Black-Box Analysis:

From Theory to **Practice**

Teseo Schneider

https://cs.nyu.edu/~teseo/

NYU | COURANT INSTITUTE OF MATHEMATICAL SCIENCES

Getting Started

- The libraries we will be using are implemented in C++ for efficiency reasons, but are exposed to python for ease of integration

- All libraries are available on Conda, they can be installed with

```
conda install meshplot
conda install igl
conda install wildmeshing
conda install polyfempy
```

Libraries Overview

Cross Platform: Windows, MacOSX, Linux

MeshPlot

Jupyter Tutorial

skoch9/meshplot
8 Stars · 2 Forks

meshplot

Home

Jupyter Tutorial

Docs

Visualizing Surfaces

We can visualize surfaces, their properties and additional debugging information through the `plot` function. Let's visualize the previously loaded triangle mesh:

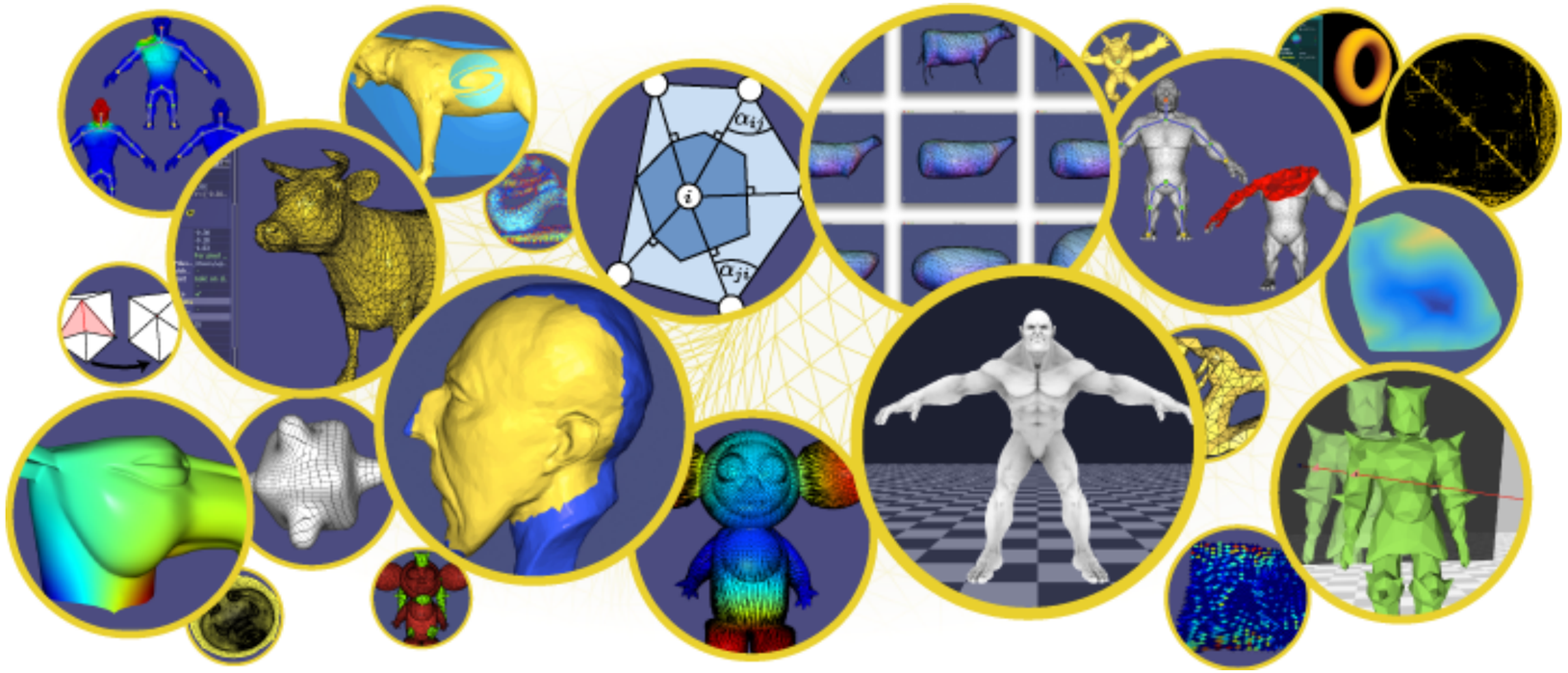
```
mp.plot(v, f)
```

Table of contents

https://skoch9.github.io/meshplot/

Install with conda

Interactive Geometry Library (libigl)



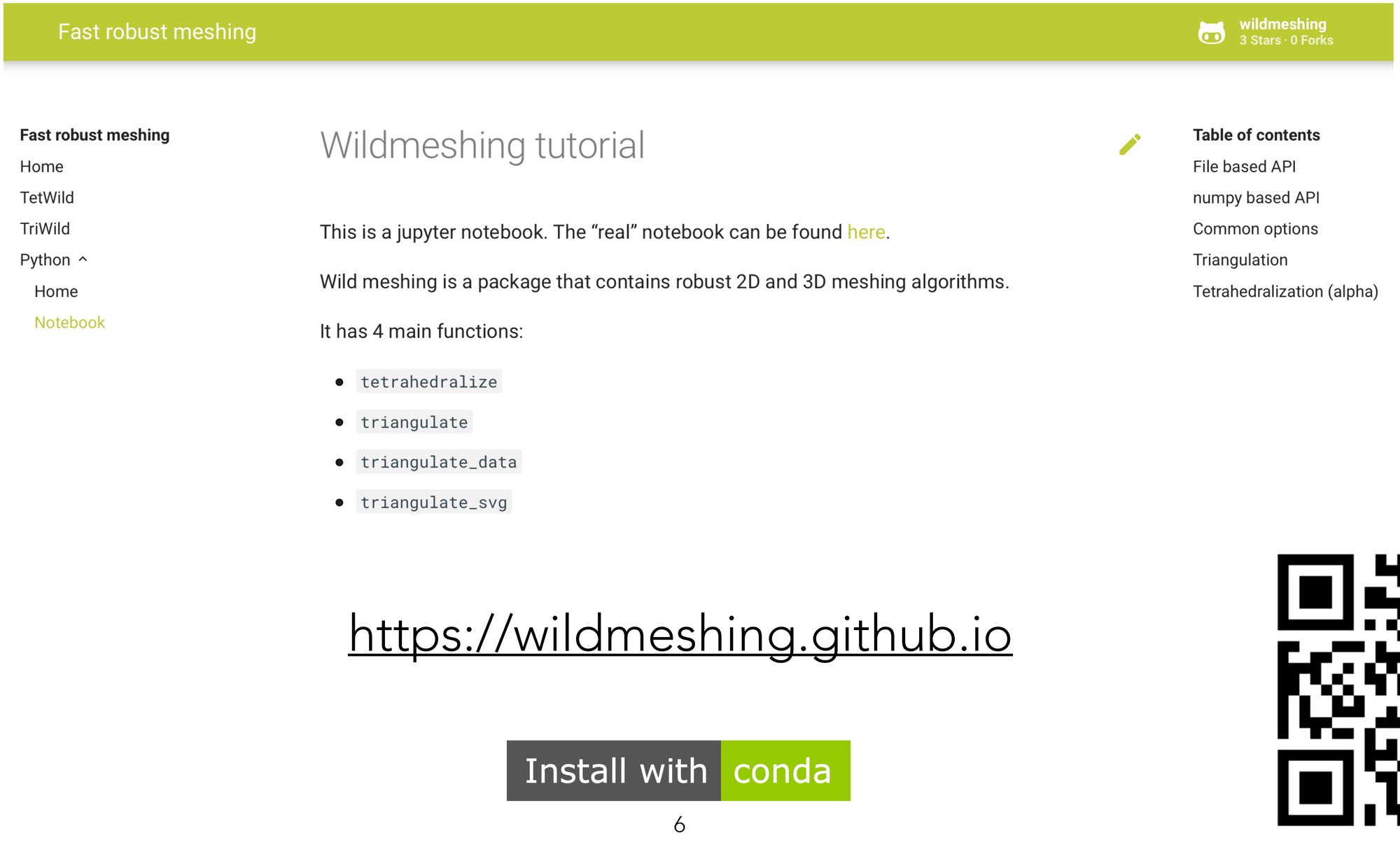
https://libigl.github.io

https://geometryprocessing.github.io/libigl-python-bindings/

Install with conda

downloads 2k total

Wild Meshing (TetWild and TriWild)



https://wildmeshing.github.io

Install with conda

downloads 2k total

PolyFEM

Home | Search | polyfem/polyfem 22 Stars · 3 Forks

polyfem

- Home
- Tutorial
- Documentation
- Python [alpha]
- Jupyter examples
- Python docs

