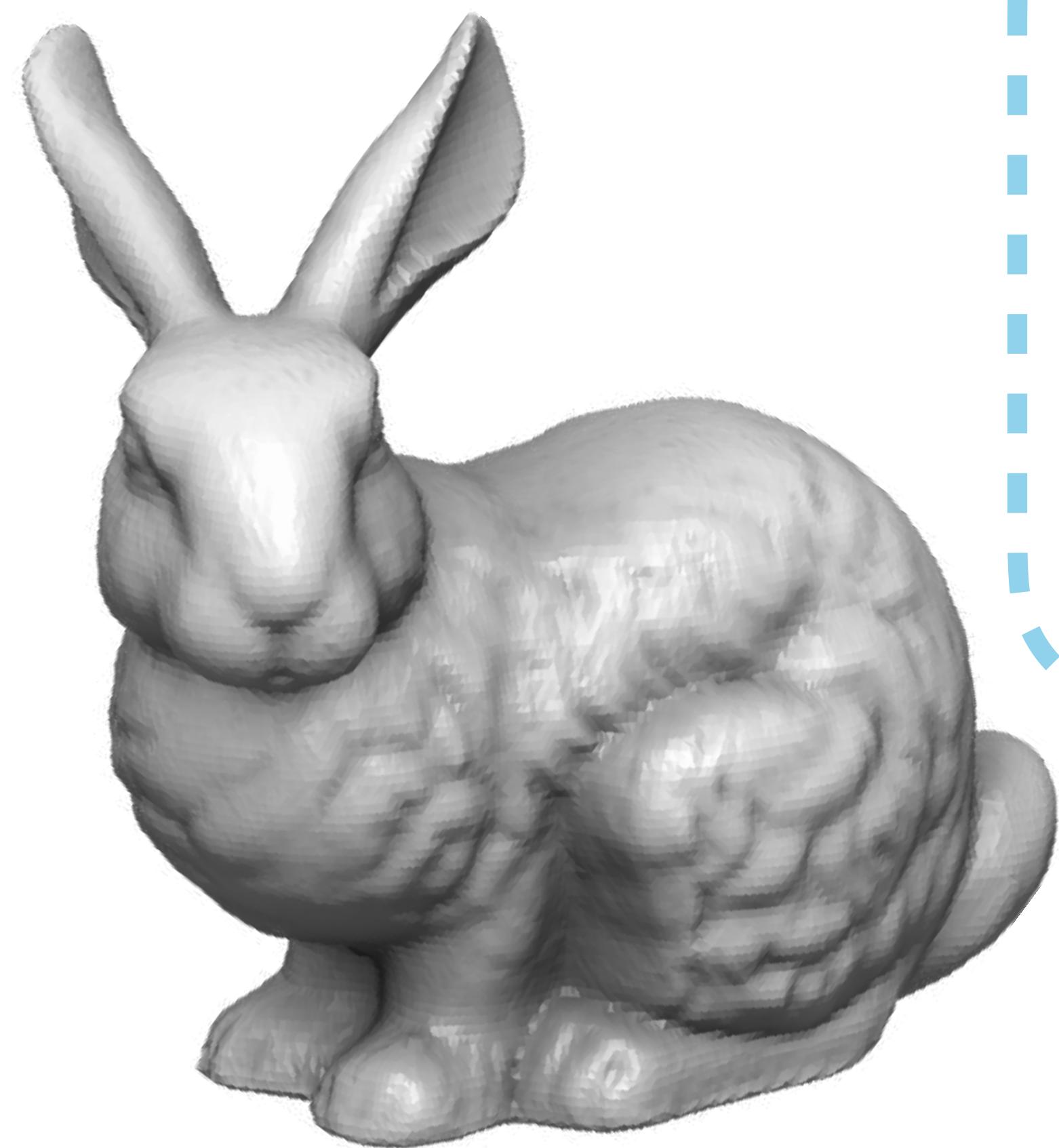
Appearance-driven Editing



Appearance-driven Editing

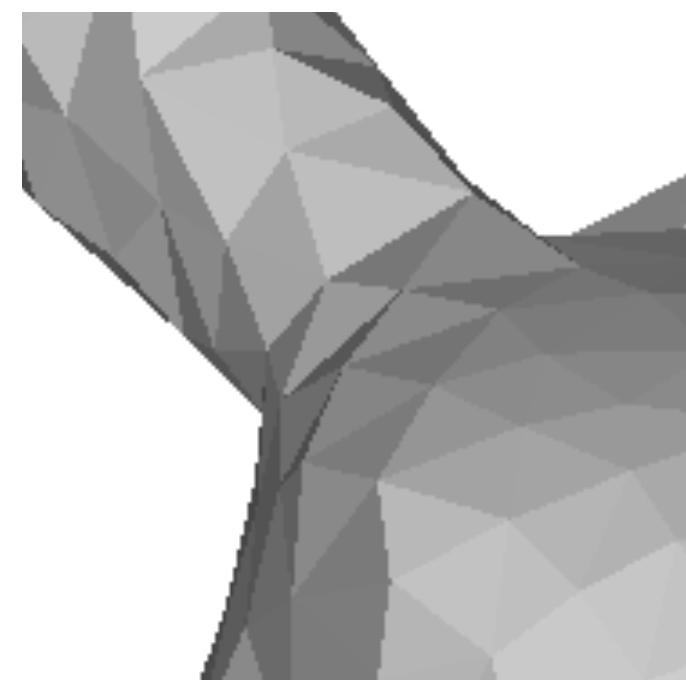


Rendering Means to an End

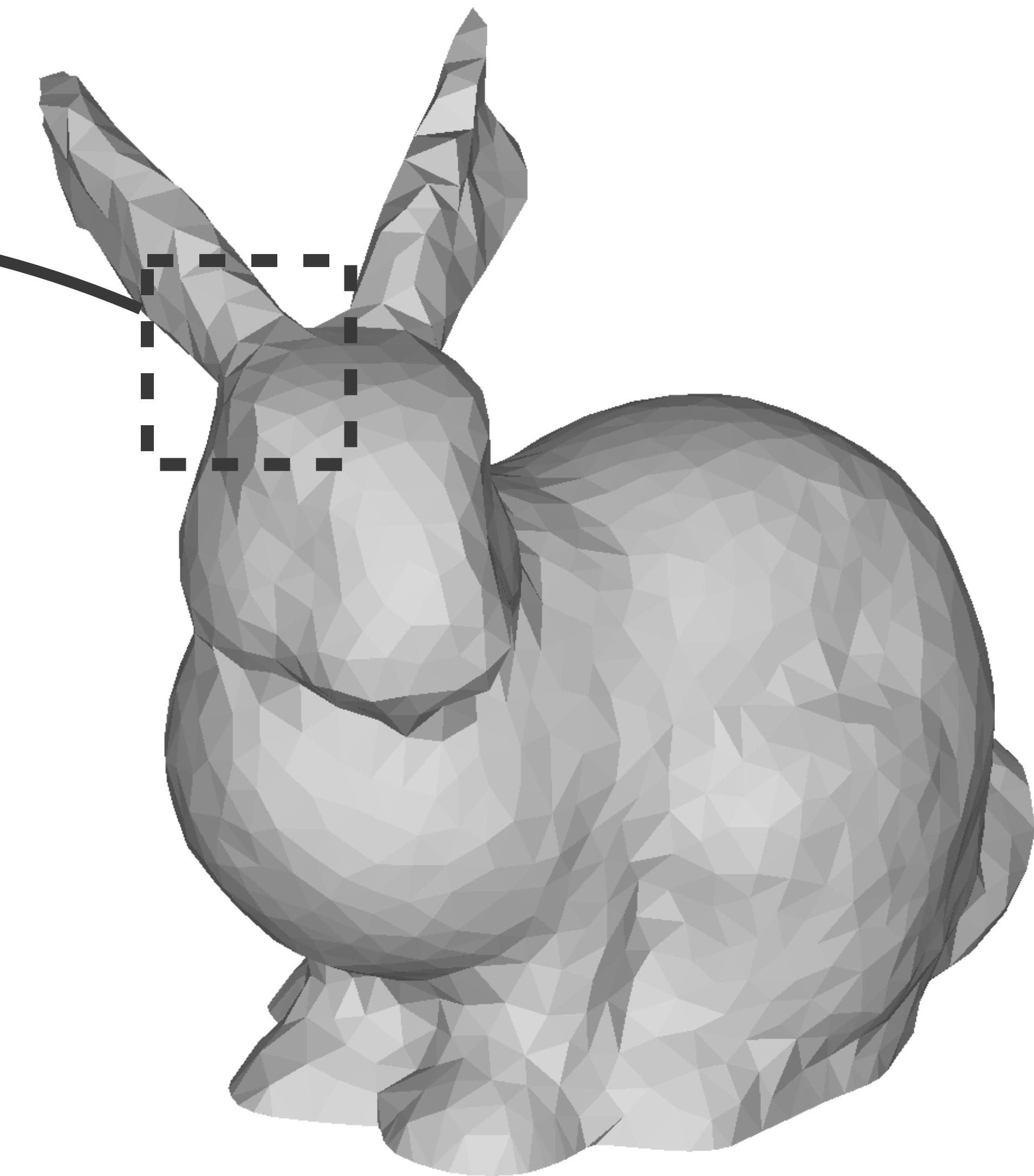
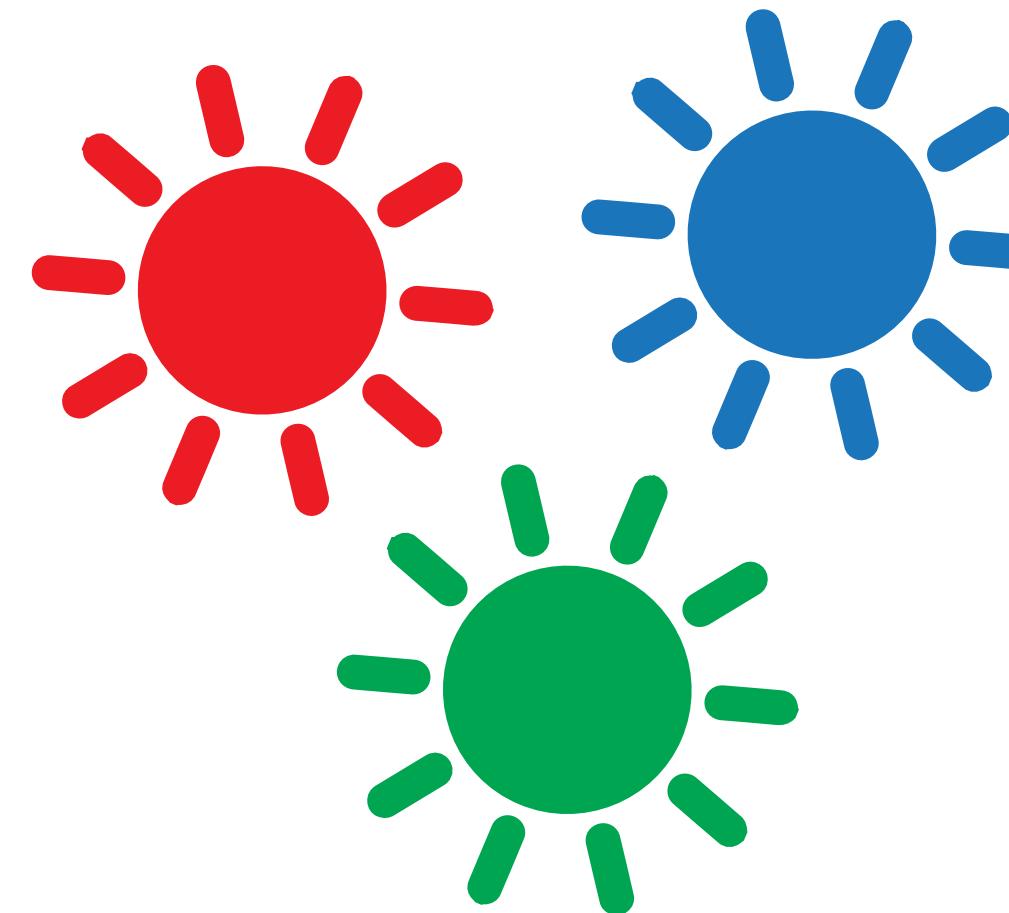


A Simple Differentiable Renderer

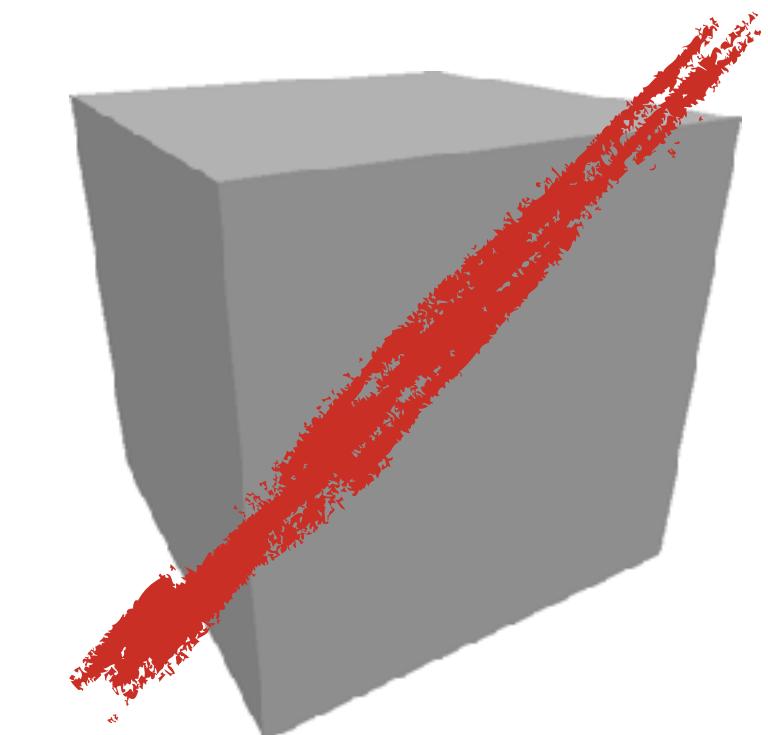
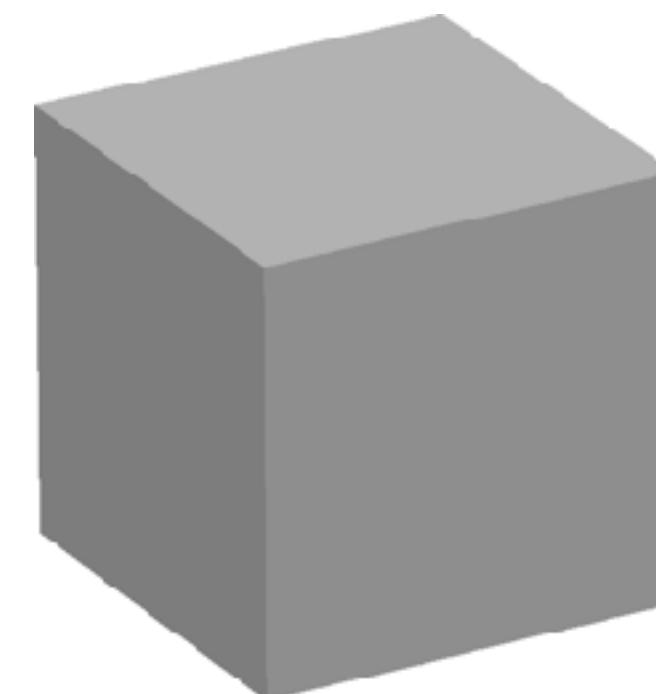
Flat shading



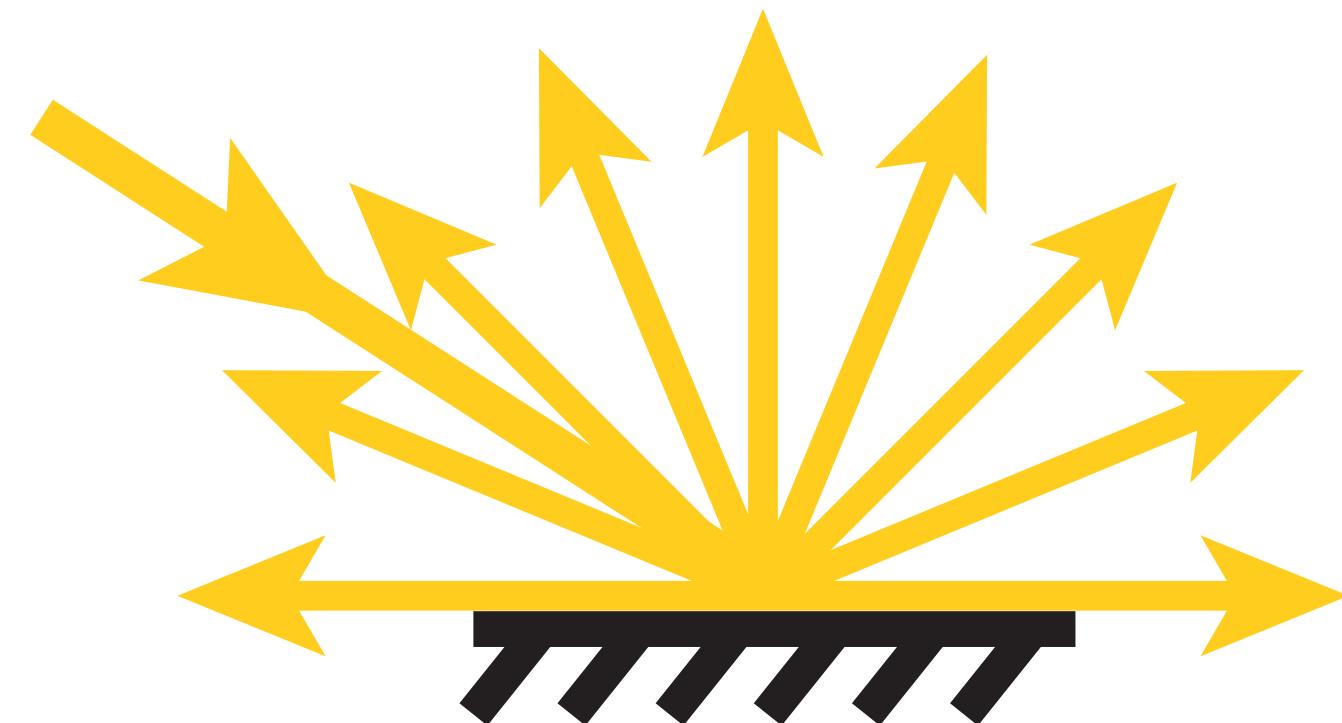
directional lights



orthographic camera



Lambertian material



A Simple Differentiable Renderer

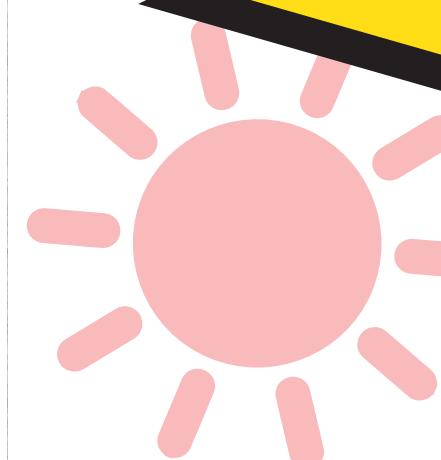
Flat shading



Fast



Simple



direc



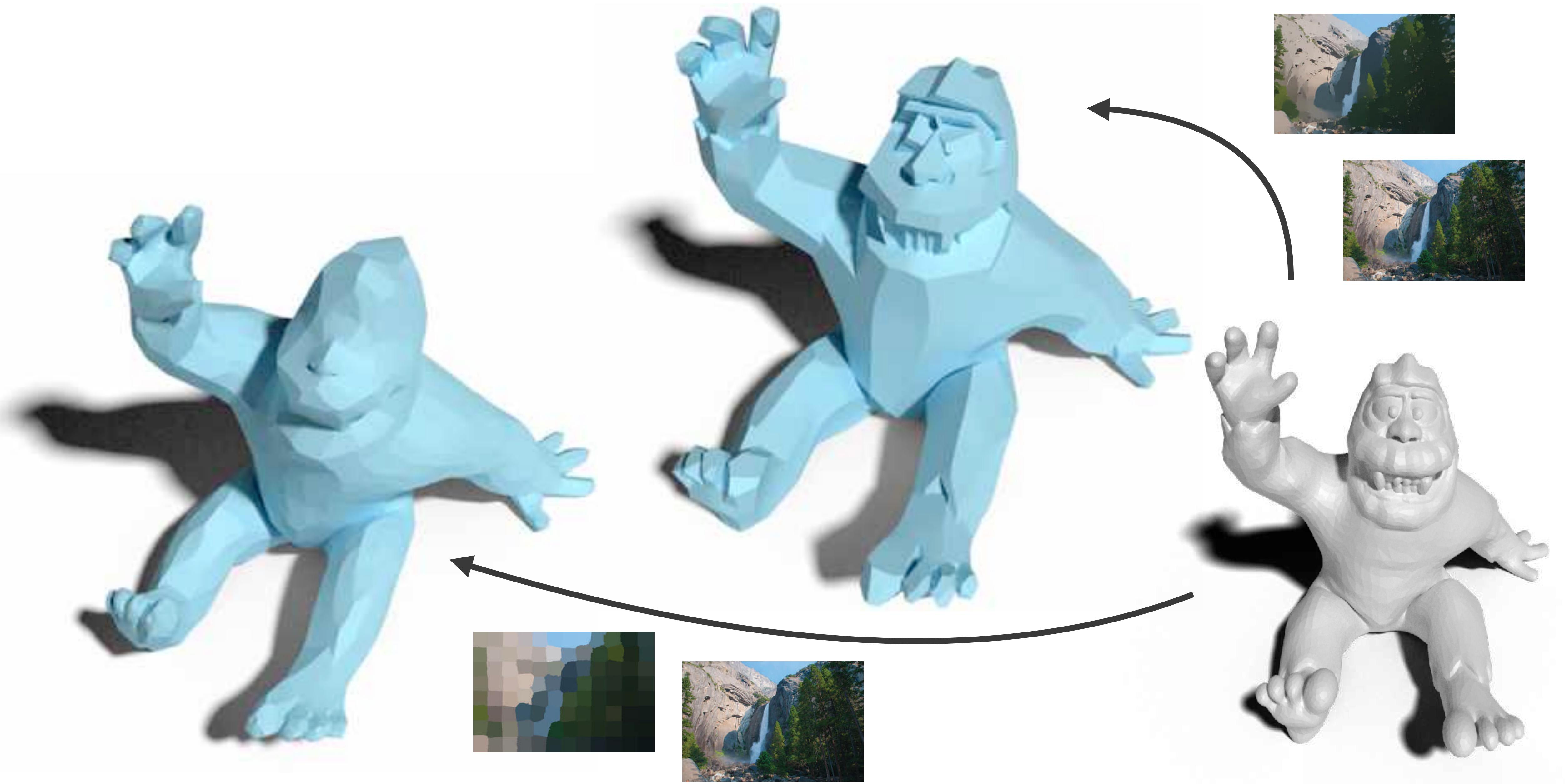
orthographic projection

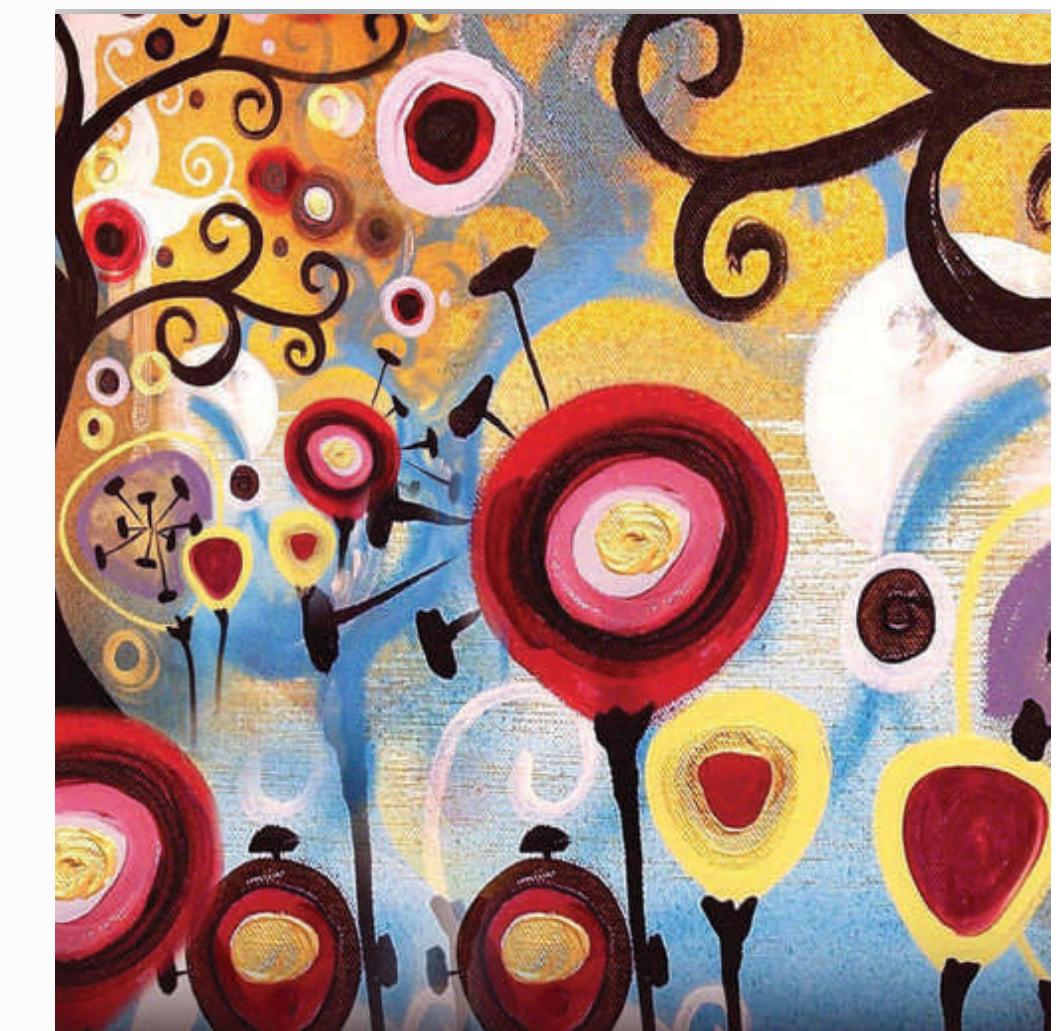
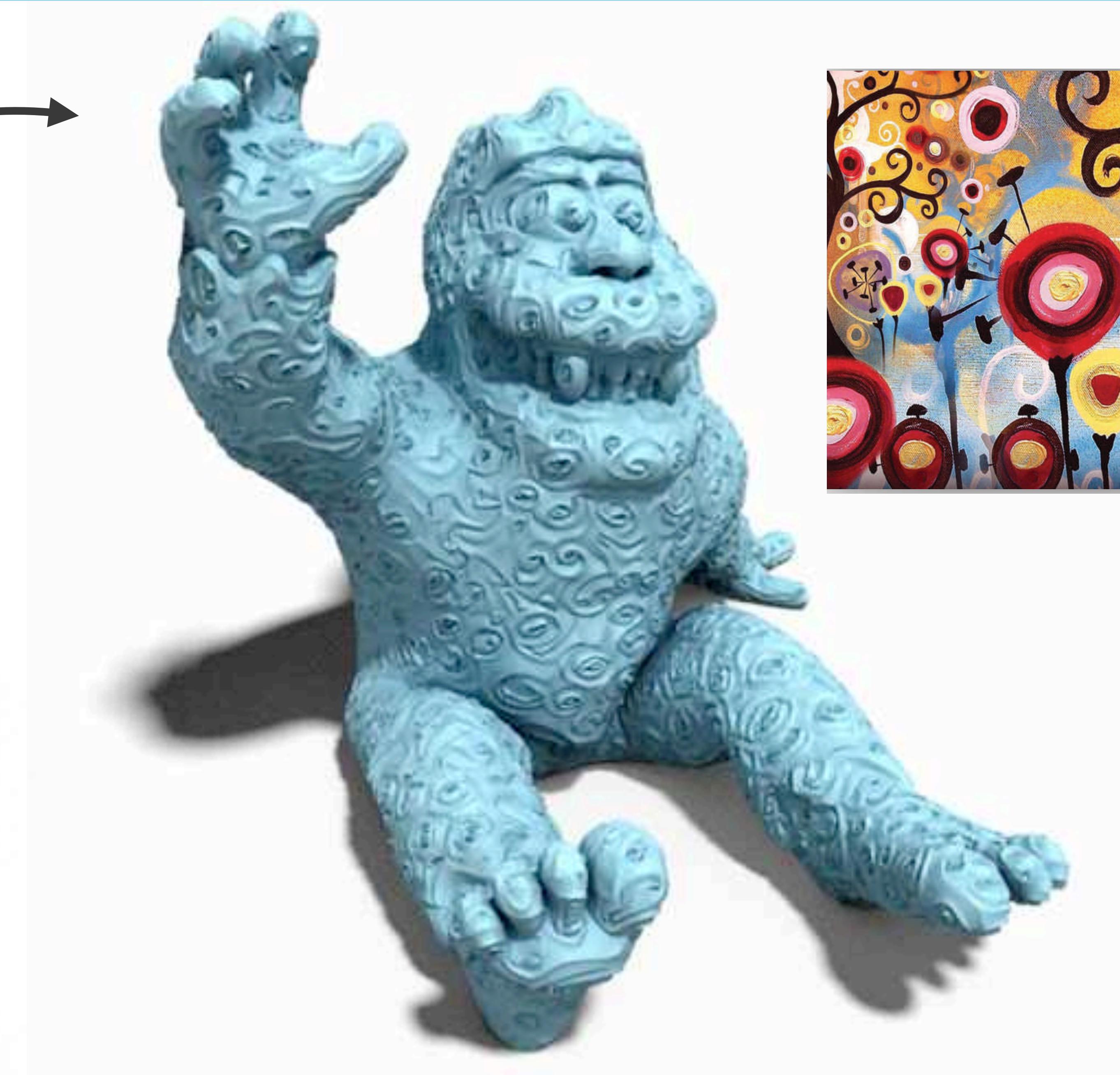


Differentiable

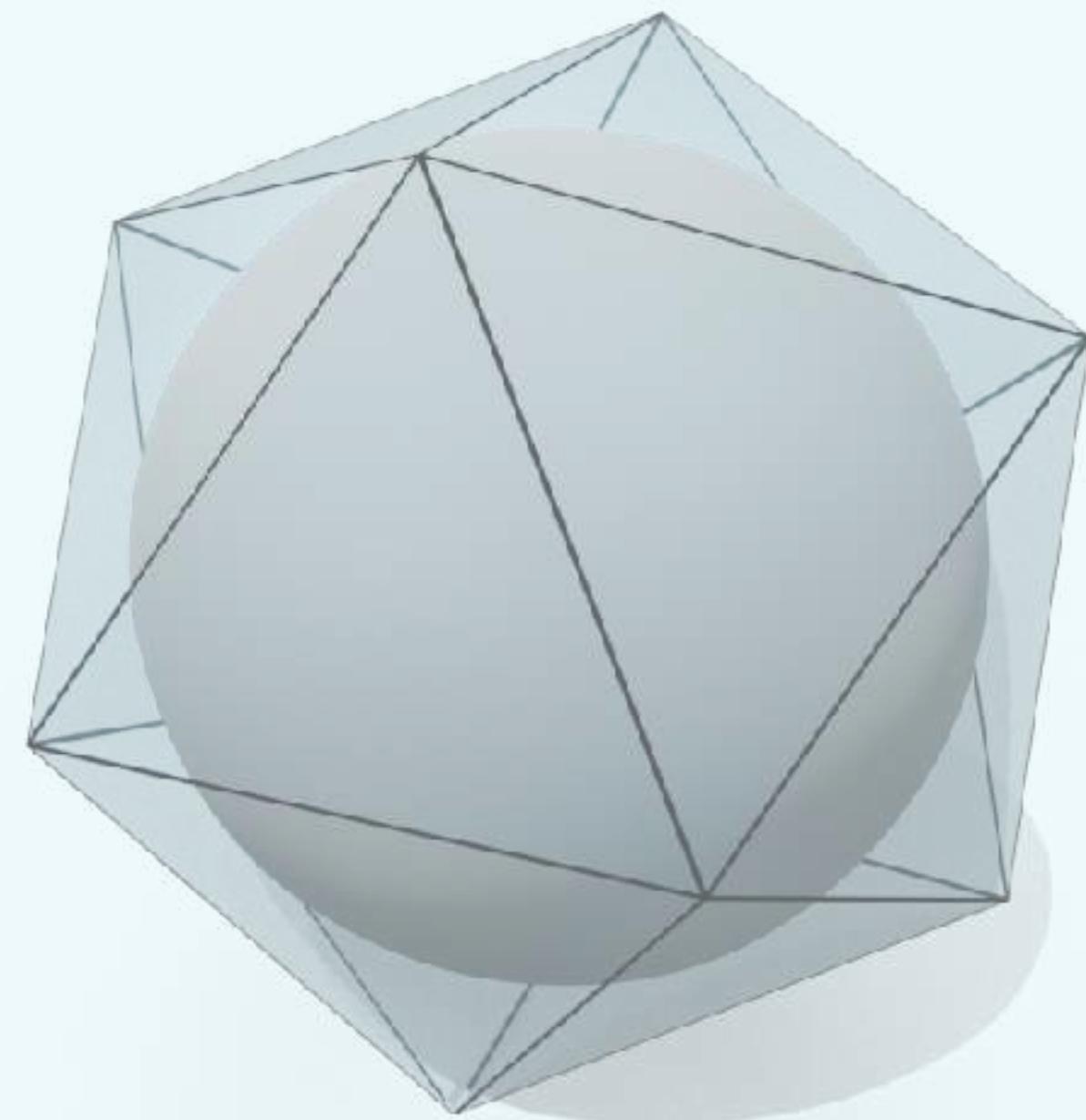


erial





~~Rendering as a tool for visualization~~
Rendering as a tool for 3D modeling



3D Stylization

Hsueh-Ti Derek Liu, Alec Jacobson, “Cubic Stylization”, SIGGRAPH Asia 2019

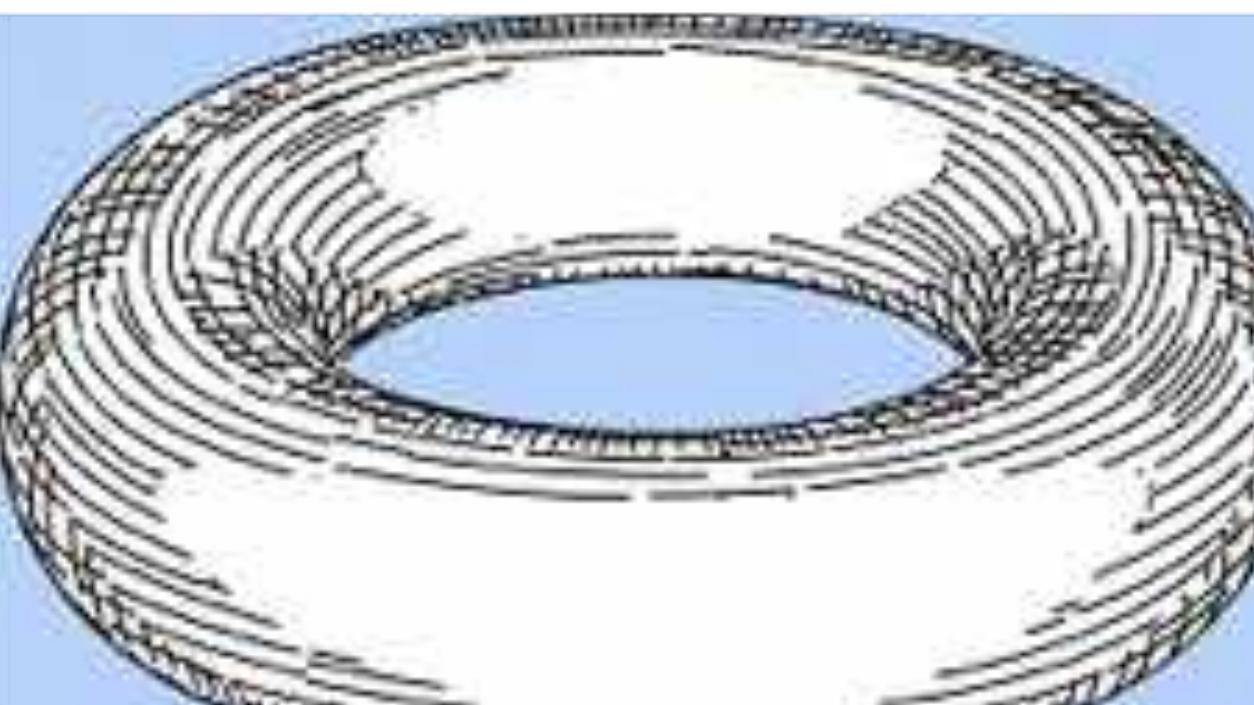
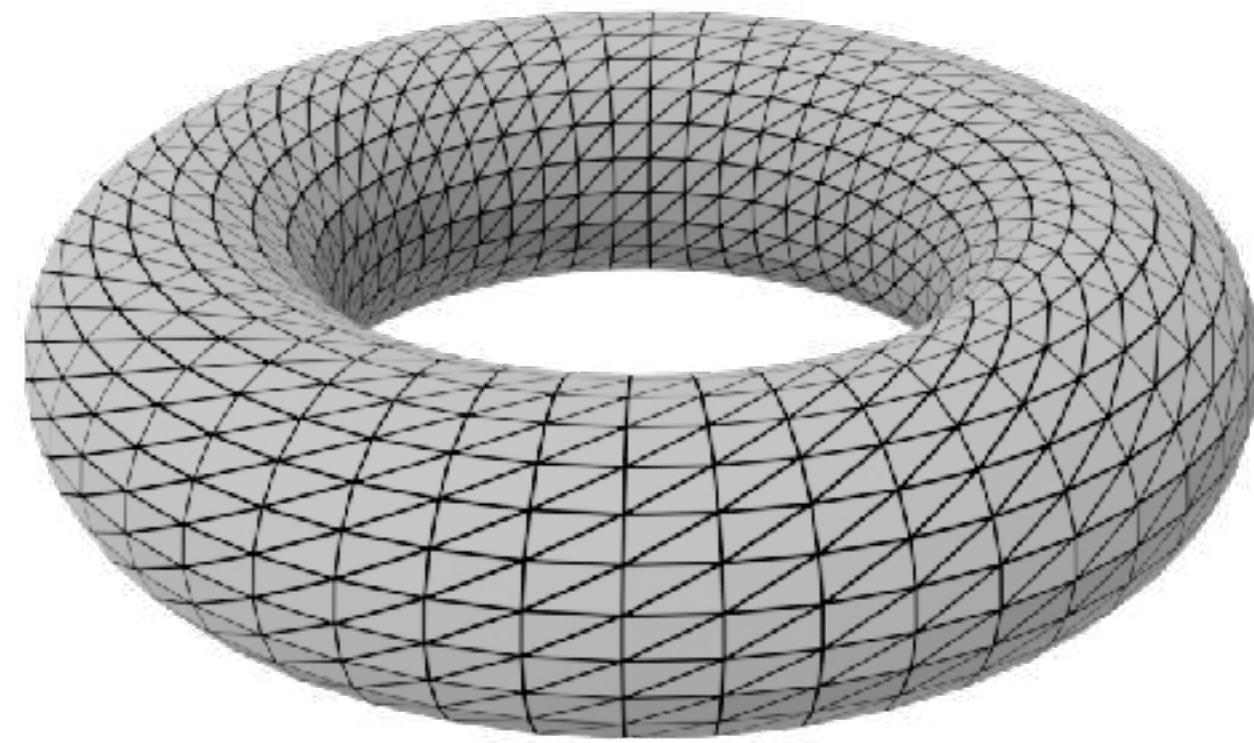
Stylization