Table of contents

https://polyfem.github.io

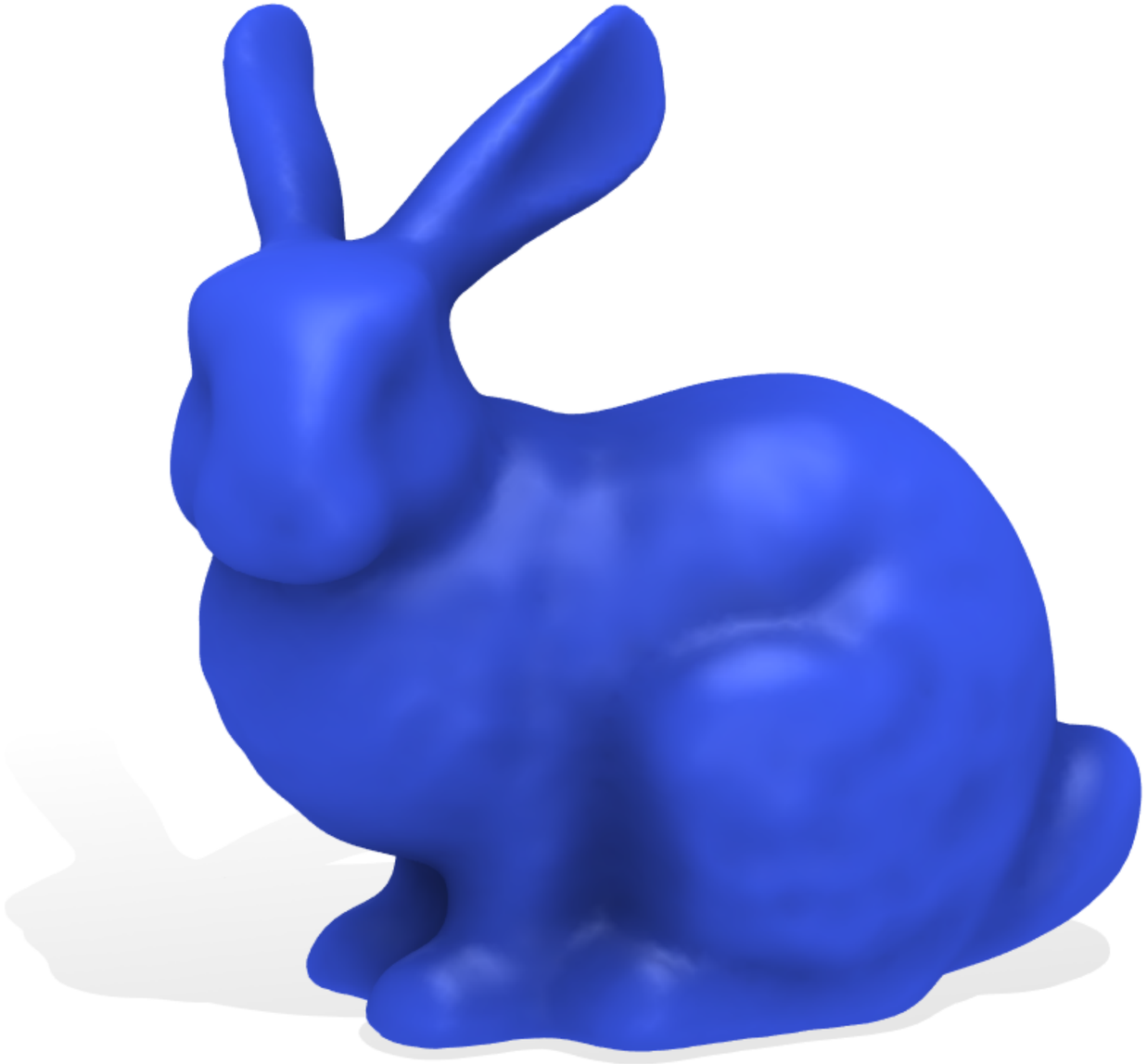
Install with conda

downloads 18k total

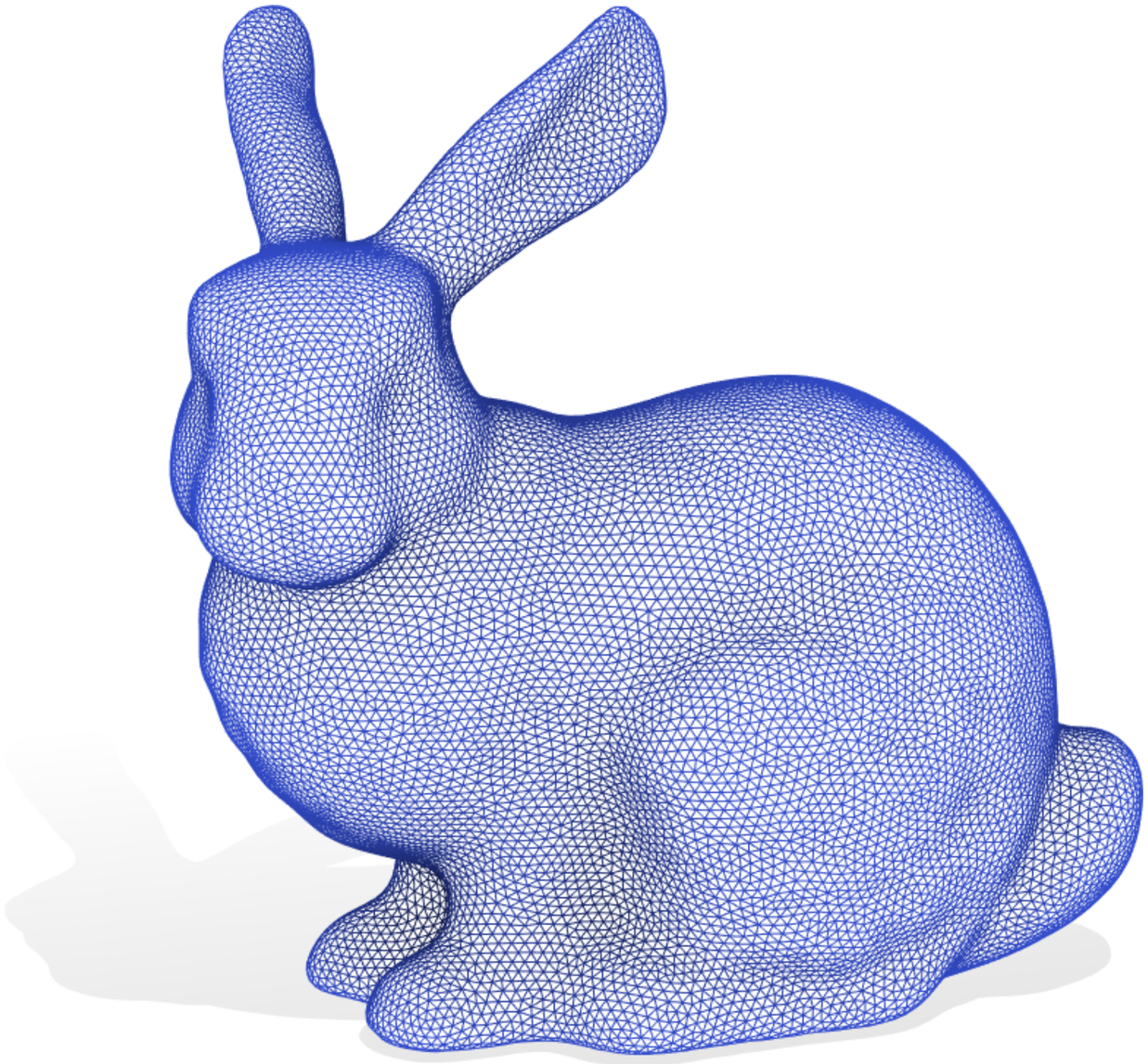
Data Structures (or lack thereof)