2D → 2D



image stylization
[Gatys et al. 2016]

3D → 2D



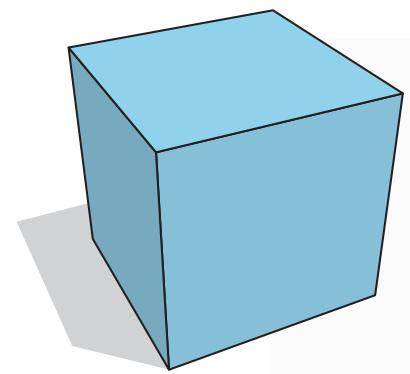
non-photorealistic rendering
[Hertzmann, Zorin 2000]

3D → 3D



3D stylization

Cubic



Stylization



Draped Seated Woman
by Henry Moore

© puffin11k



Chichén Itzá



Feathered Serpent

source: [wikimedia.org](#)



The Kiss by
Constantin Brâncuș
© Art Poskanzer



Taichi by Ju Ming

source: [wikimedia.org](#)

Block Statue
source: [wikimedia.org](#)



Draped Seated Woman
by Henry Moore
© puffin11k

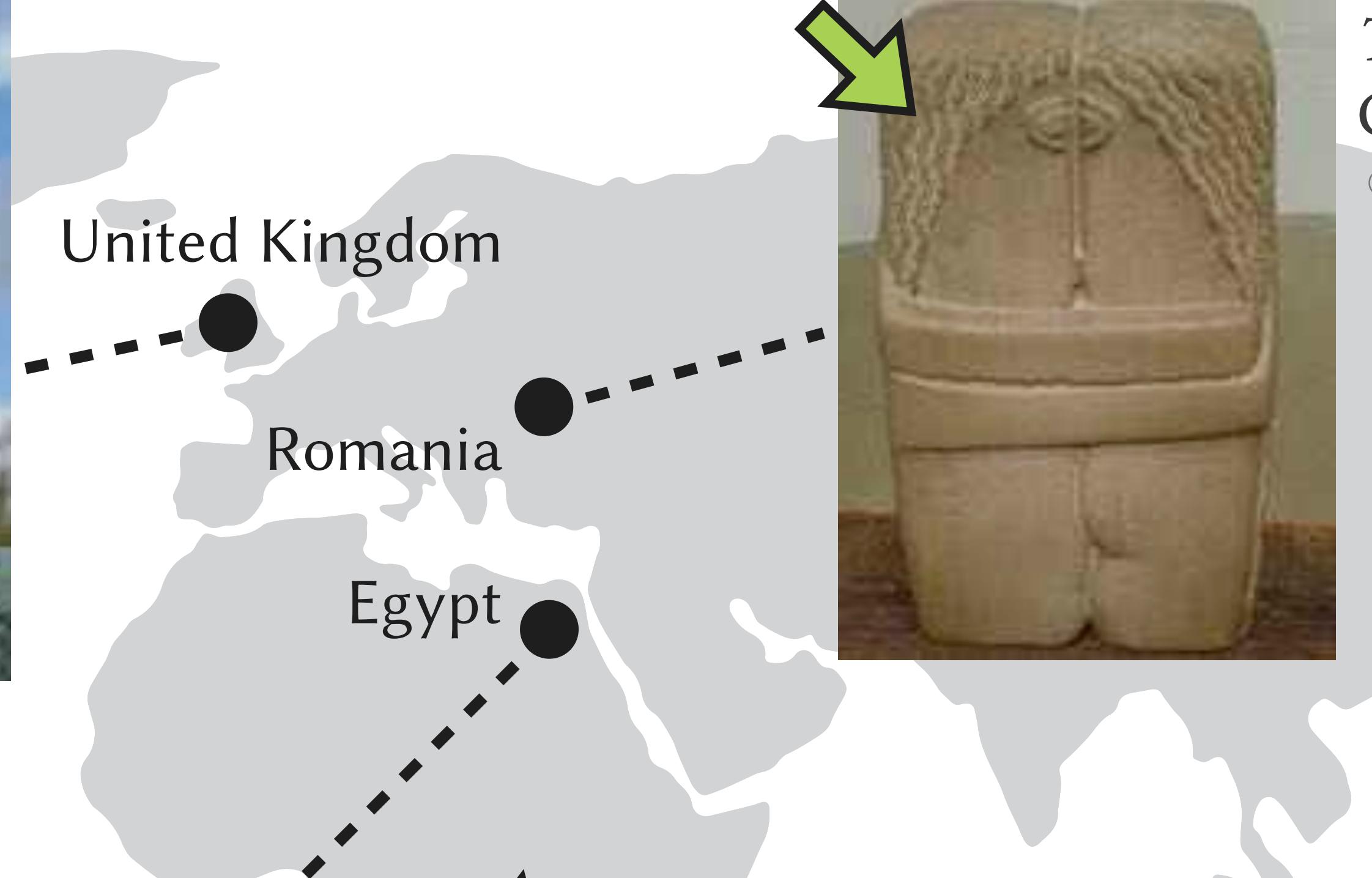


Chichén Itzá



Feathered Serpent

source: [wikimedia.org](#)



Block Statue

source: [wikimedia.org](#)

The Kiss by
Constantin Brâncuș
© Art Poskanzer



Taiwan

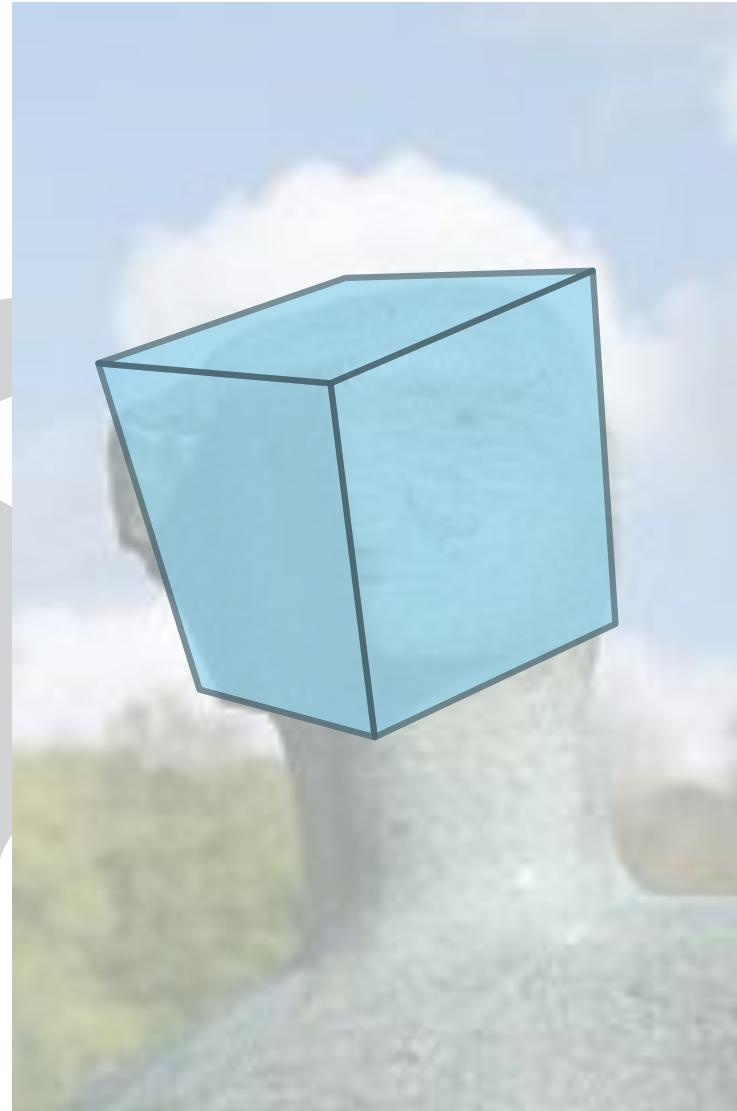


Taichi by Ju Ming

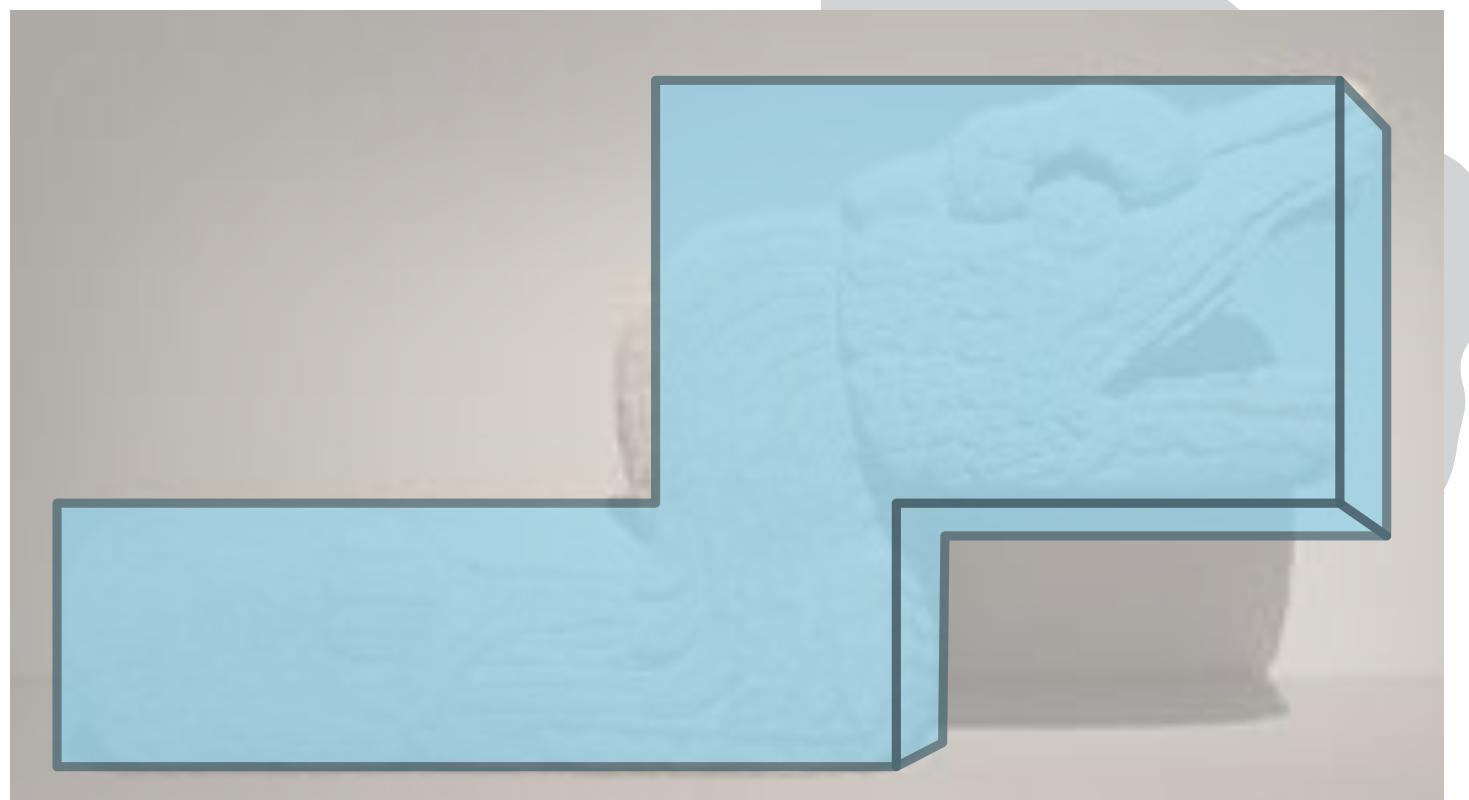
source: [wikimedia.org](#)

Draped Seated Woman
by Henry Moore

© puffin11k



Chichén Itzá



Feathered Serpent

source: [wikimedia.org](#)



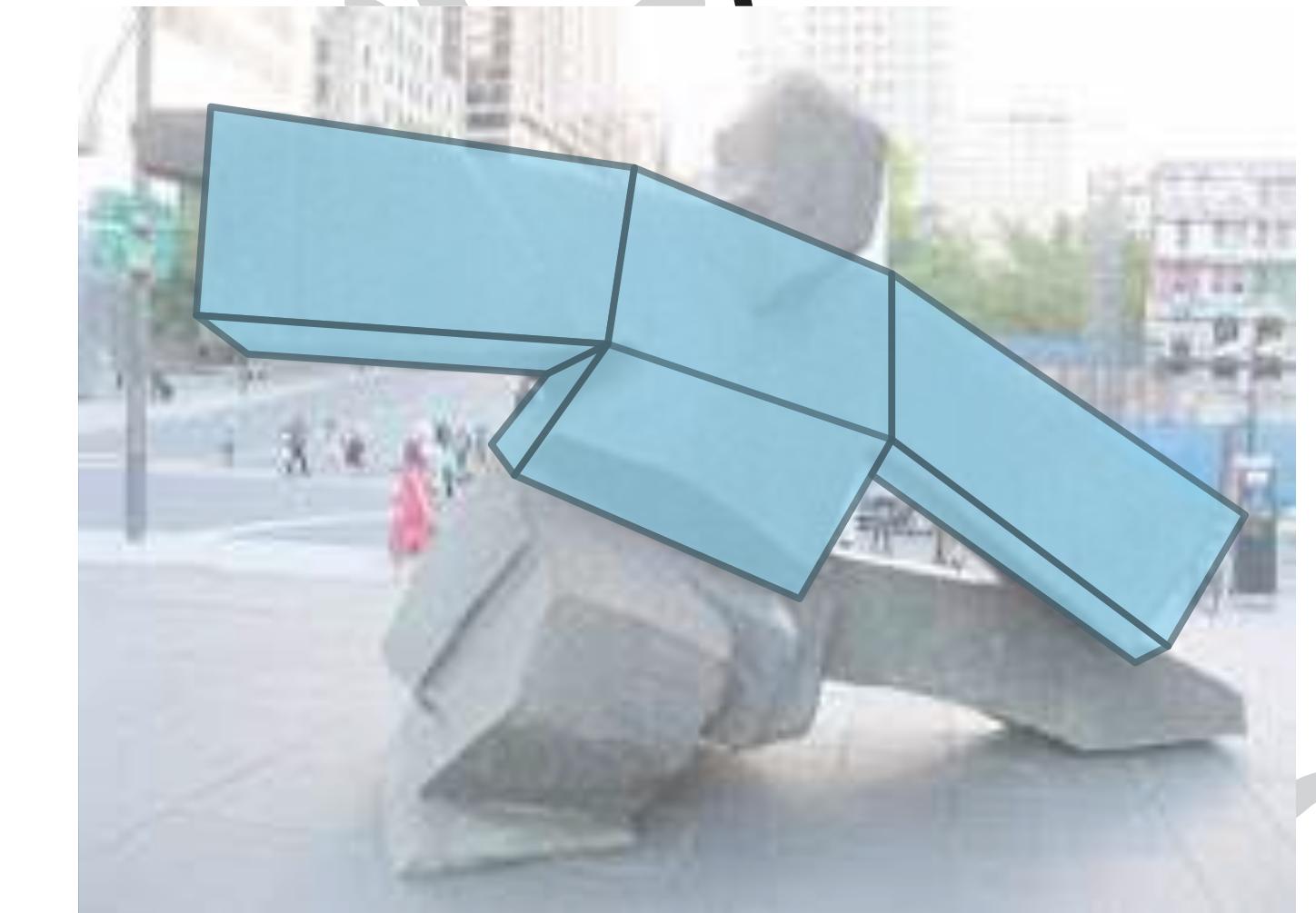
The Kiss by
Constantin Brâncuși

© Art Poskanzer



Block Statue

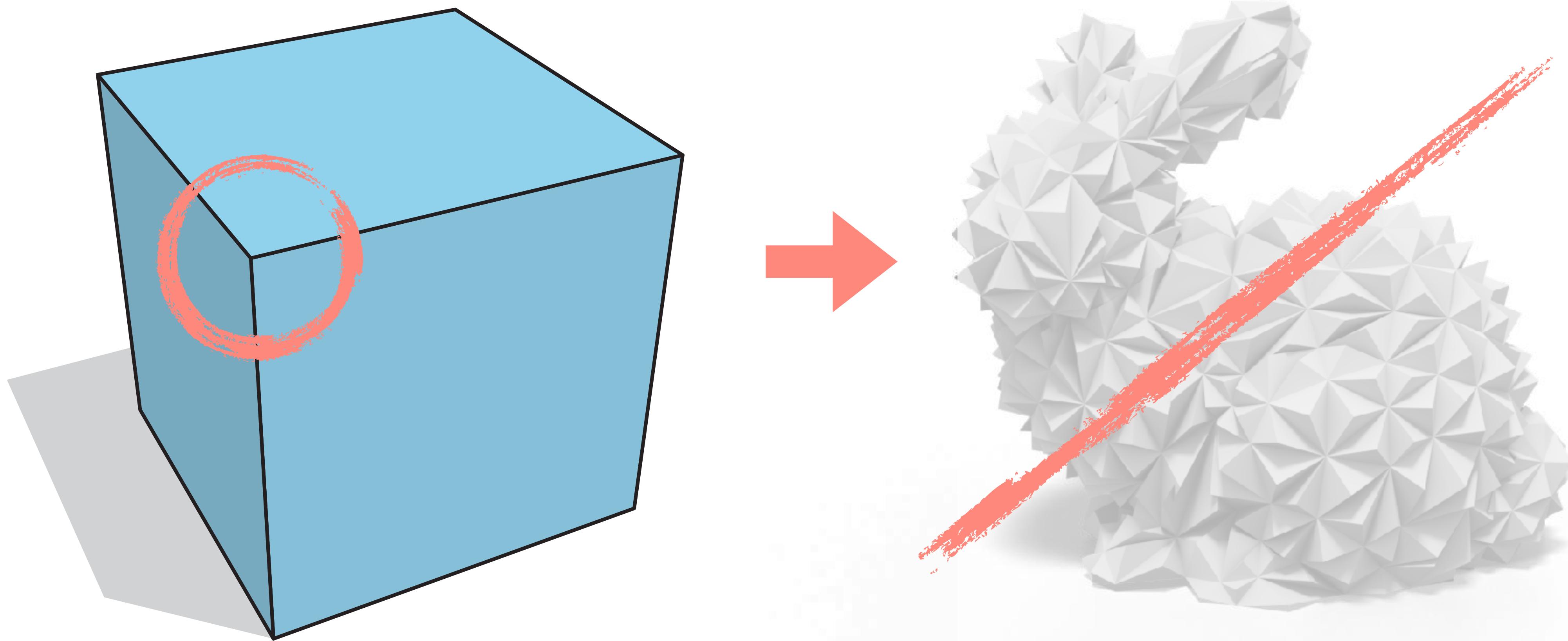
source: [wikimedia.org](#)



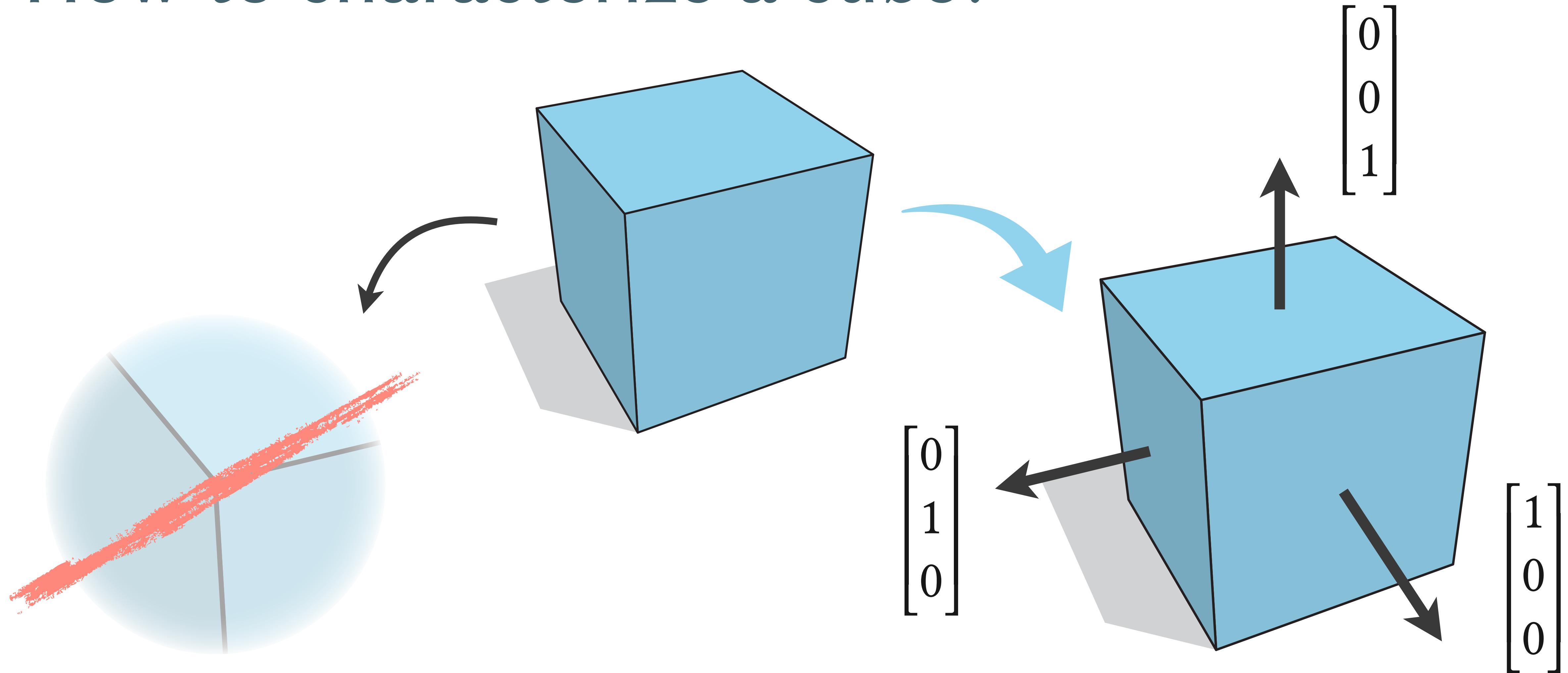
Taichi by Ju Ming

source: [wikimedia.org](#)

How to characterize a cube?



How to characterize a cube?



“cubic geometry has axis-aligned surface normals”

L1-Norm

$$\|n\|_1 = |n_x| + |n_y|$$

$$\| \uparrow \|_1$$

$$= \left\| \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right\|_1 = |0| + |1| = 1$$

$$\| \nearrow \|_1$$

$$= \left\| \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix} \right\|_1 = 1.414 \dots$$

→ $\| \text{axis-aligned unit vector} \|_1 < \| \text{non axis-aligned unit vector} \|_1$

→ “minimize L1 norm on normals leads to axis-aligned normals”

L1-norm on Surface Normals

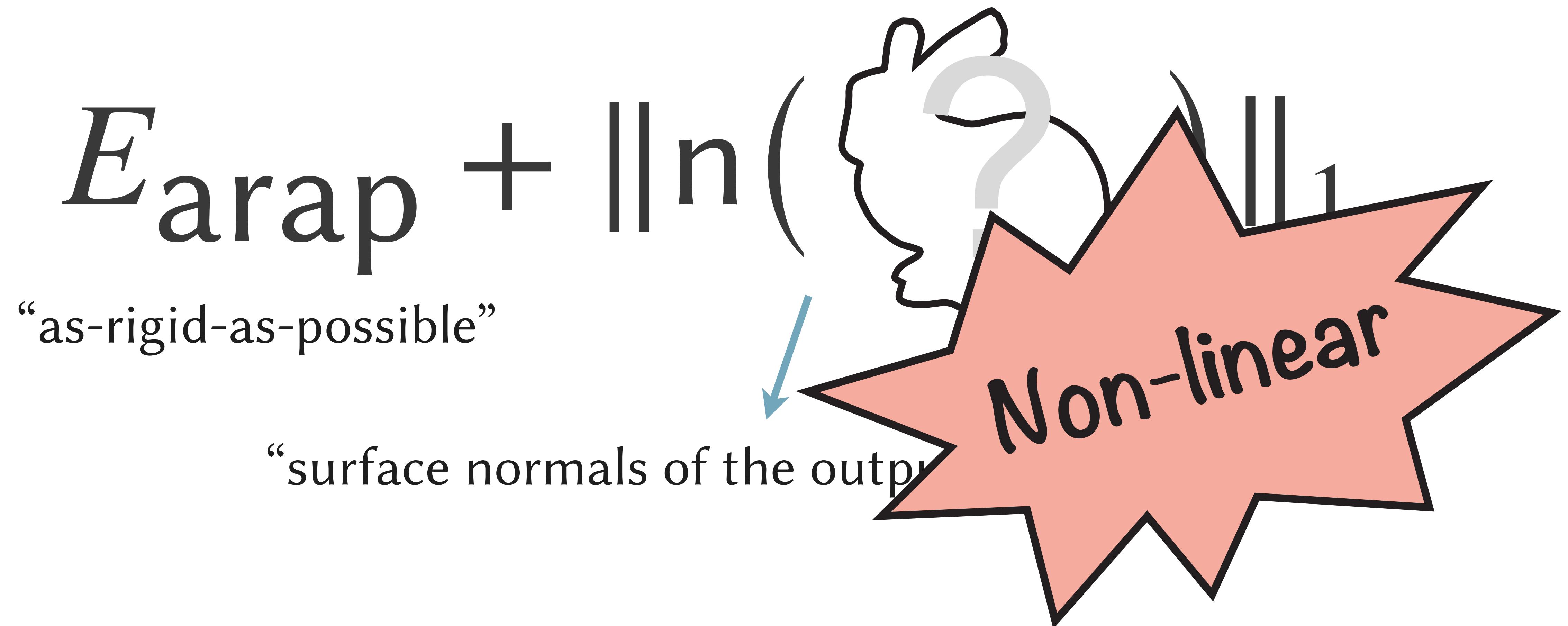
$$E_{\text{arap}} + \|\mathbf{n}(\text{?})\|_1$$

“as-rigid-as-possible”



“surface normals of the output mesh”

L1-norm on Surface Normals



Rotated Input Normals

$$E_{\text{arap}} + \| R \times n(\quad) \|_1$$

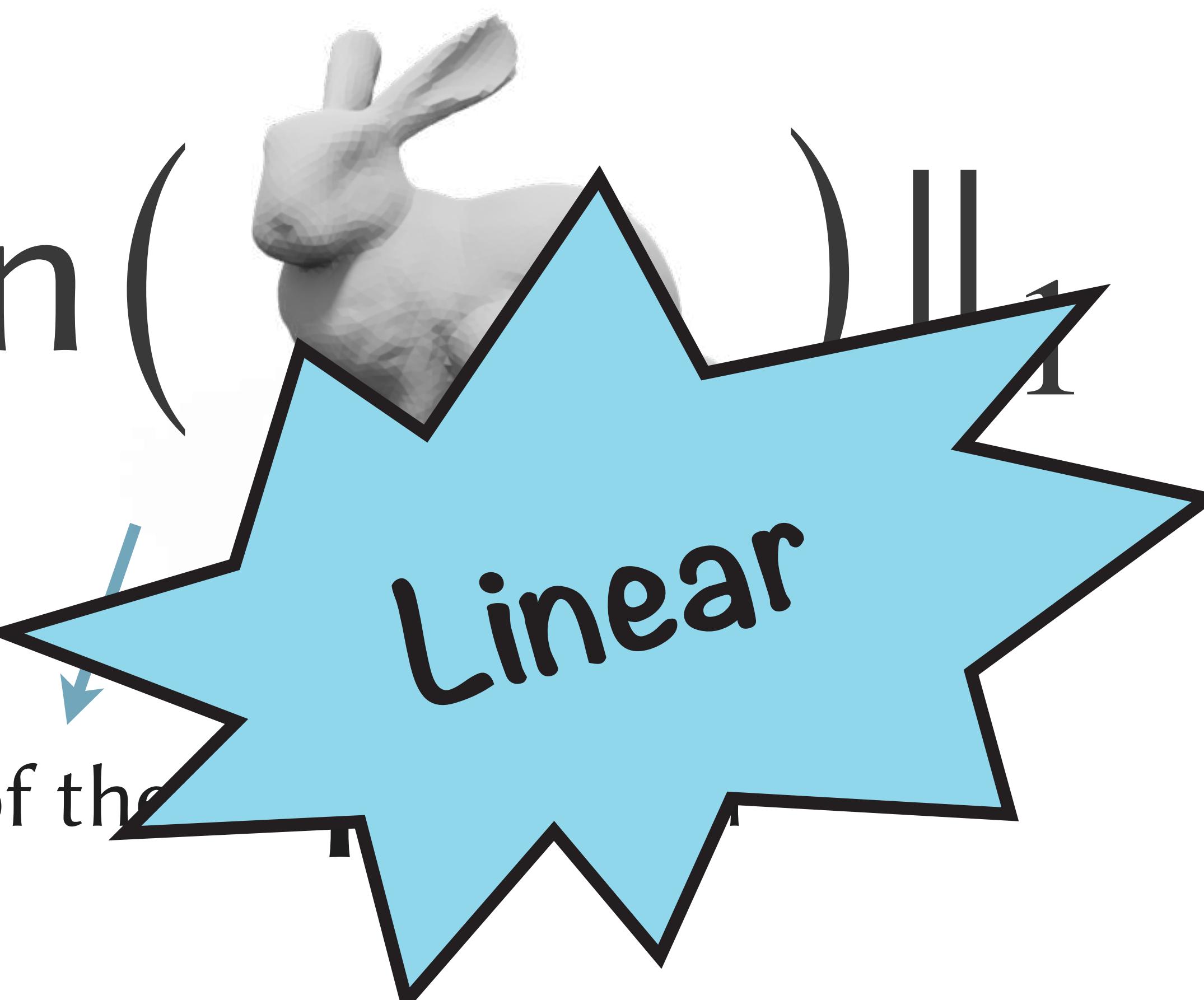


“**rotated** surface normals of the **input** mesh”

Rotated Input Normals

$E_{\text{arap}} + \|\mathbf{R} \times \mathbf{n}\|_1$

“rotated” surface normals of the mesh

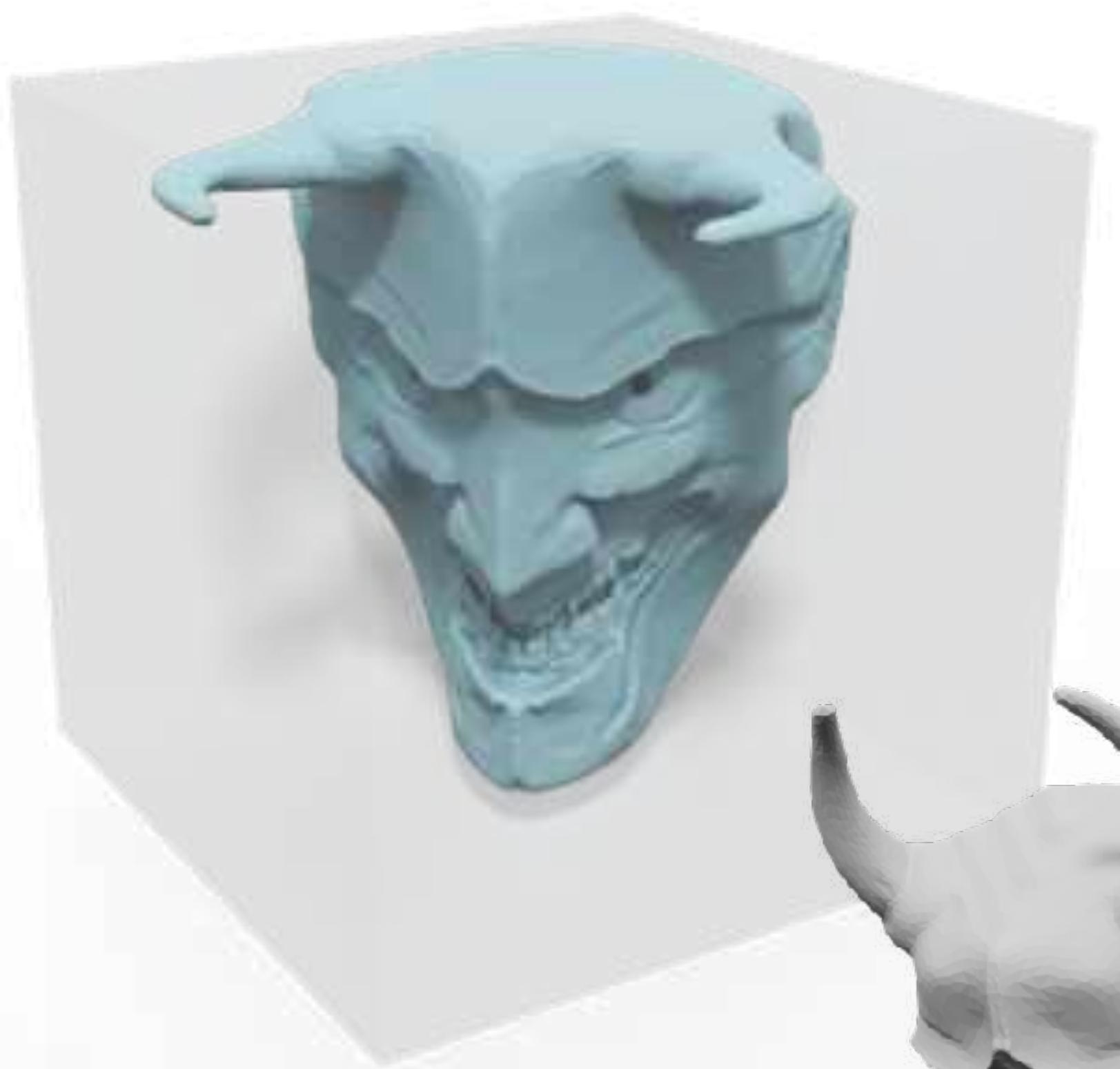
A 3D rendering of a hand holding a blue star-shaped mesh. The mesh has sharp, angular edges and faces. The word "Linear" is written across the middle face of the star. A small blue arrow points from the text "‘rotated’ surface normals of the mesh" towards the mesh itself.



Different “Cubeness”