Acknowledgement: Alec Jacobson

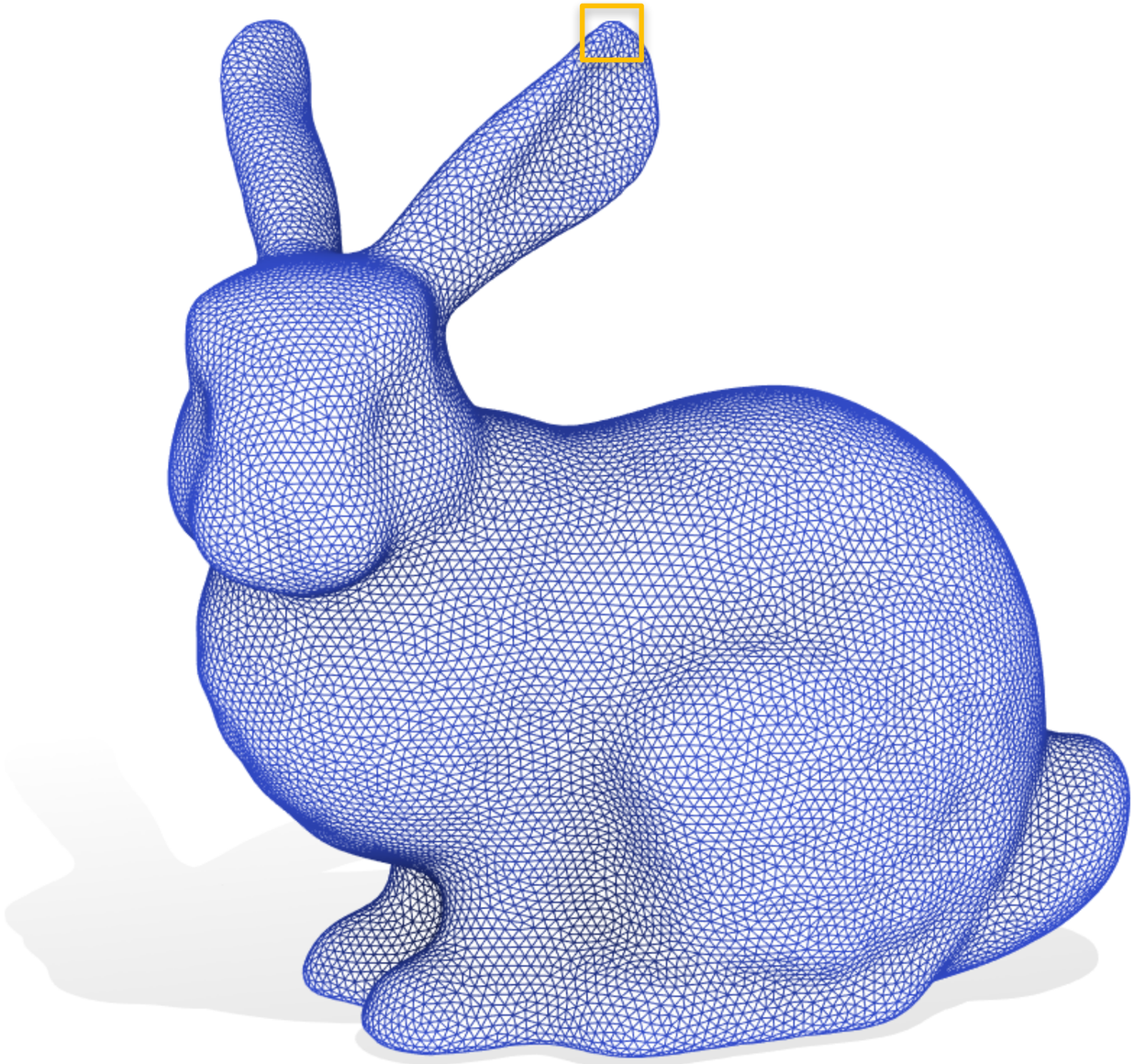
Triangle meshes discretize surfaces…

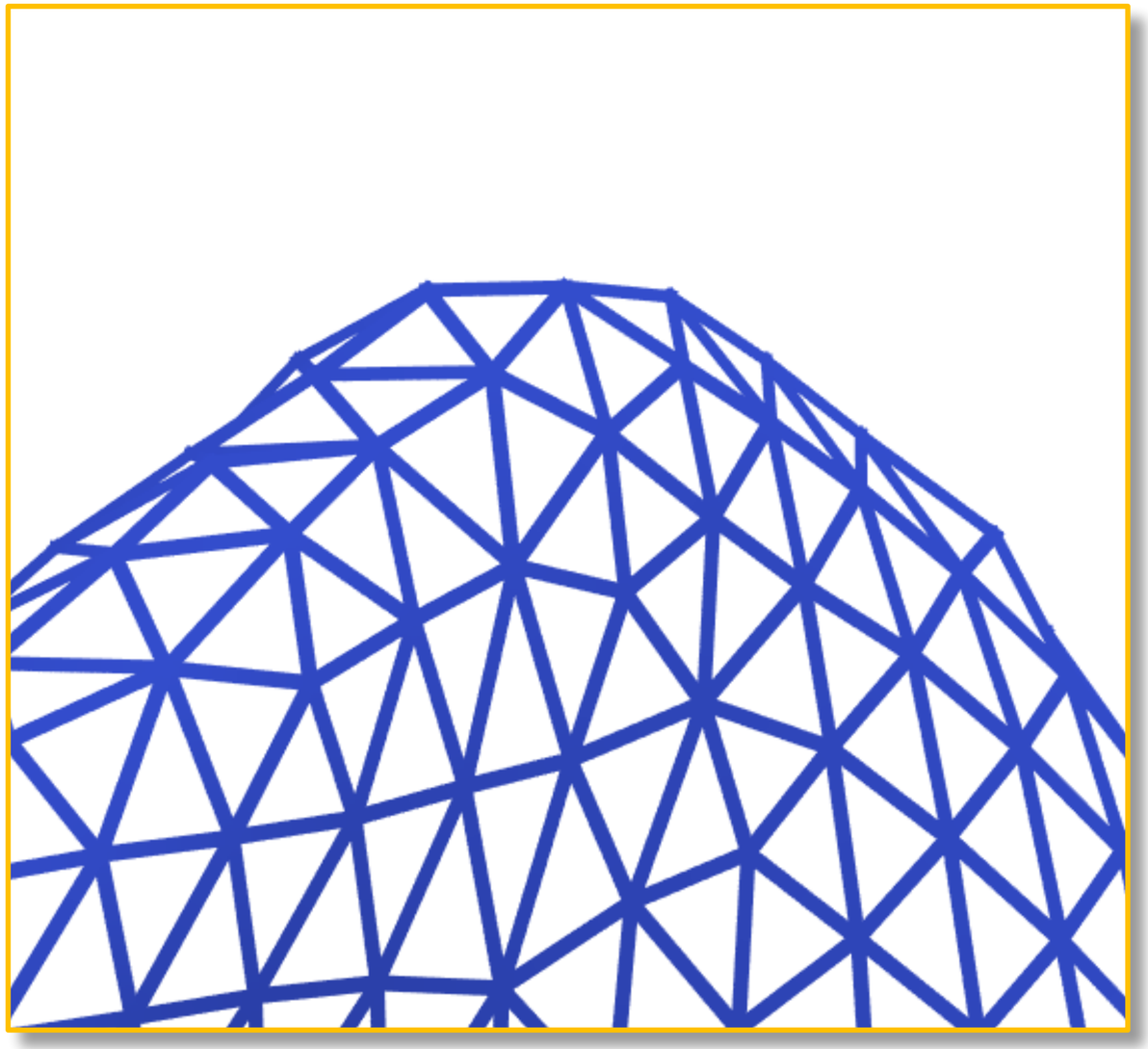


Triangle meshes discretize surfaces…