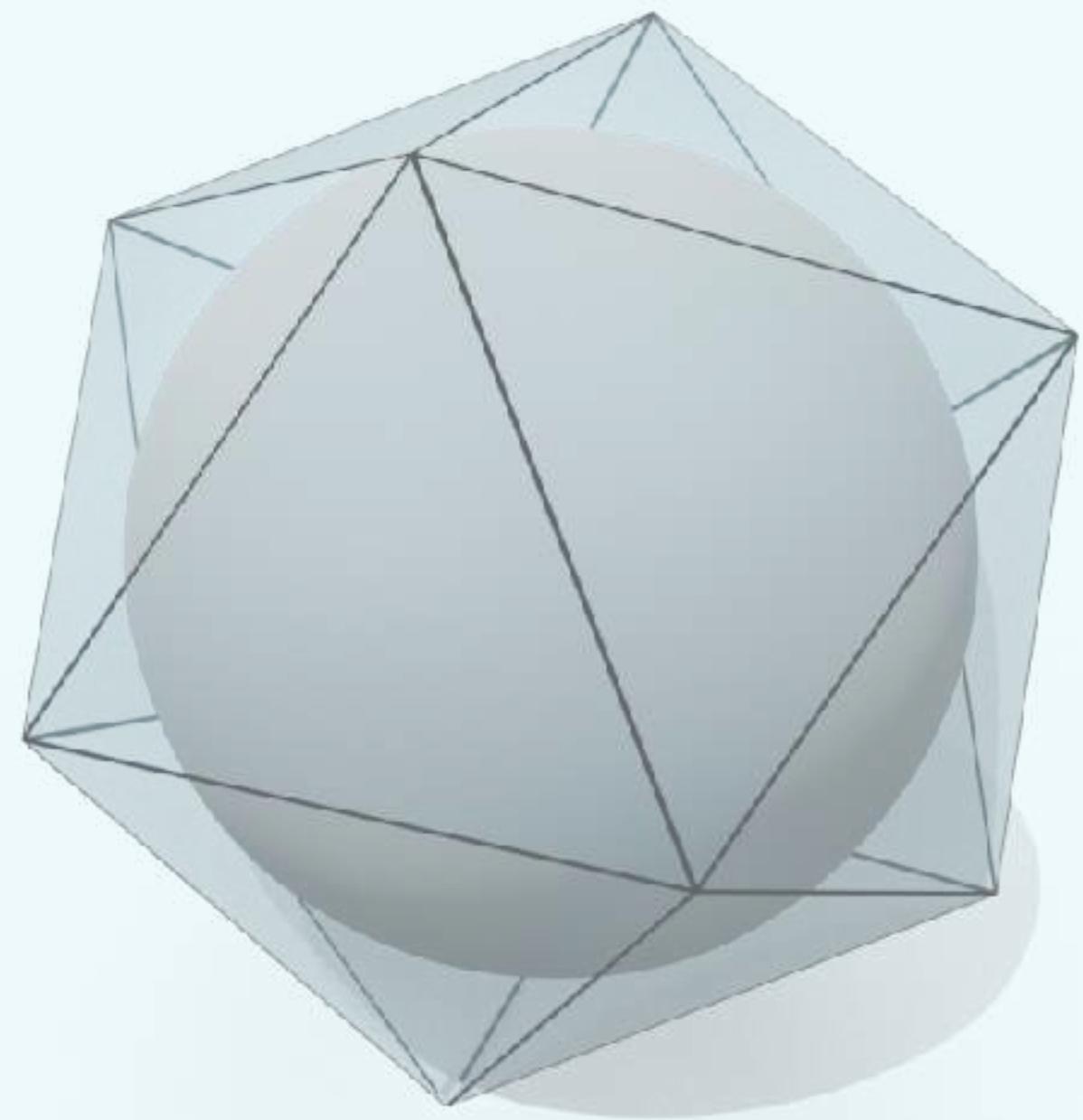
Orientation Dependent



Polygonal Boxes Stylization

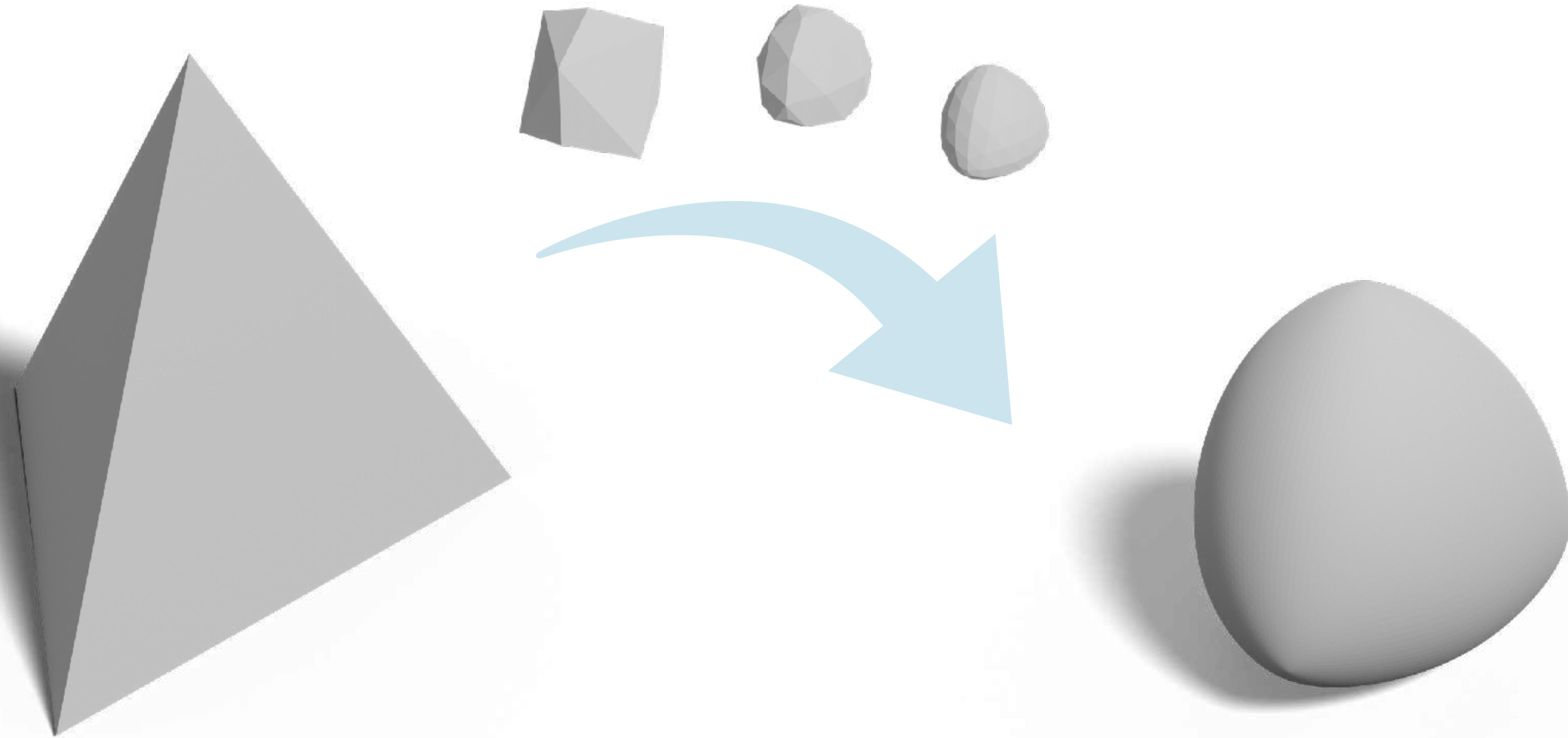


~~Characterizing a shape using surface vertices~~
Characterizing a shape using surface normals

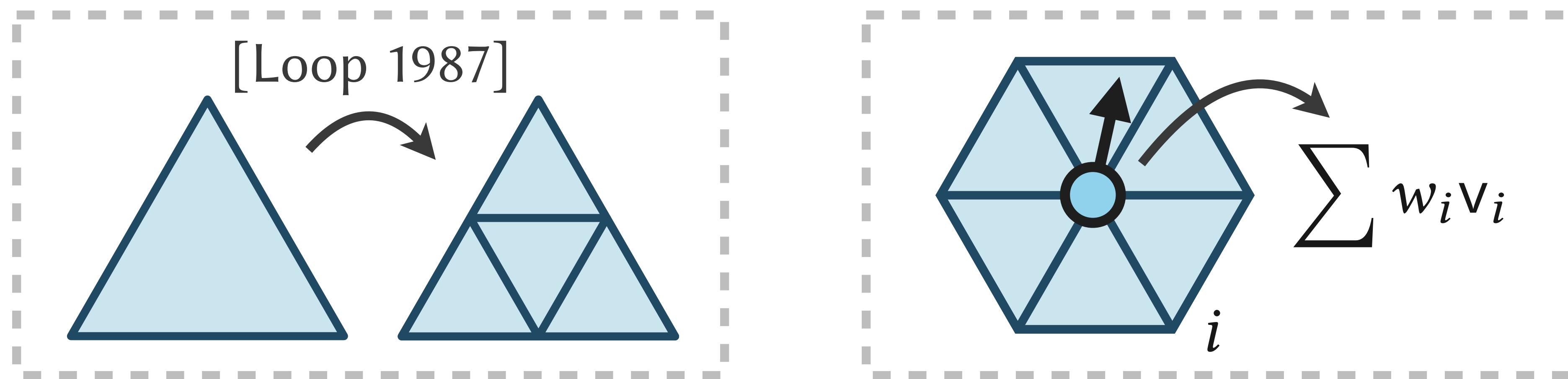


Data-Driven Subdivision

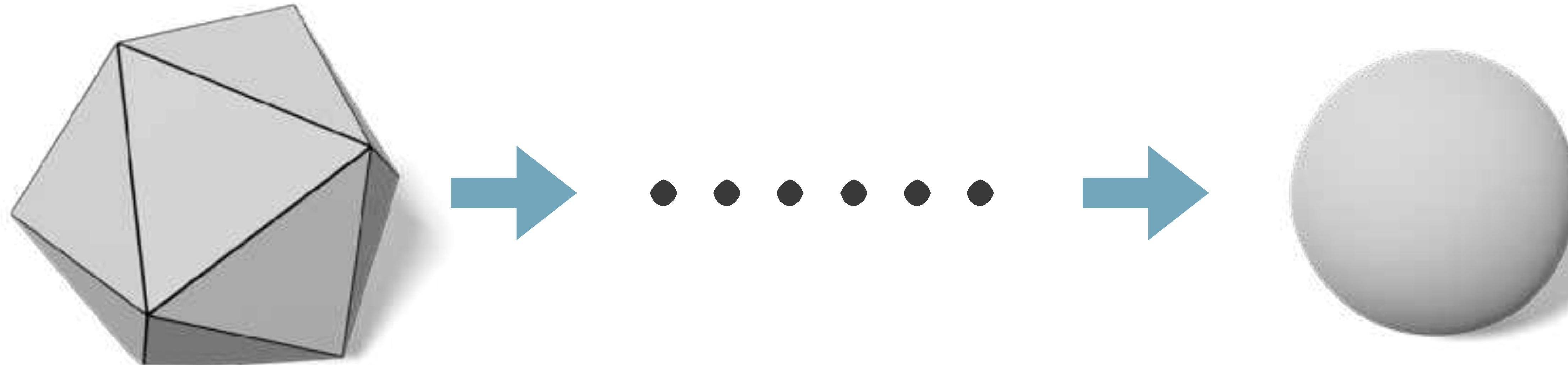
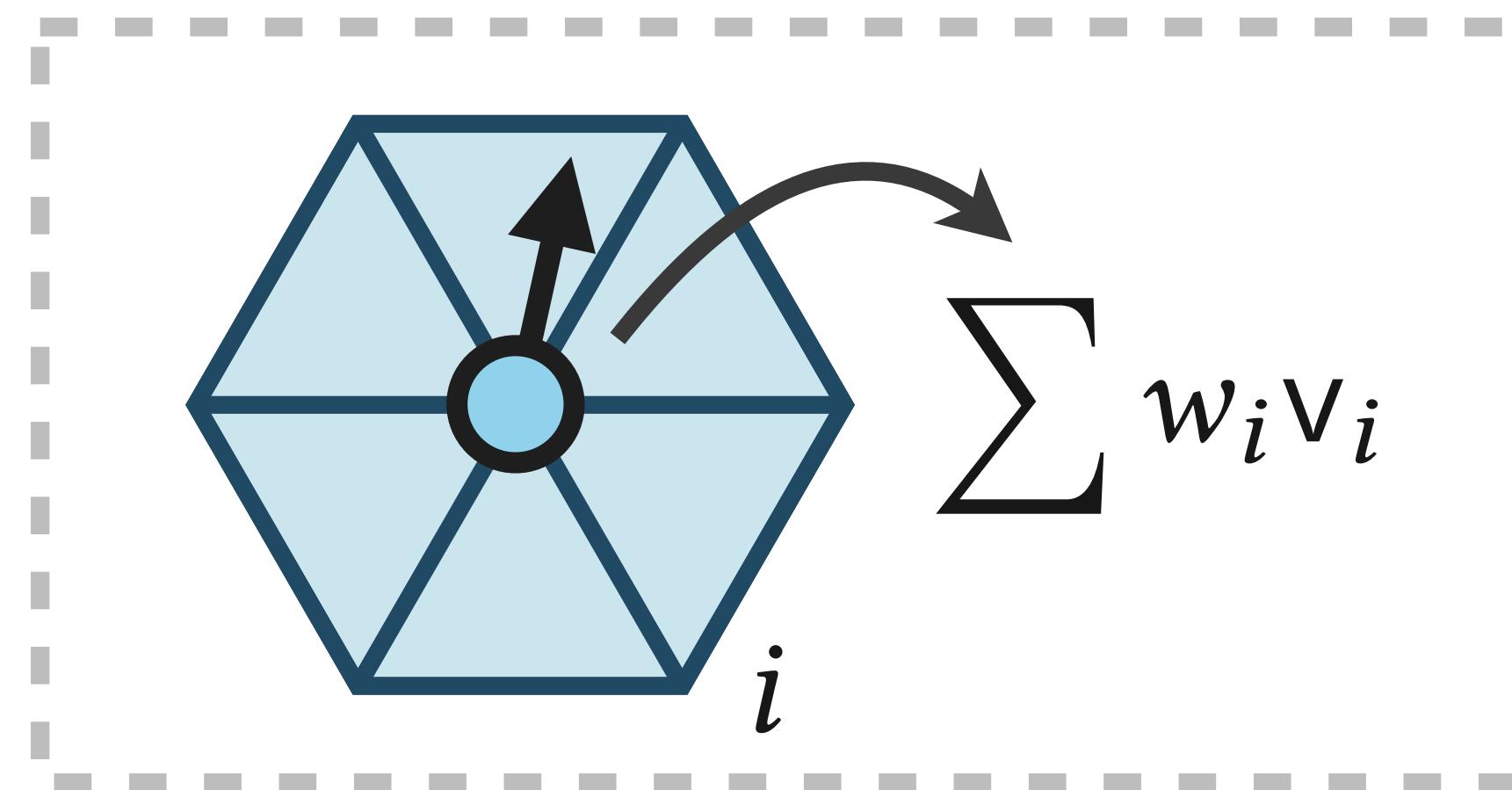
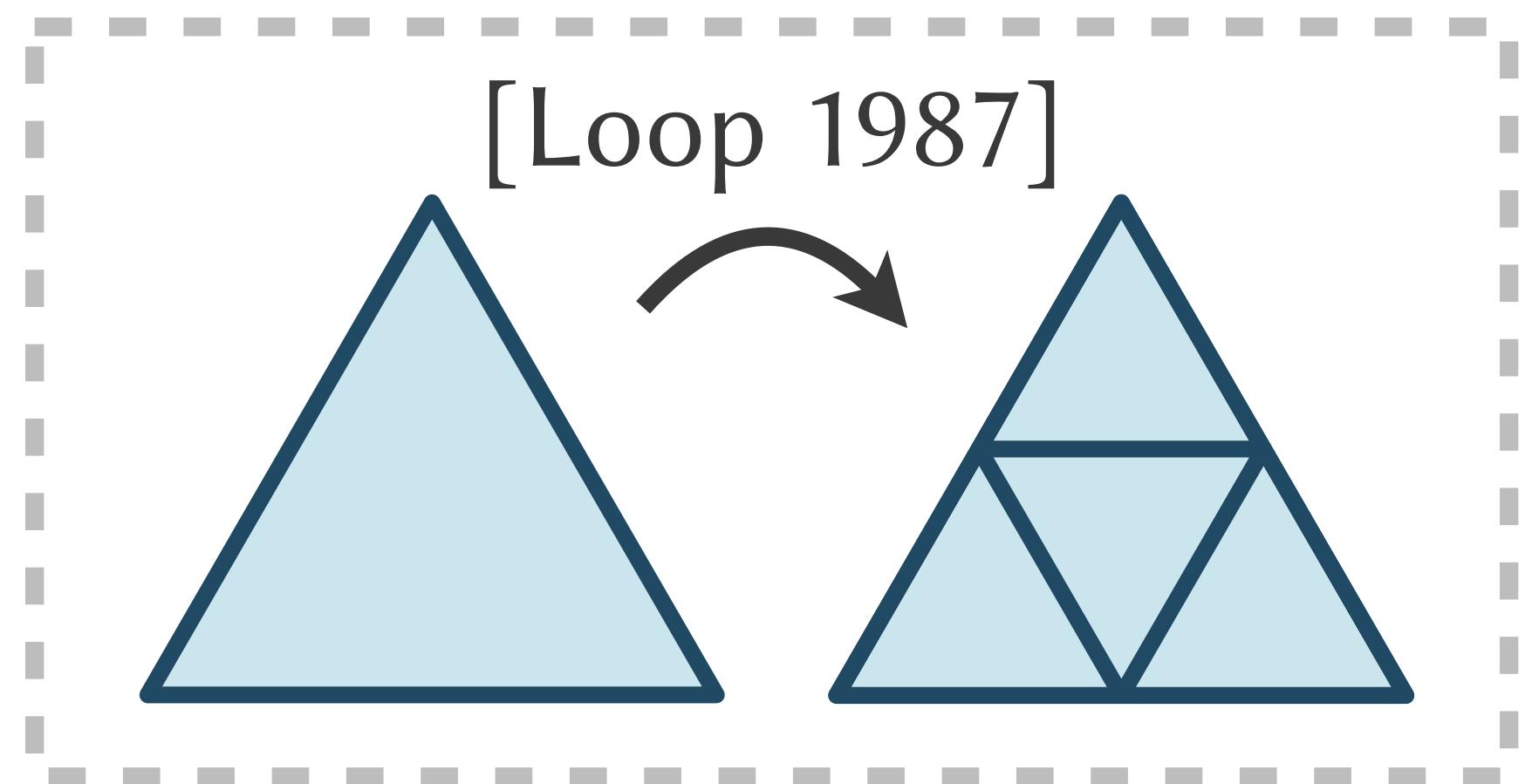
Subdivision Surfaces



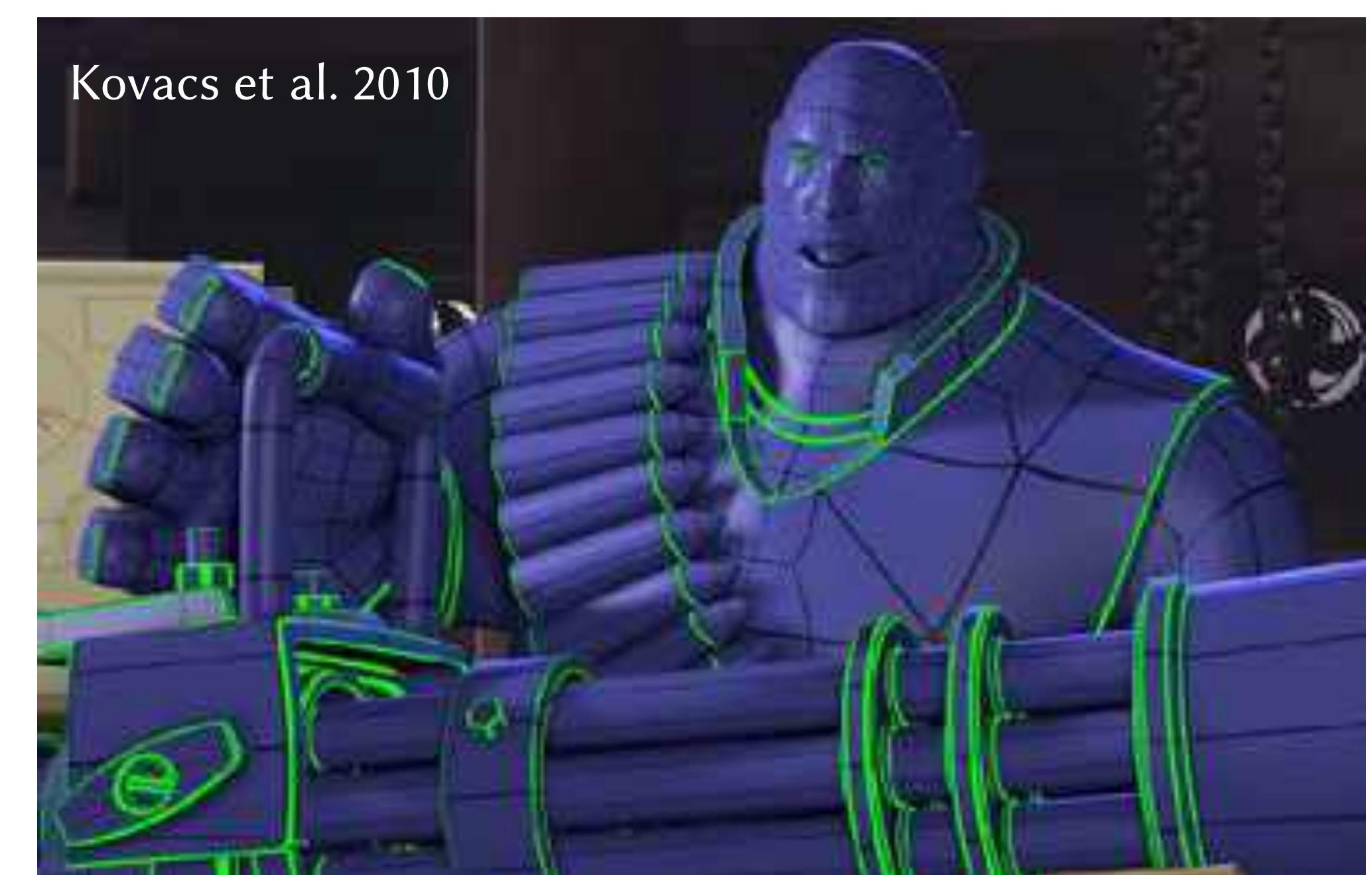
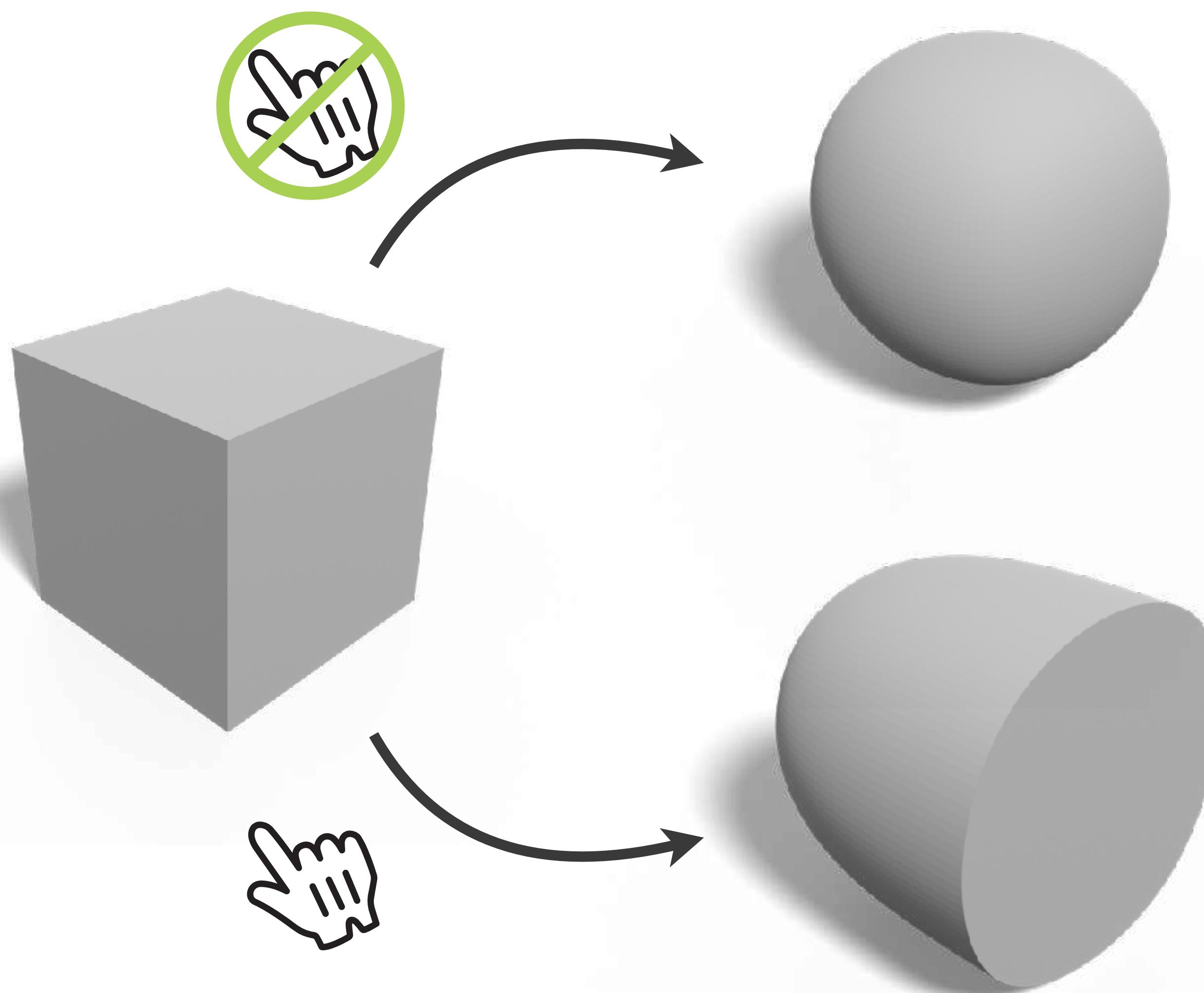
Classic Subdivision



Classic Subdivision

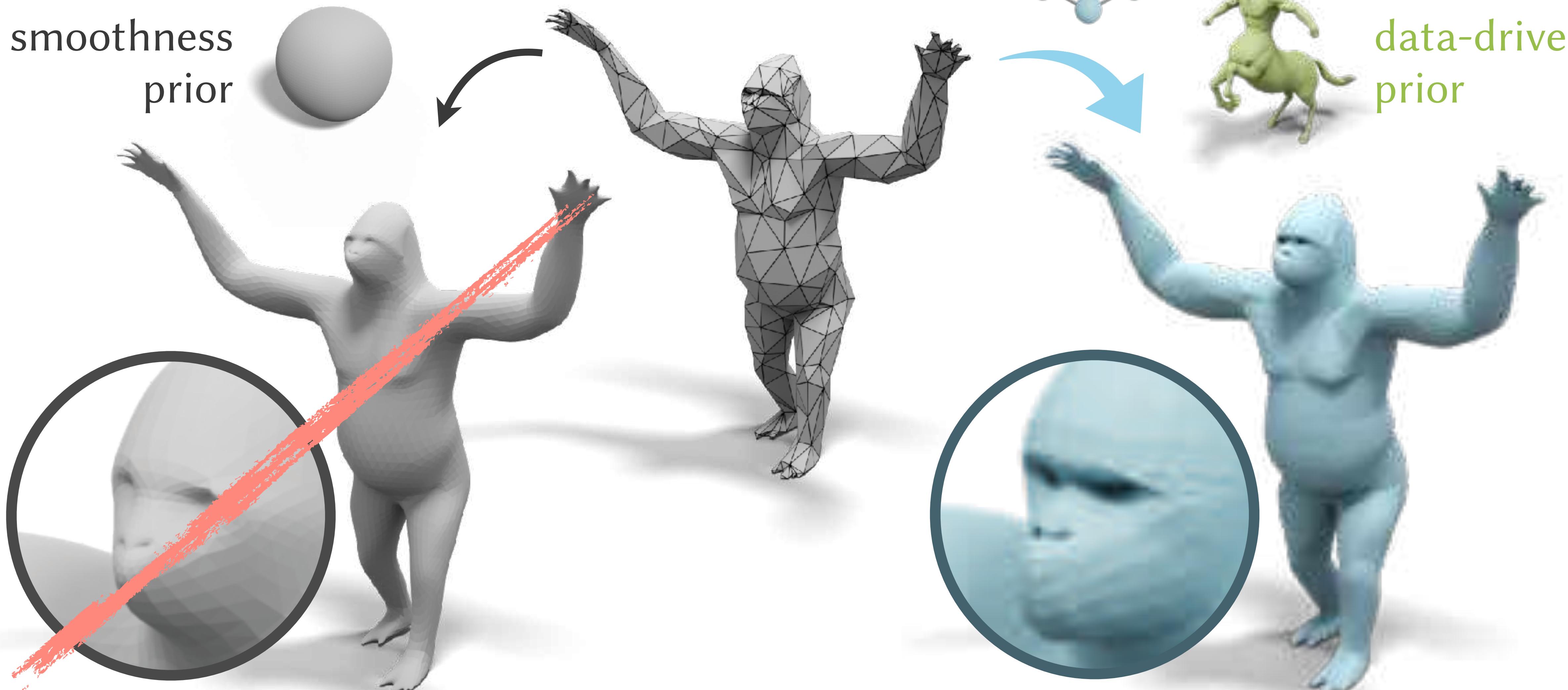


Smoothness prior

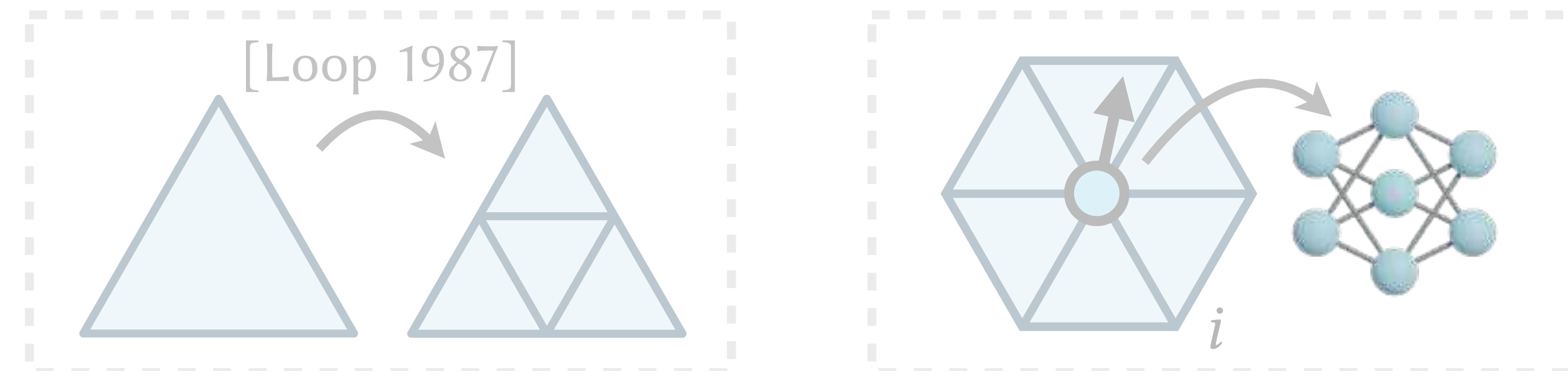


Neural Subdivision

smoothness
prior



Subdivision Network

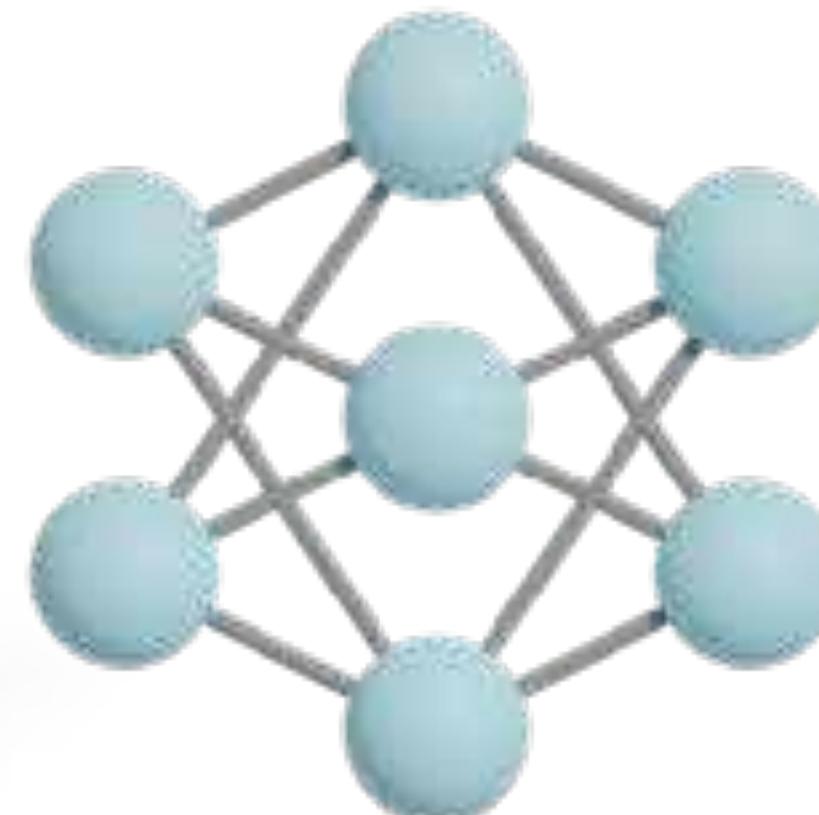


How to train subdivision networks

TRAINING
DATA



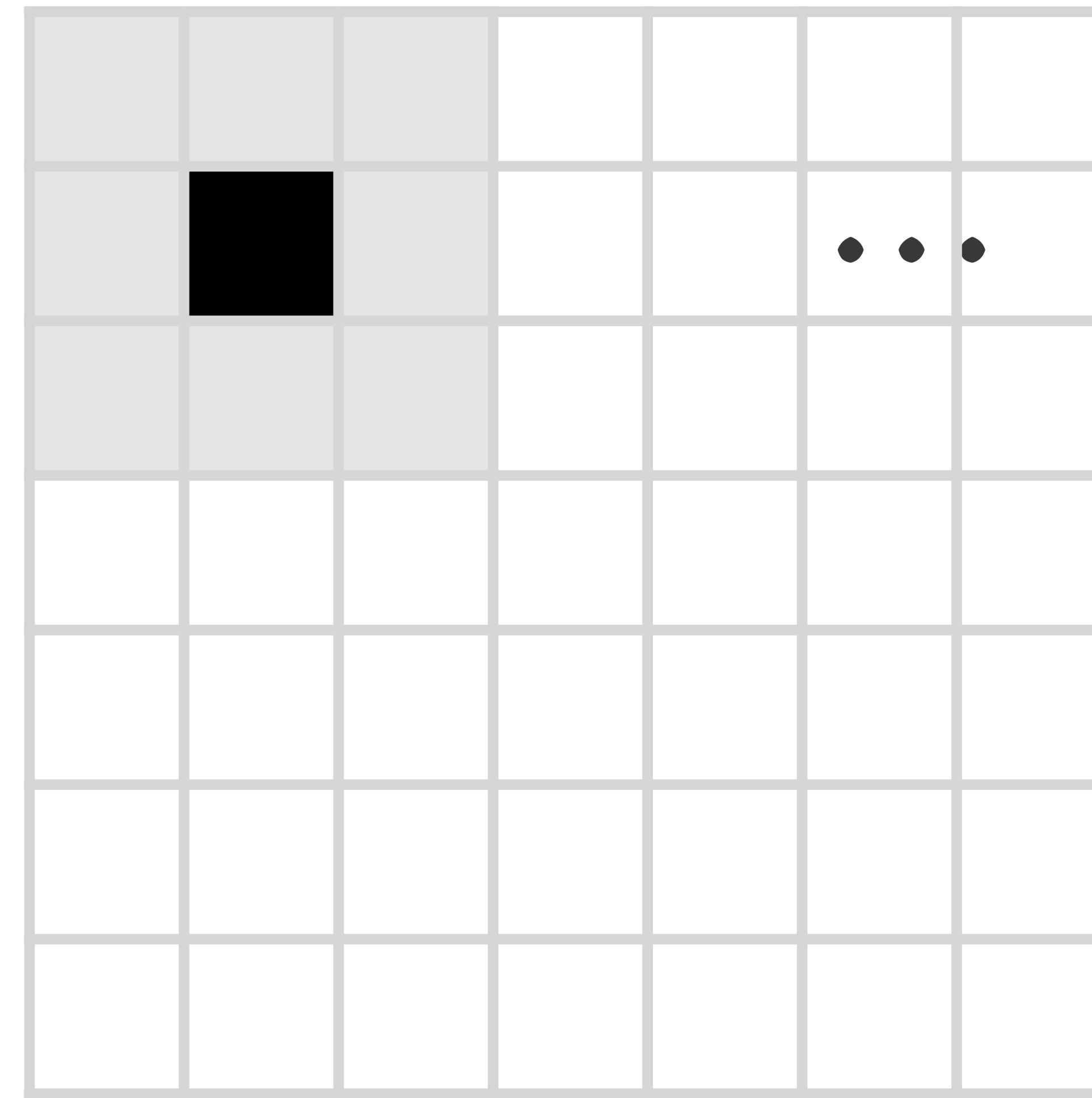
NETWORK
DESIGN



LOSS FUNCTION

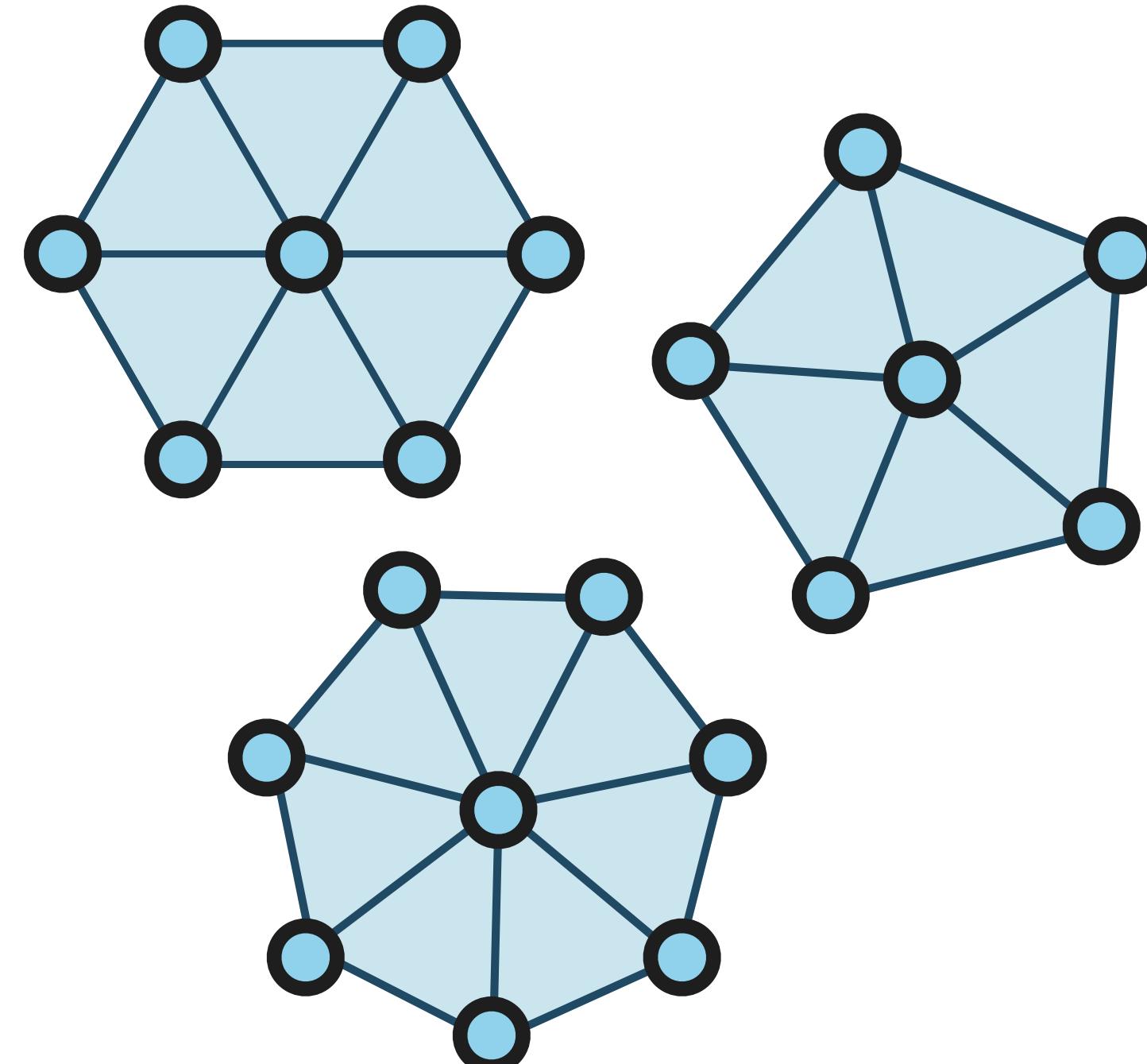
$$J(\text{ } \cdot \text{ } , \text{ } \cdot \text{ })$$