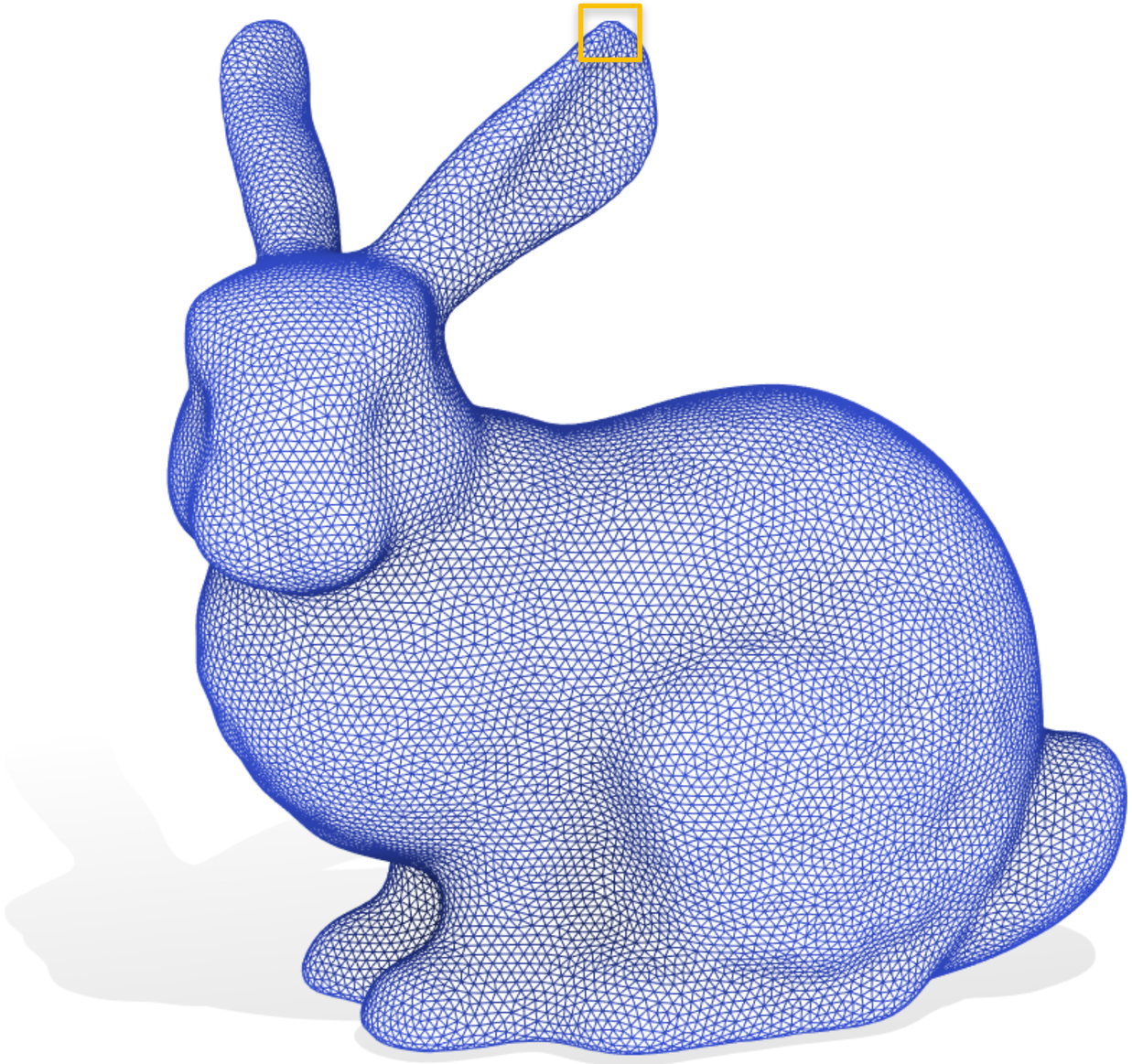


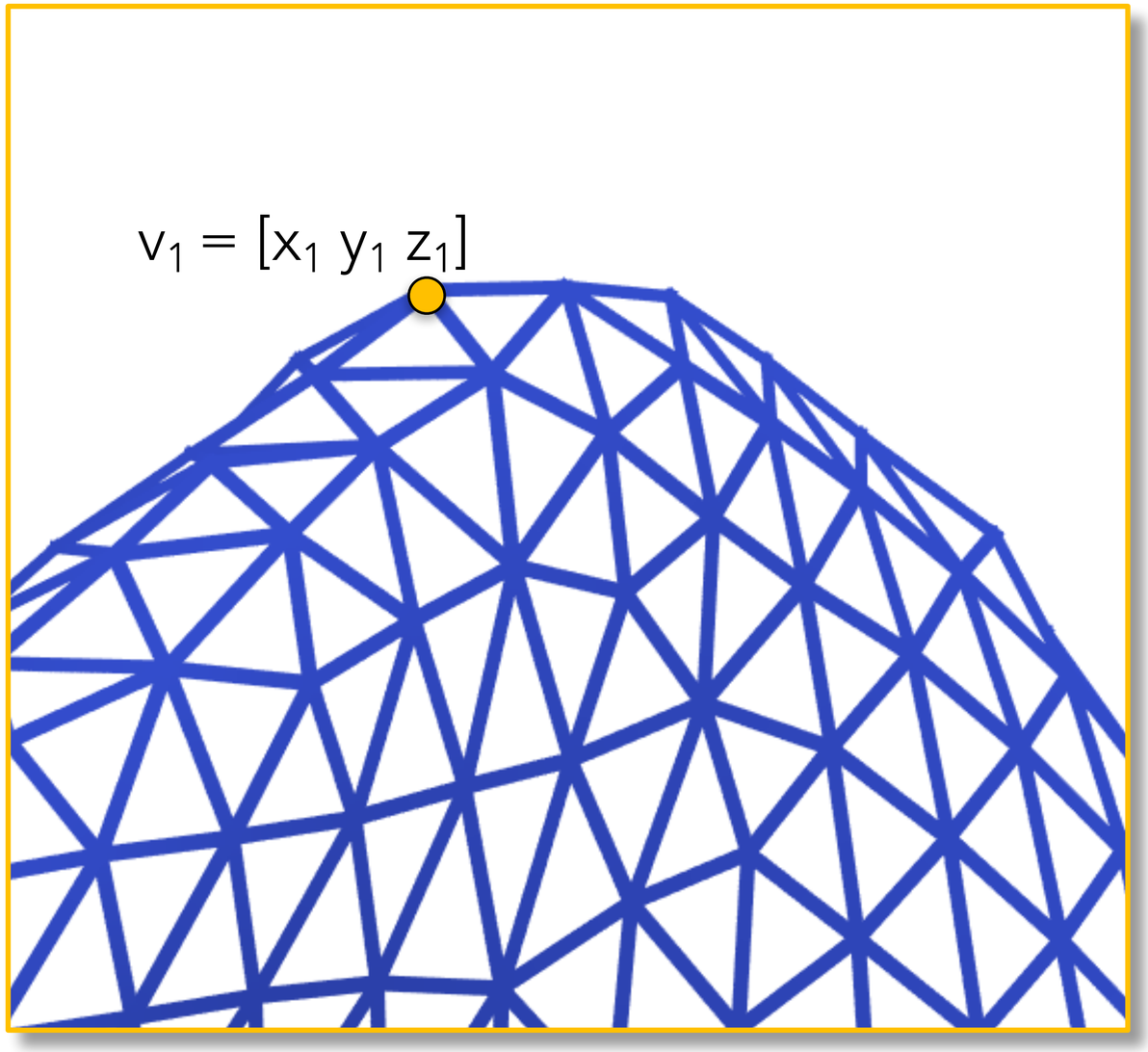
Triangle meshes discretize surfaces…



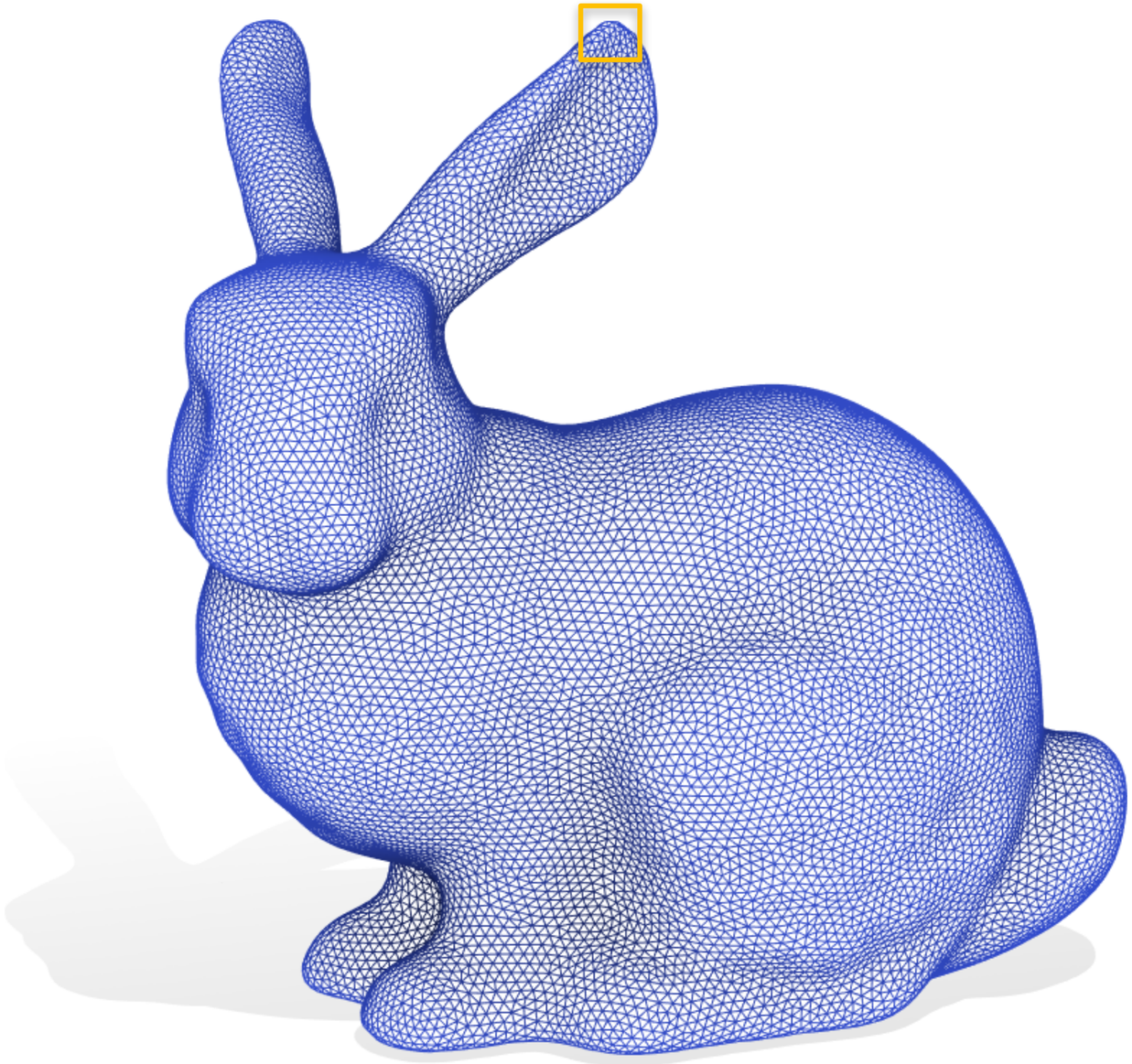


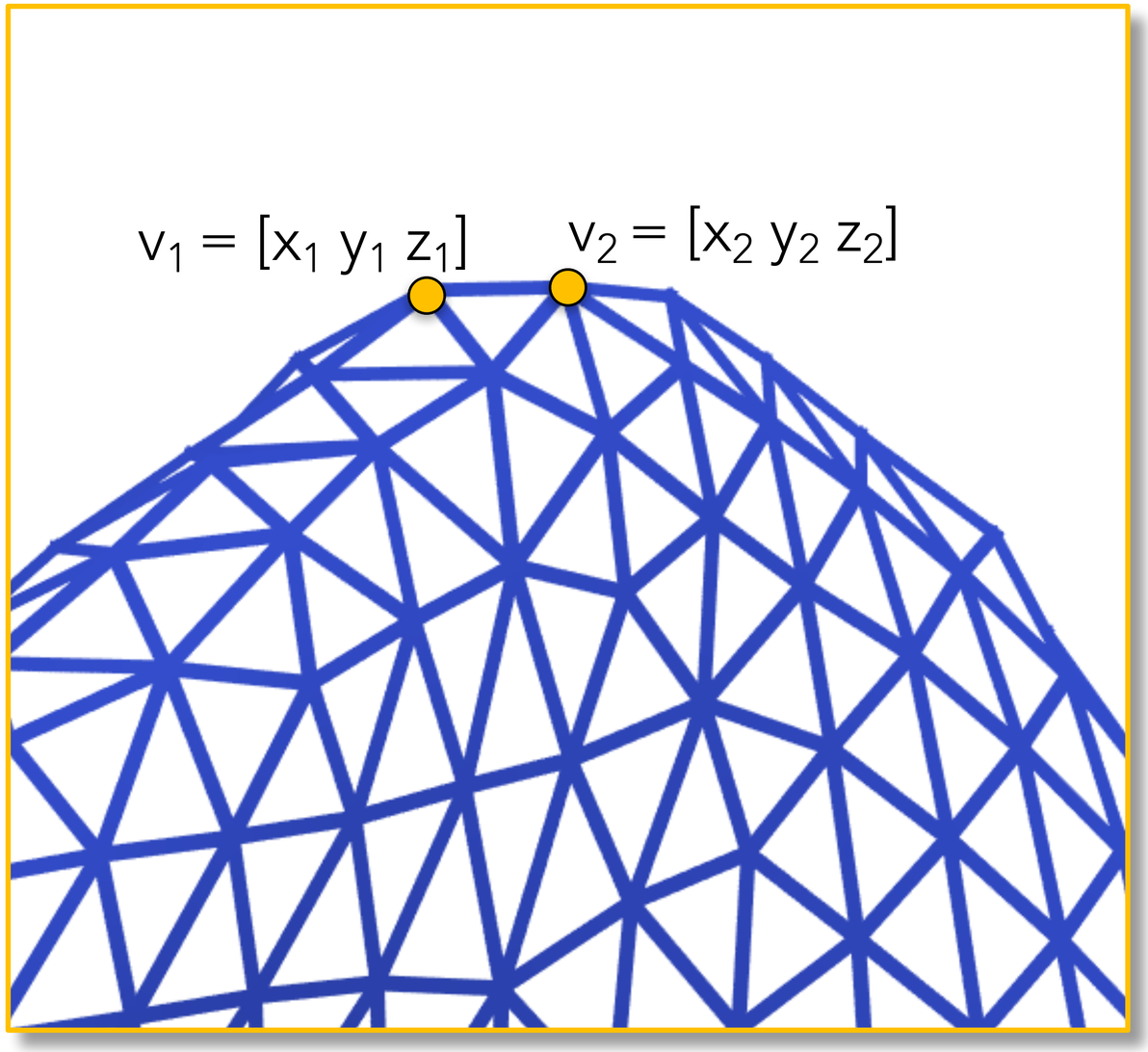