Machine Learning on Regular Grid



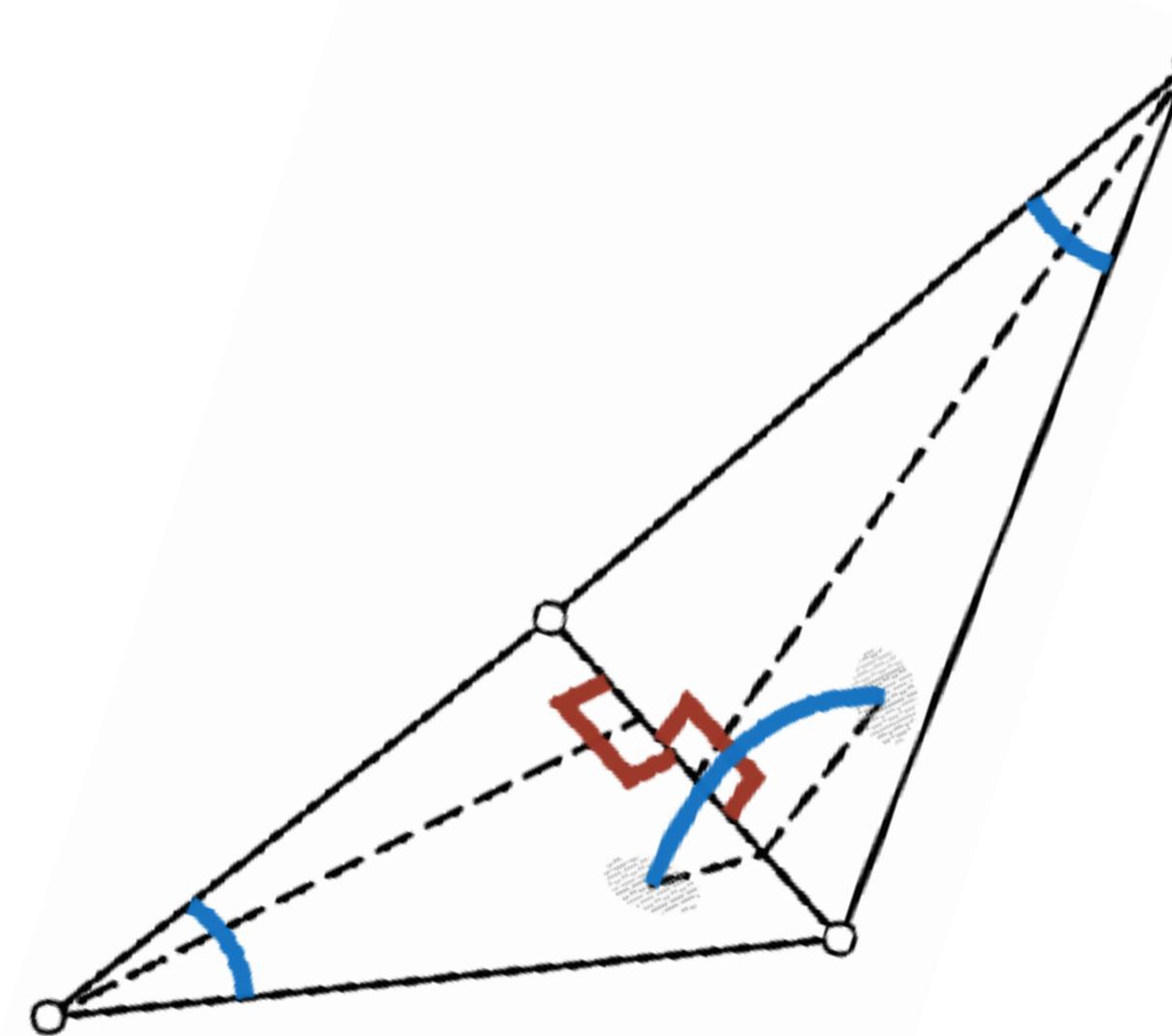
Handling Irregular Mesh

vertex rings



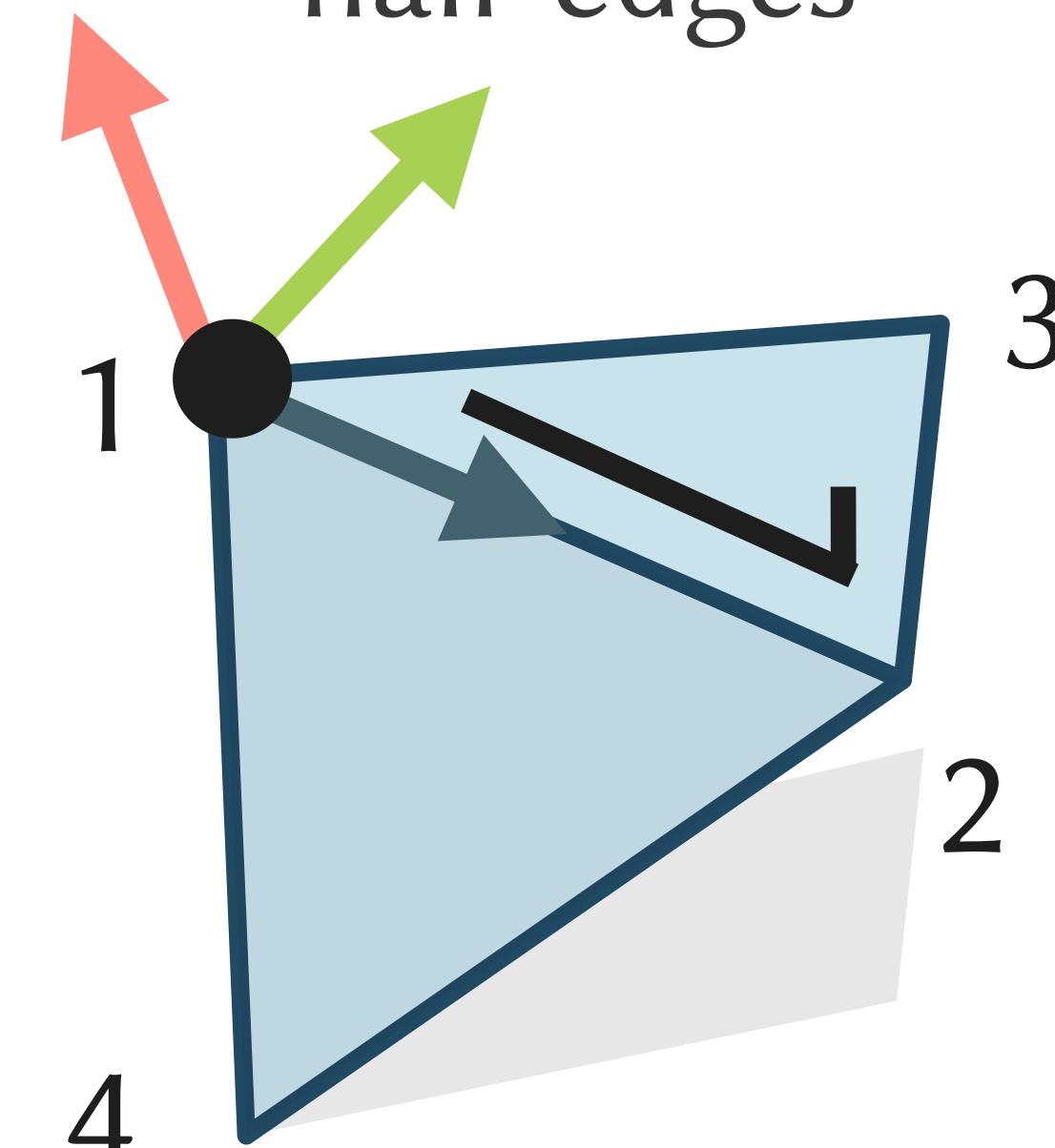
✗ no fixed input dim.
✗ no ordering

flaps of undirected edges
[Hanocka et al. 2019]



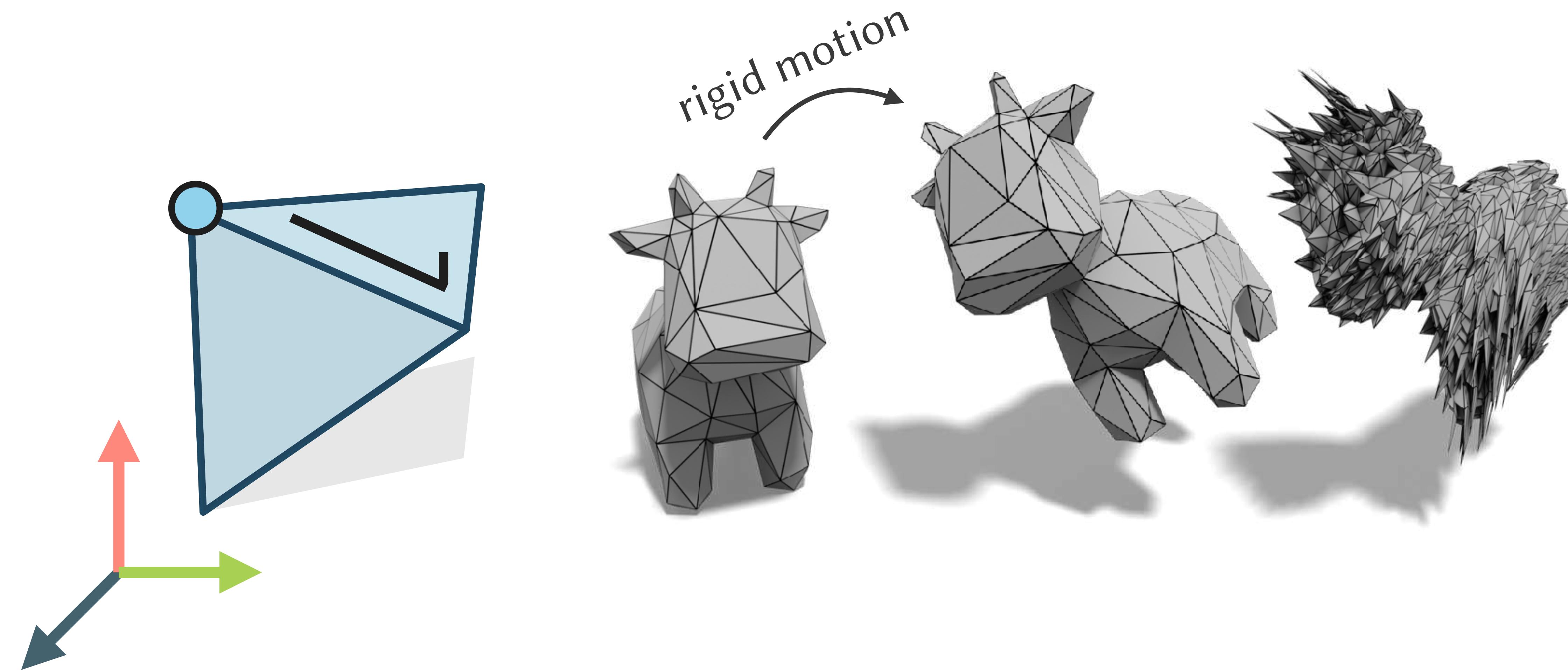
✓ fixed input dim.
✗ no ordering

flaps of half-edges

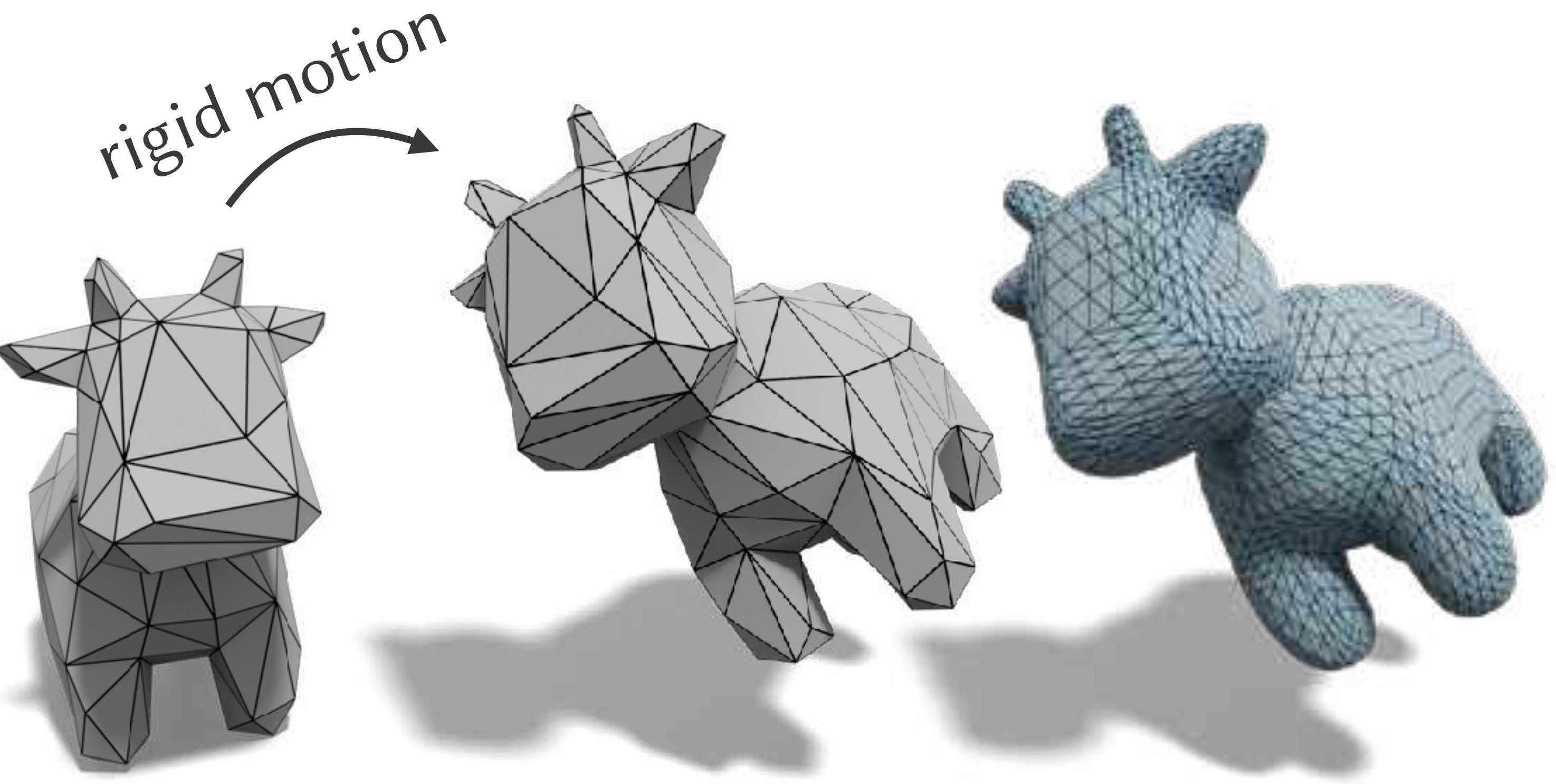
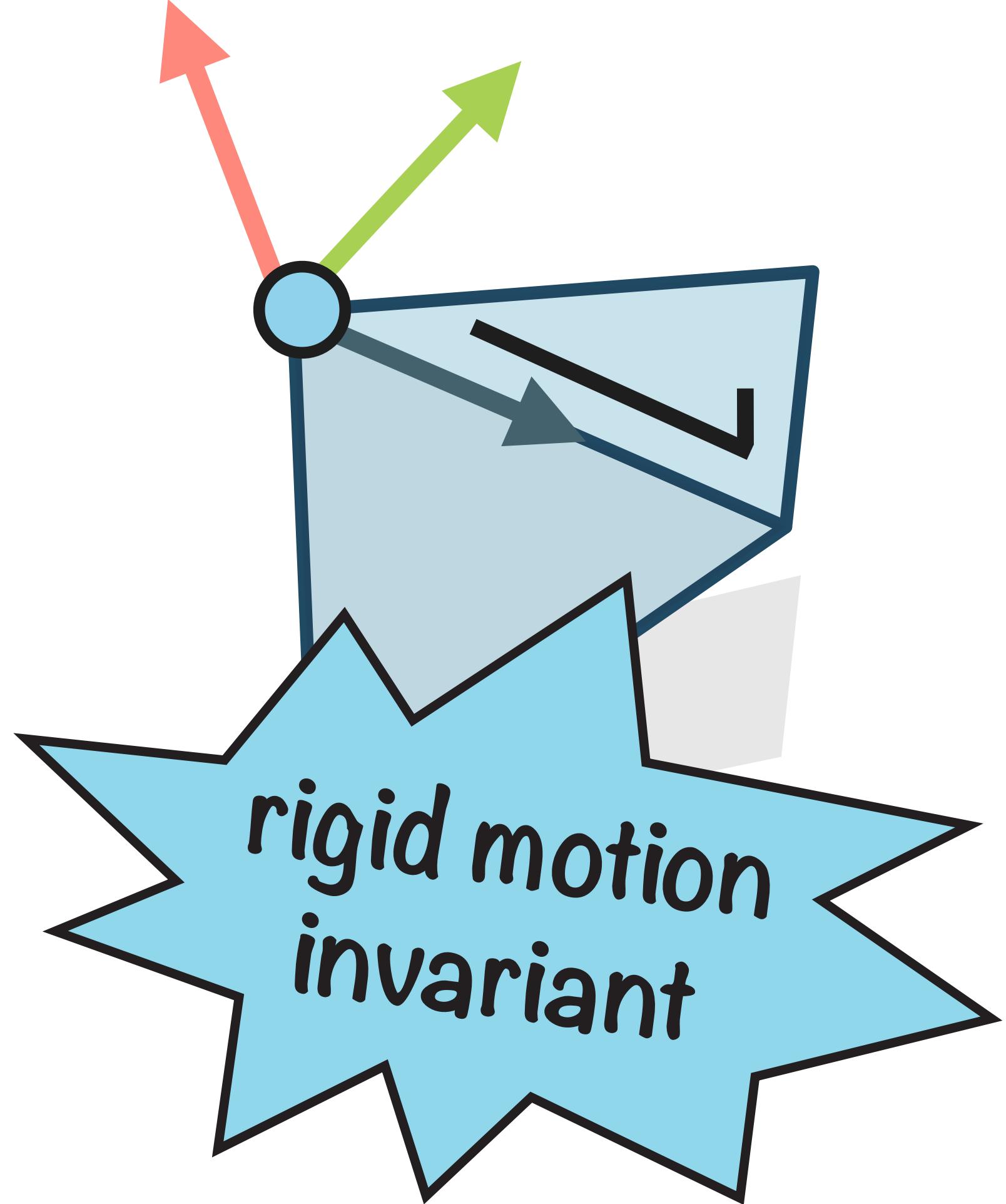