Triangle meshes discretize surfaces…



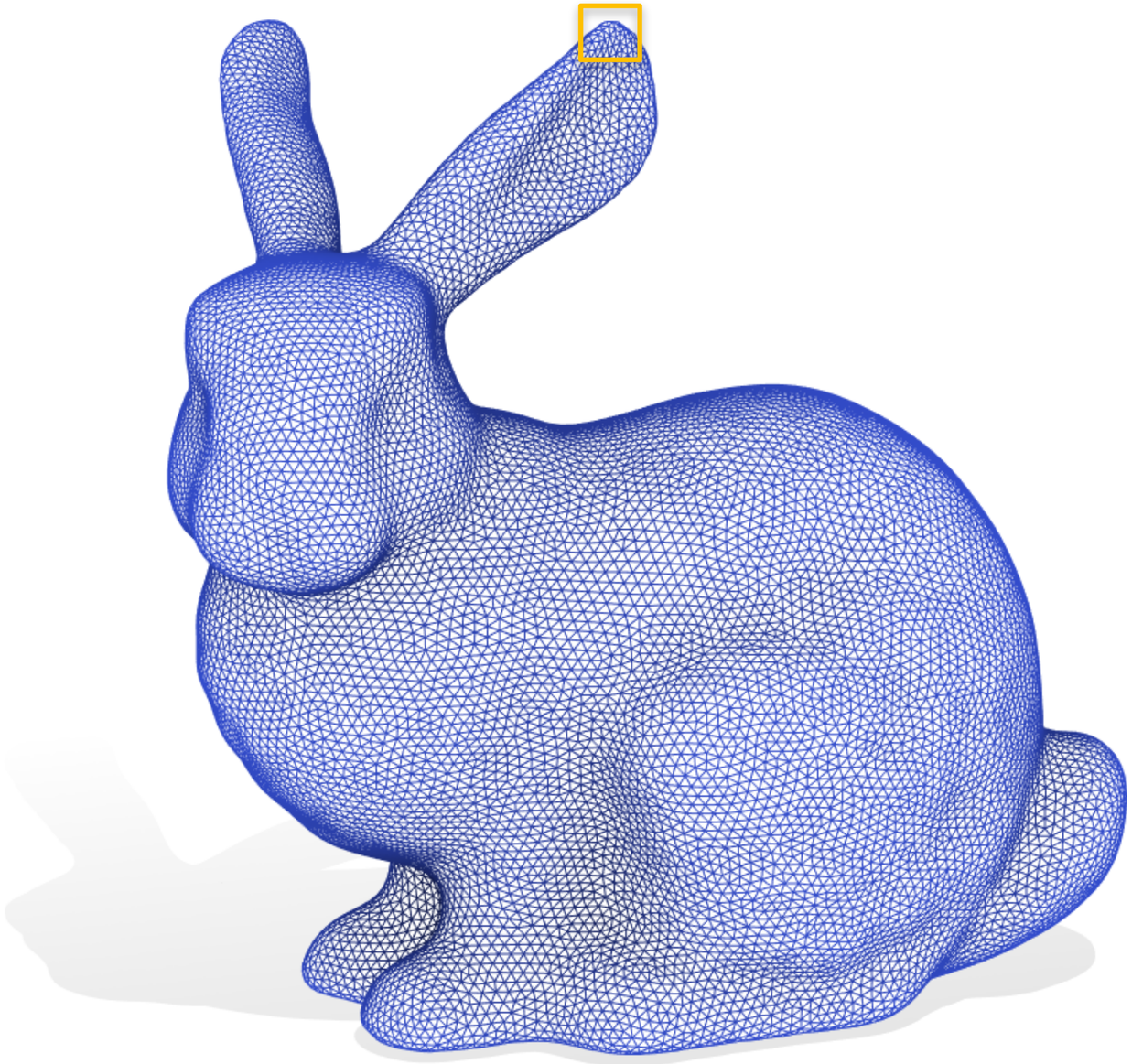


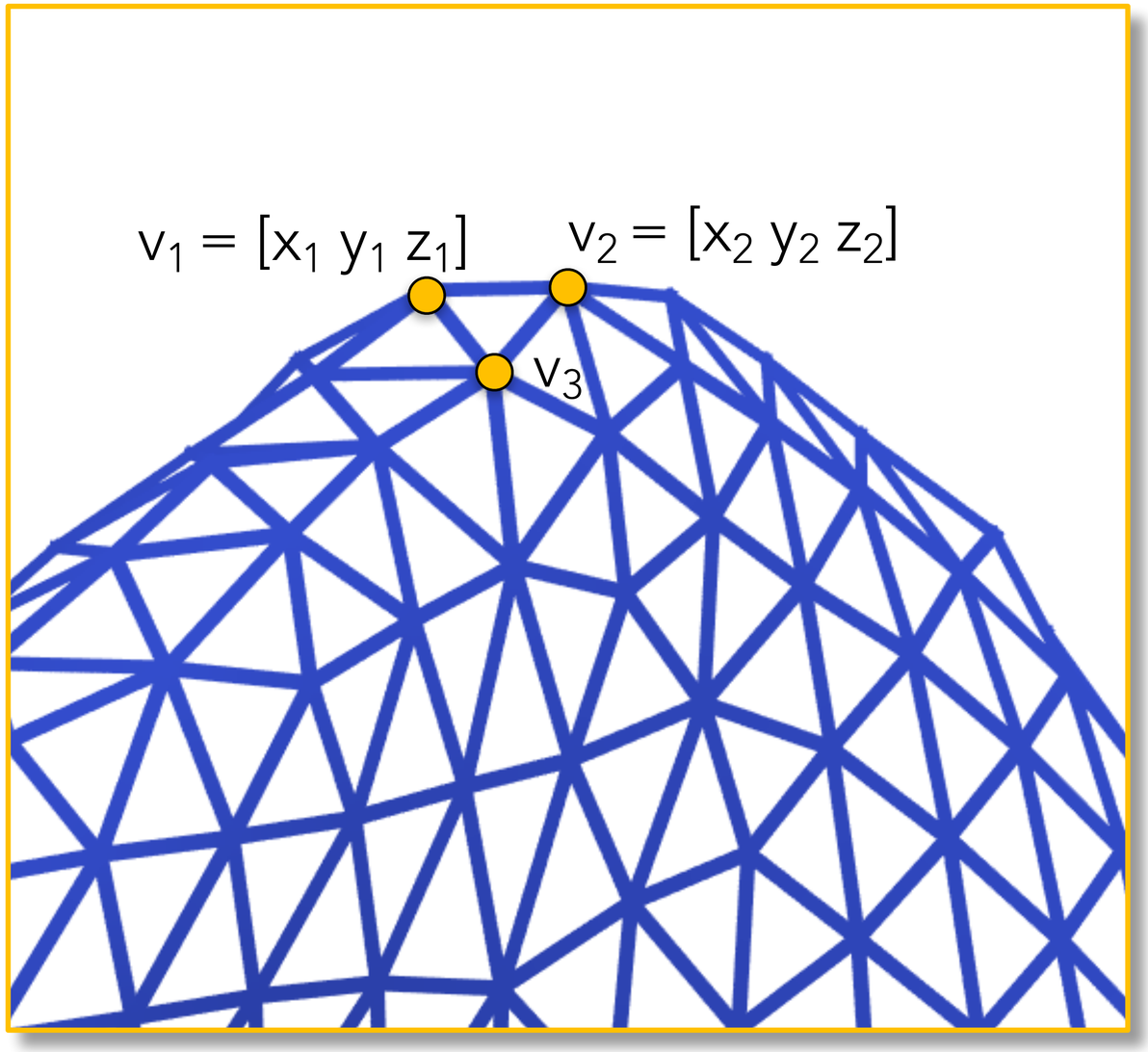
Triangle meshes discretize surfaces…



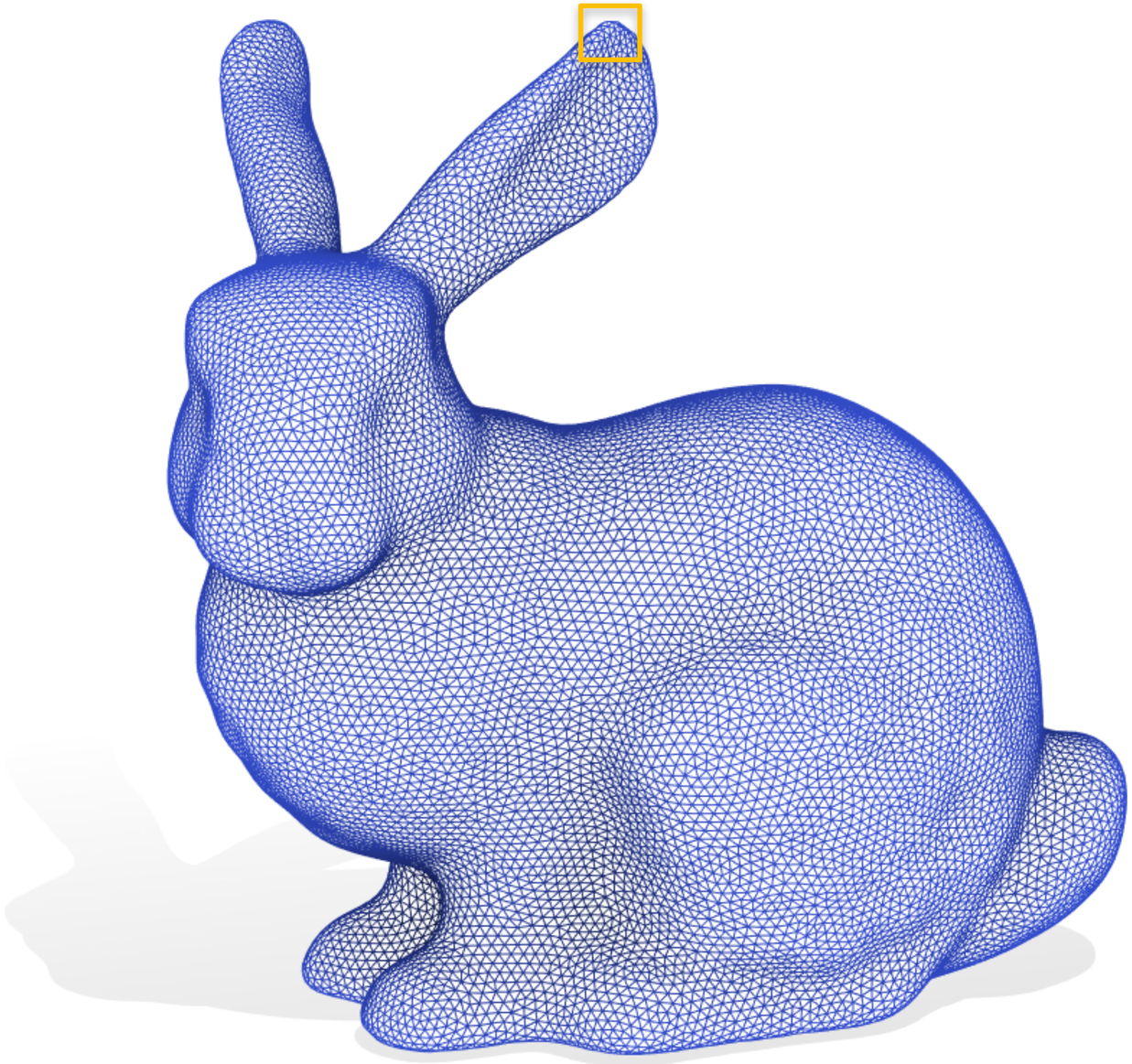


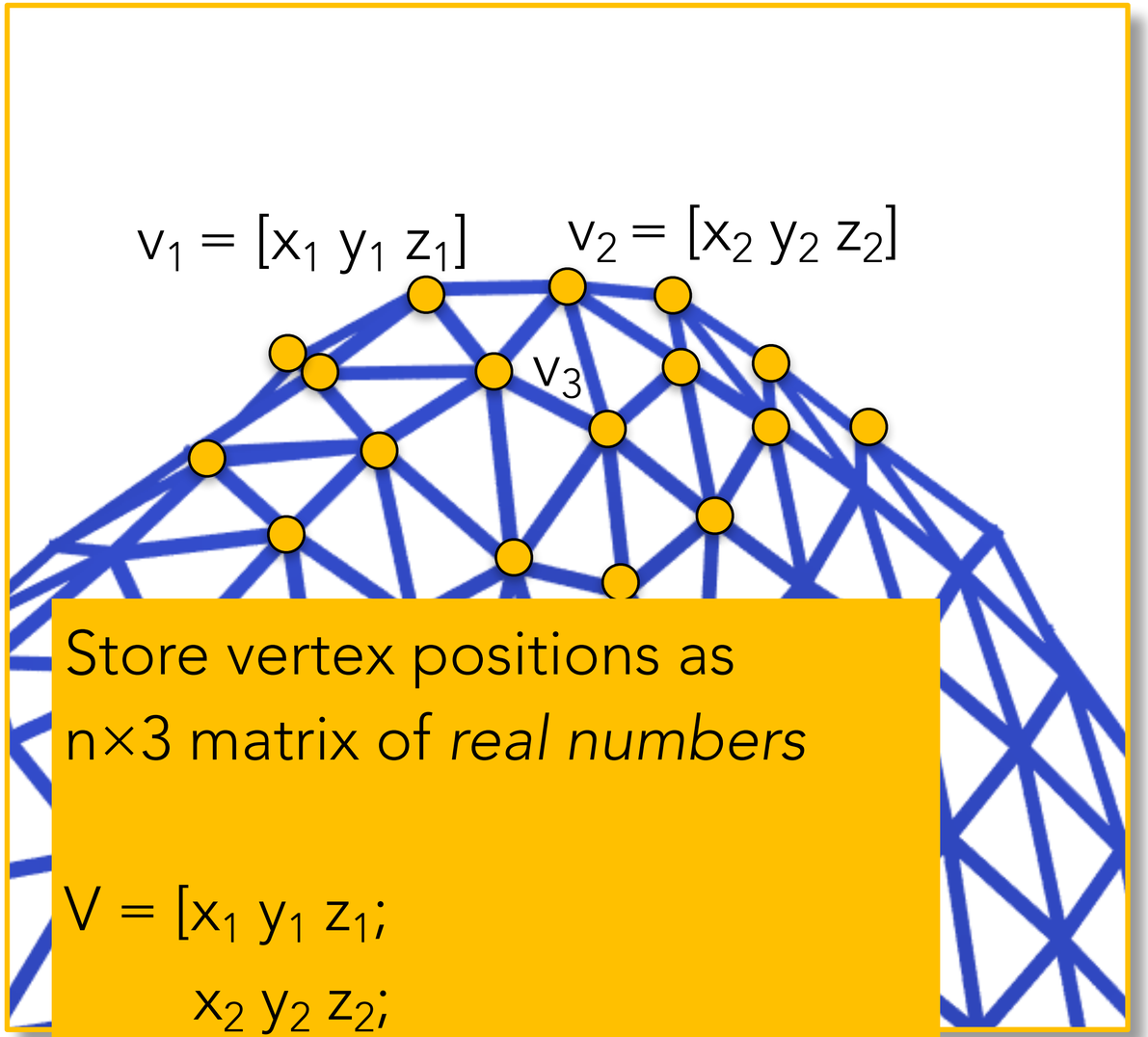
Triangle meshes discretize surfaces…





Triangle meshes discretize surfaces…

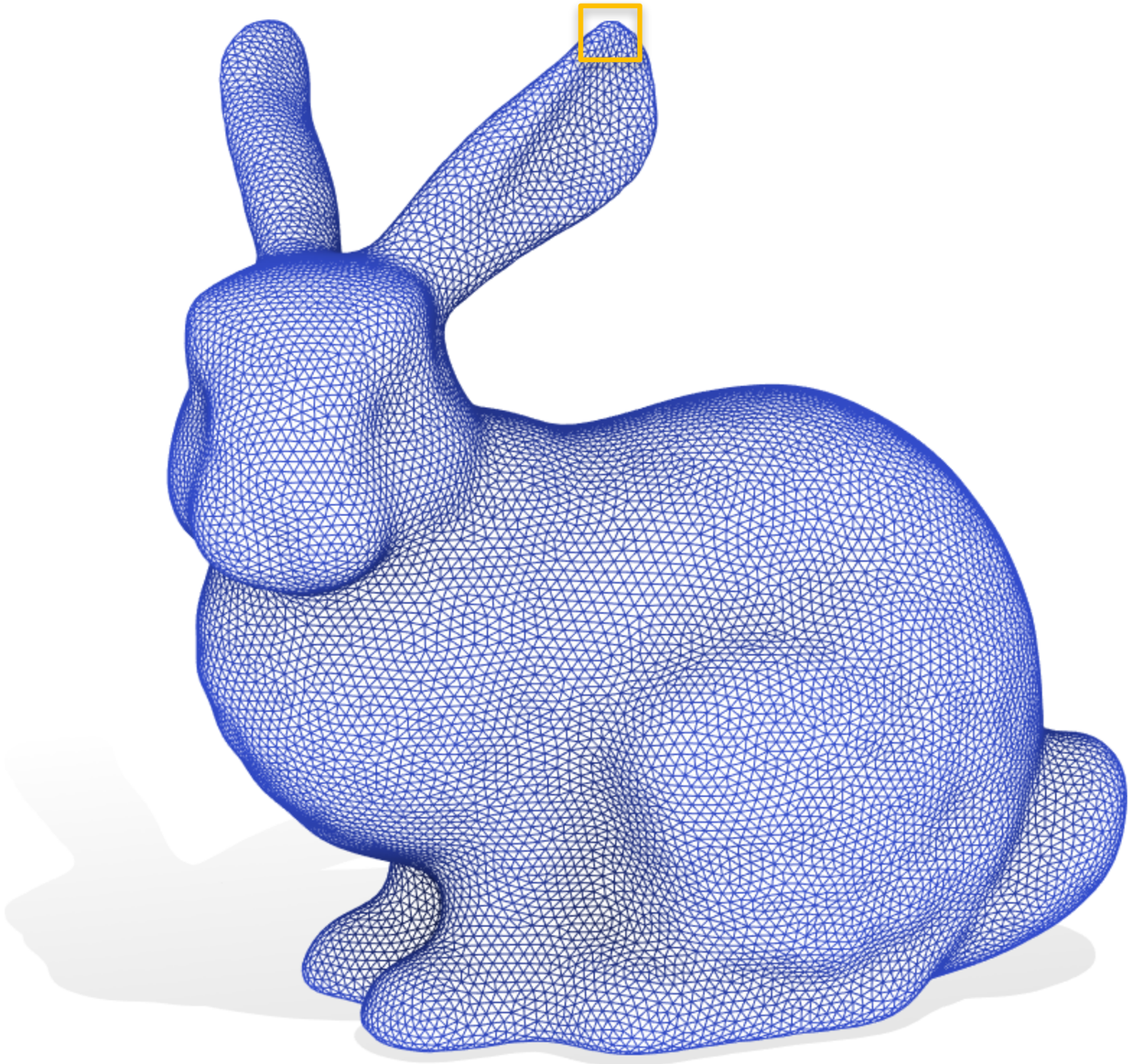