✓ fixed input dim.
✓ canonical ordering

Rigid Motion Invariant



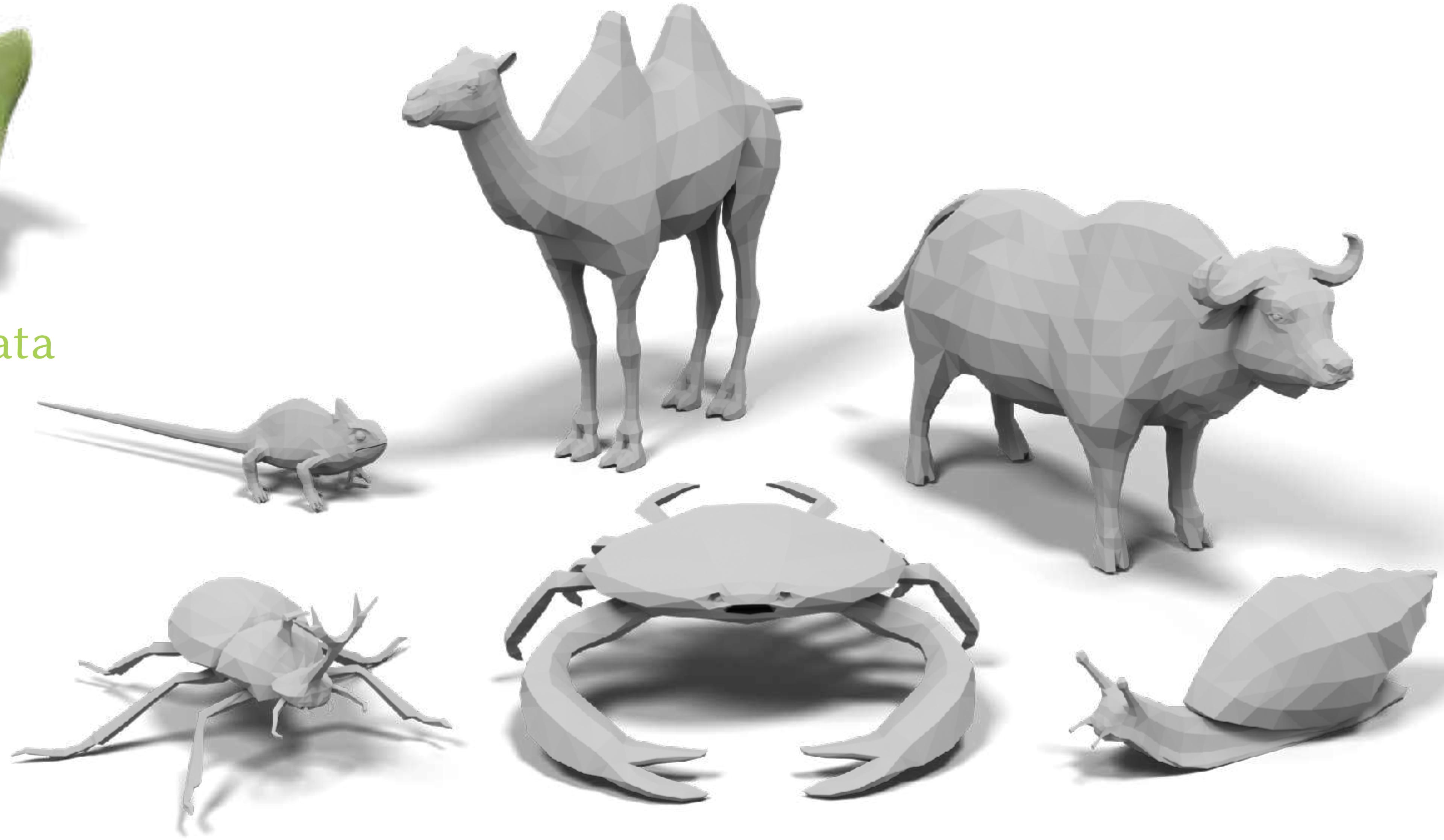
Rigid Motion Invariant



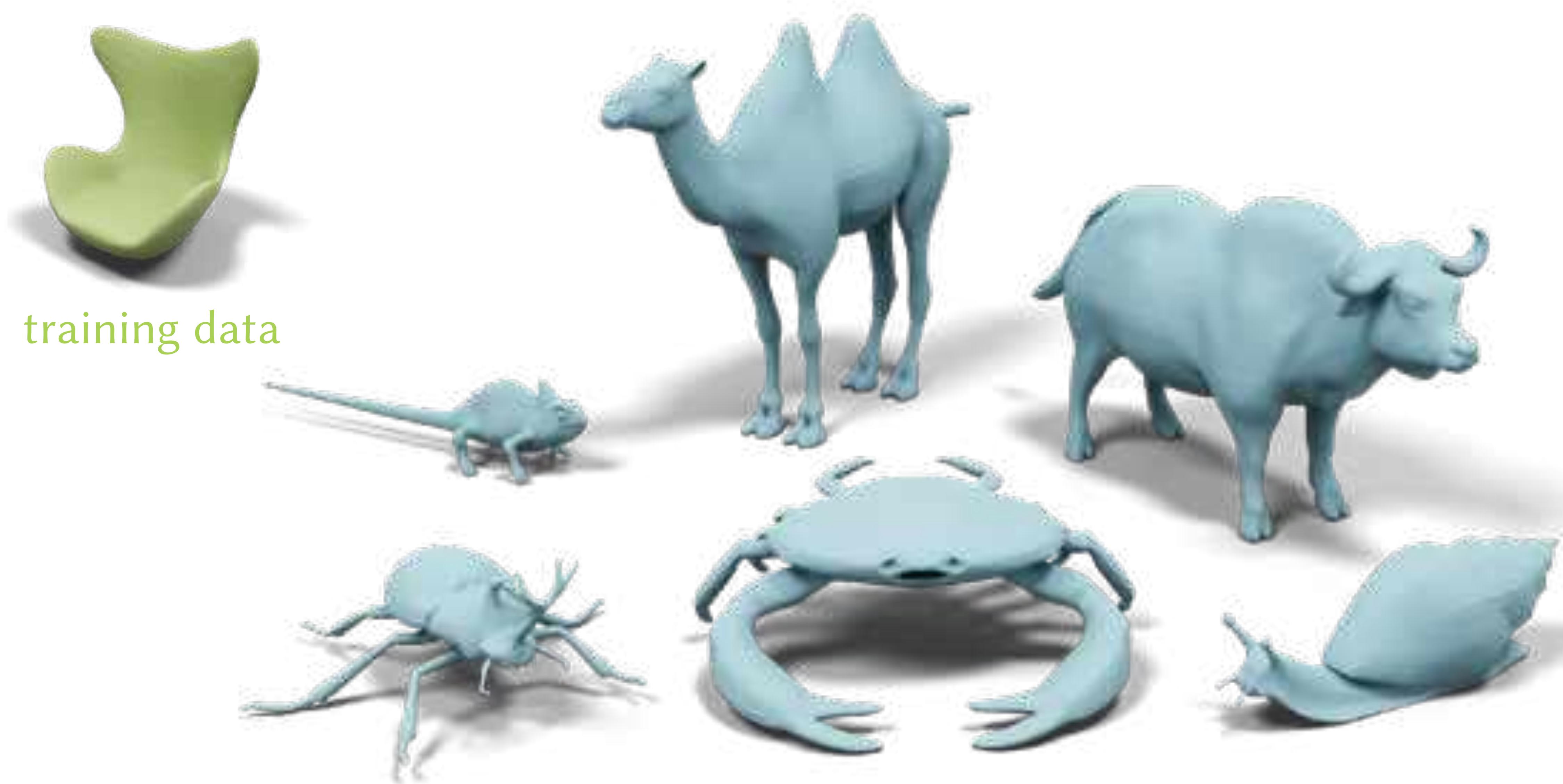
Generalize when trained on single shape



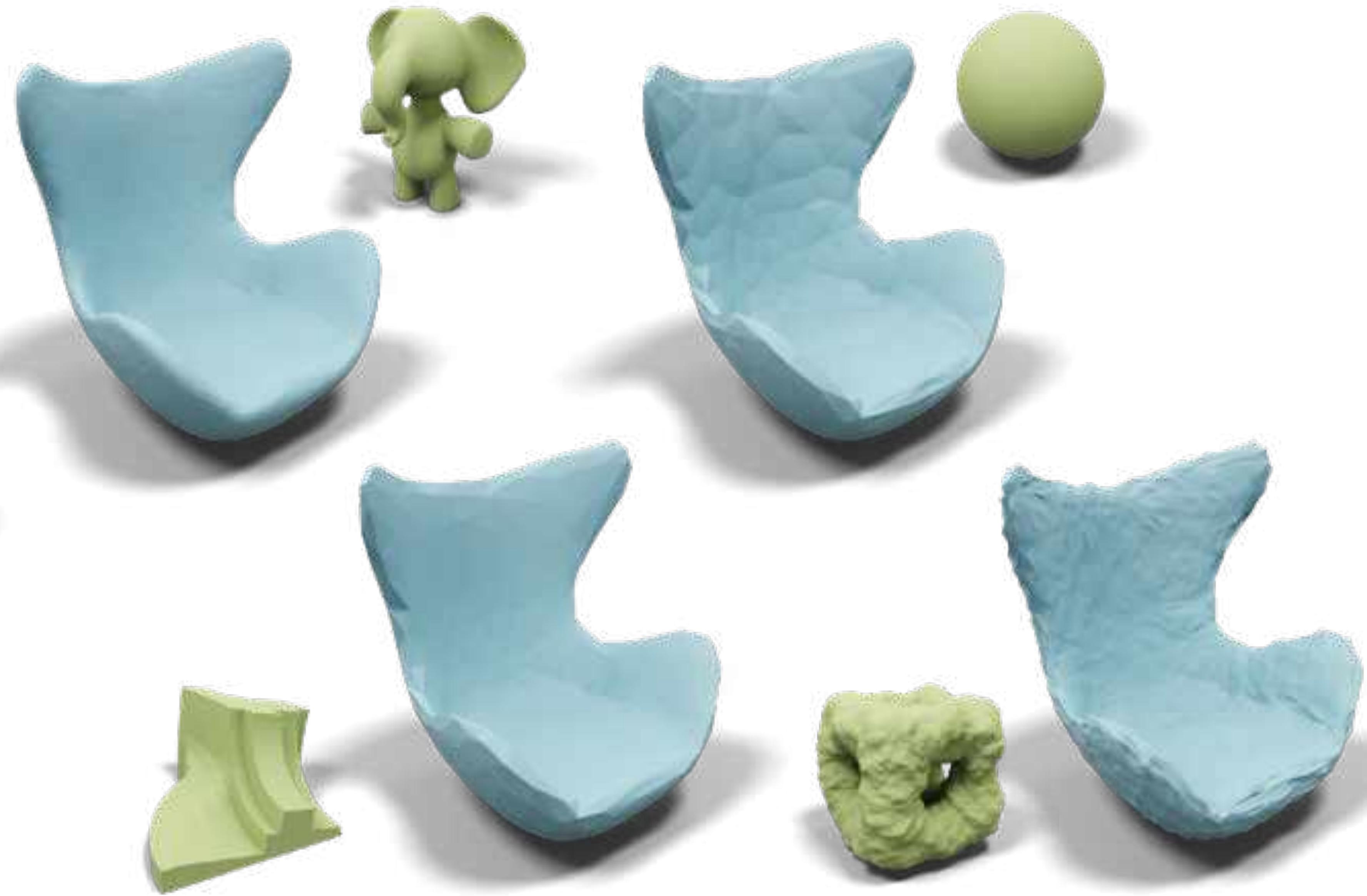
training data



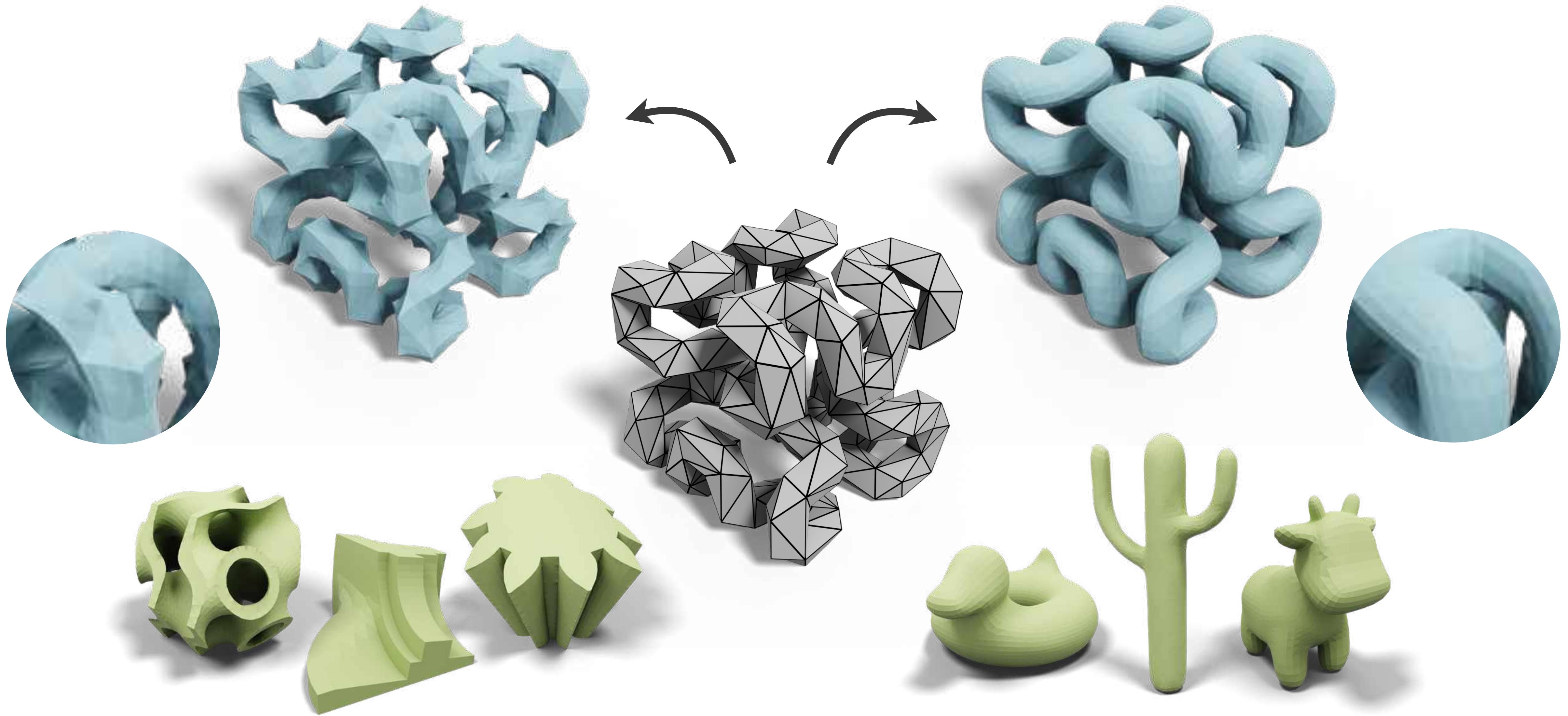
Generalize when trained on single shape



Stylized Subdivision



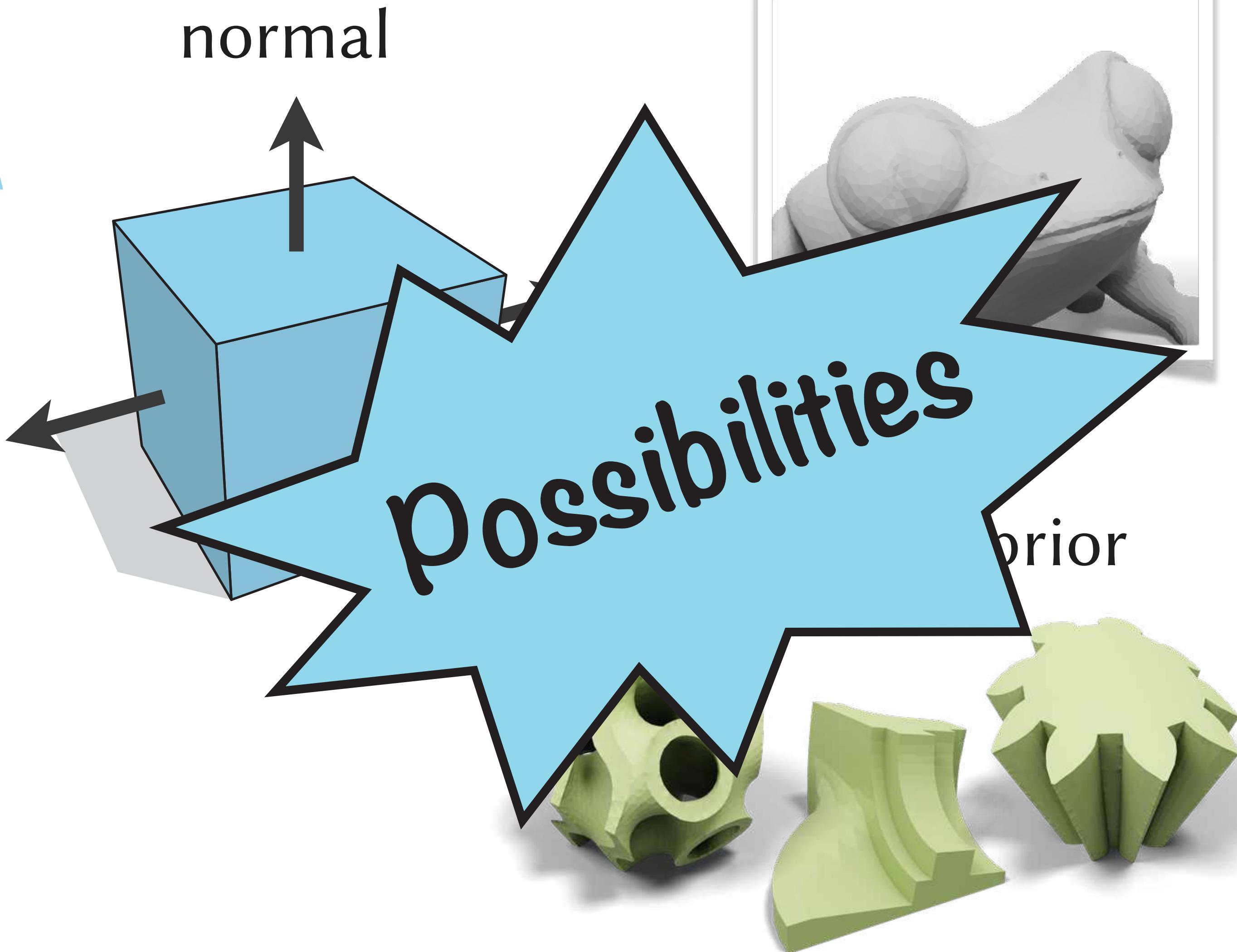
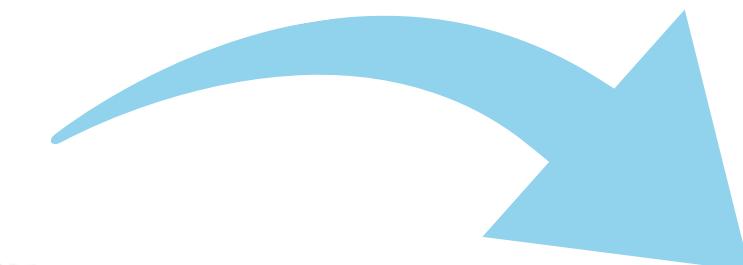
Data-Driven prior



~~Smoothness prior~~
Data-driven prior

Different Perspectives

appearance





3D Modeling for Everyone

Hsueh-Ti Derek Liu
hsuehtil@cs.toronto.edu