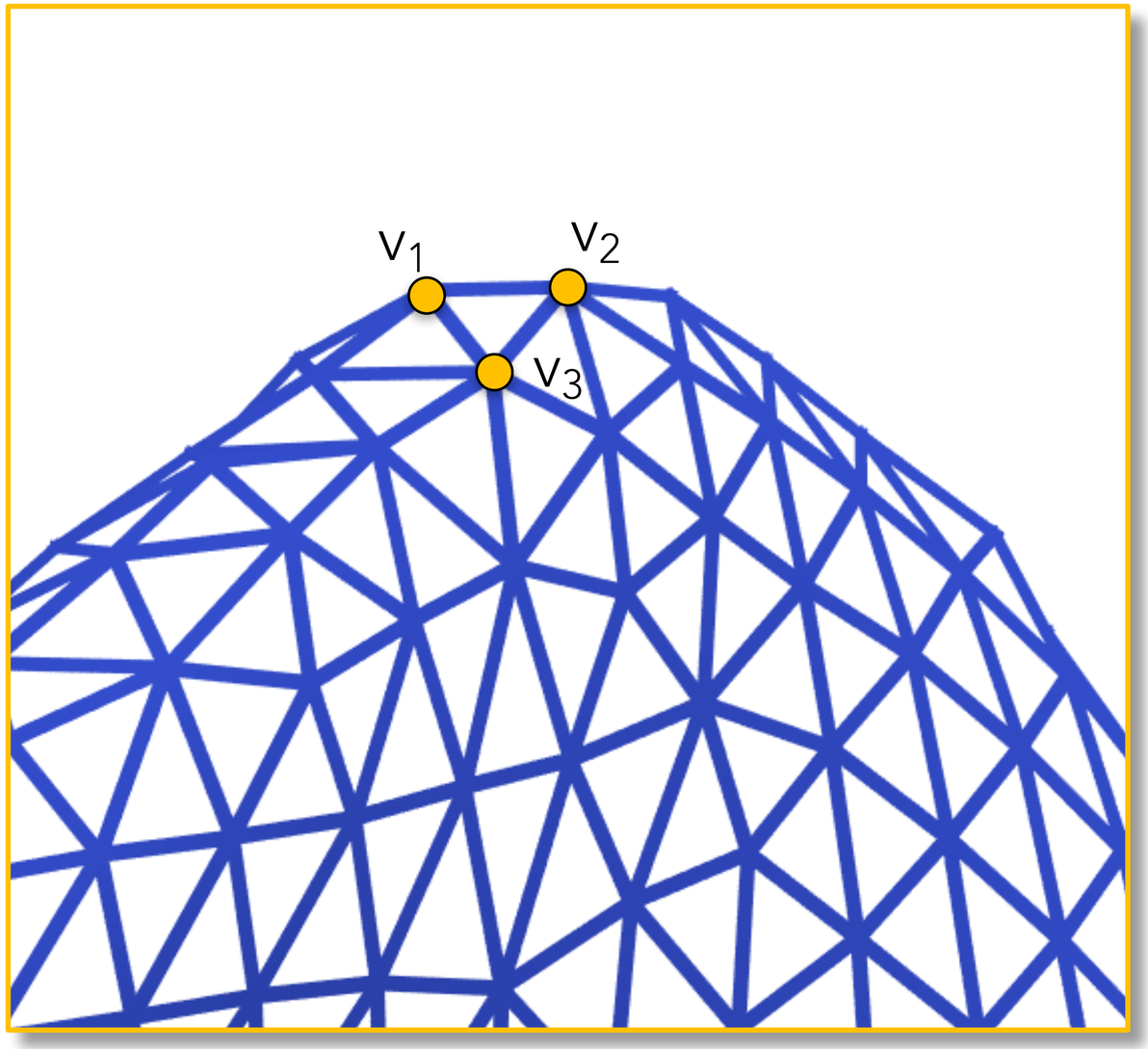


Store vertex positions as
n×3 matrix of *real numbers*

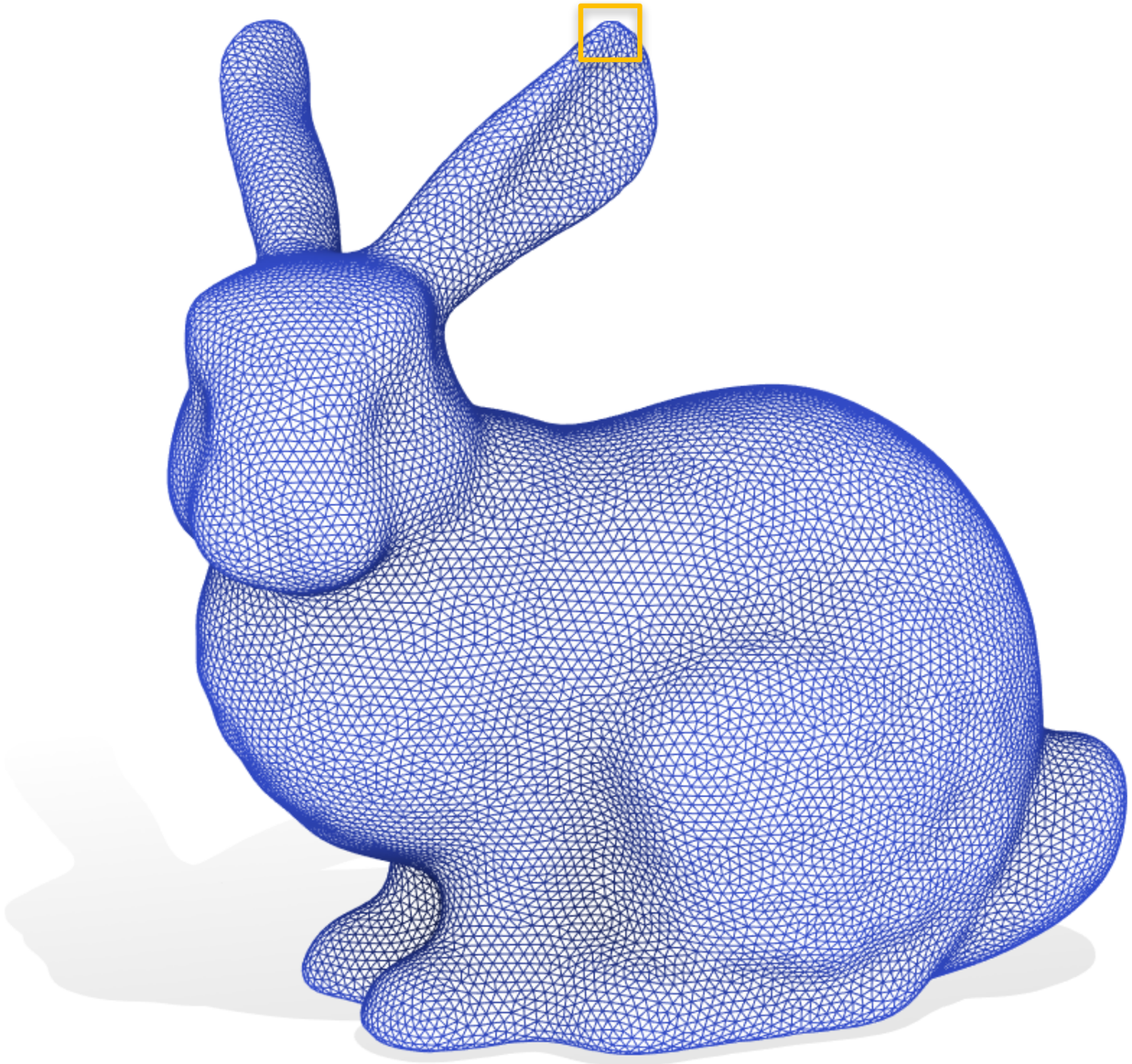
$V = [x_1\ y_1\ z_1;$
$x_2\ y_2\ z_2;$
…
$x_n\ y_n\ z_n]$

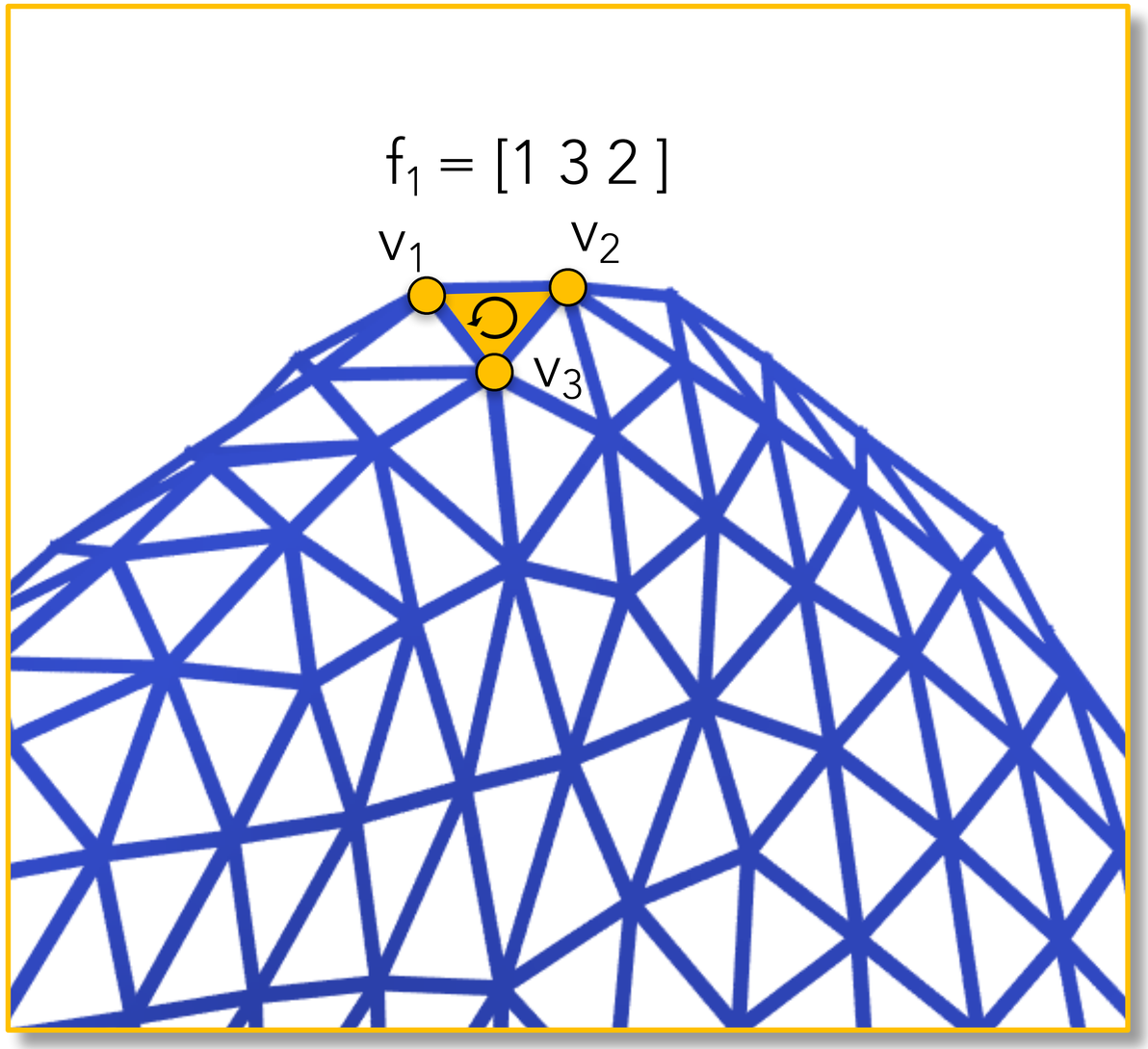
Triangle meshes discretize surfaces…



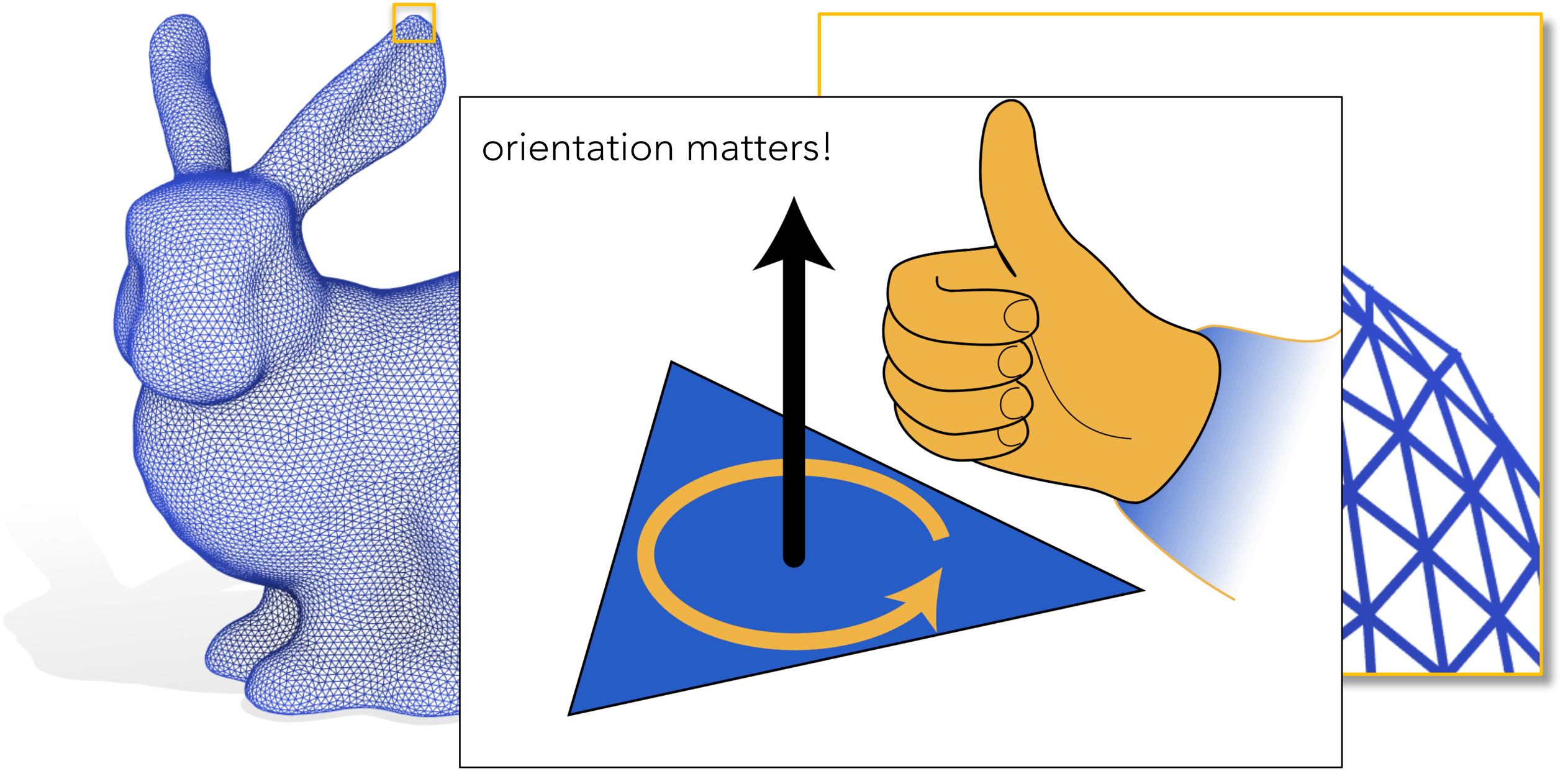


Triangle meshes discretize surfaces…

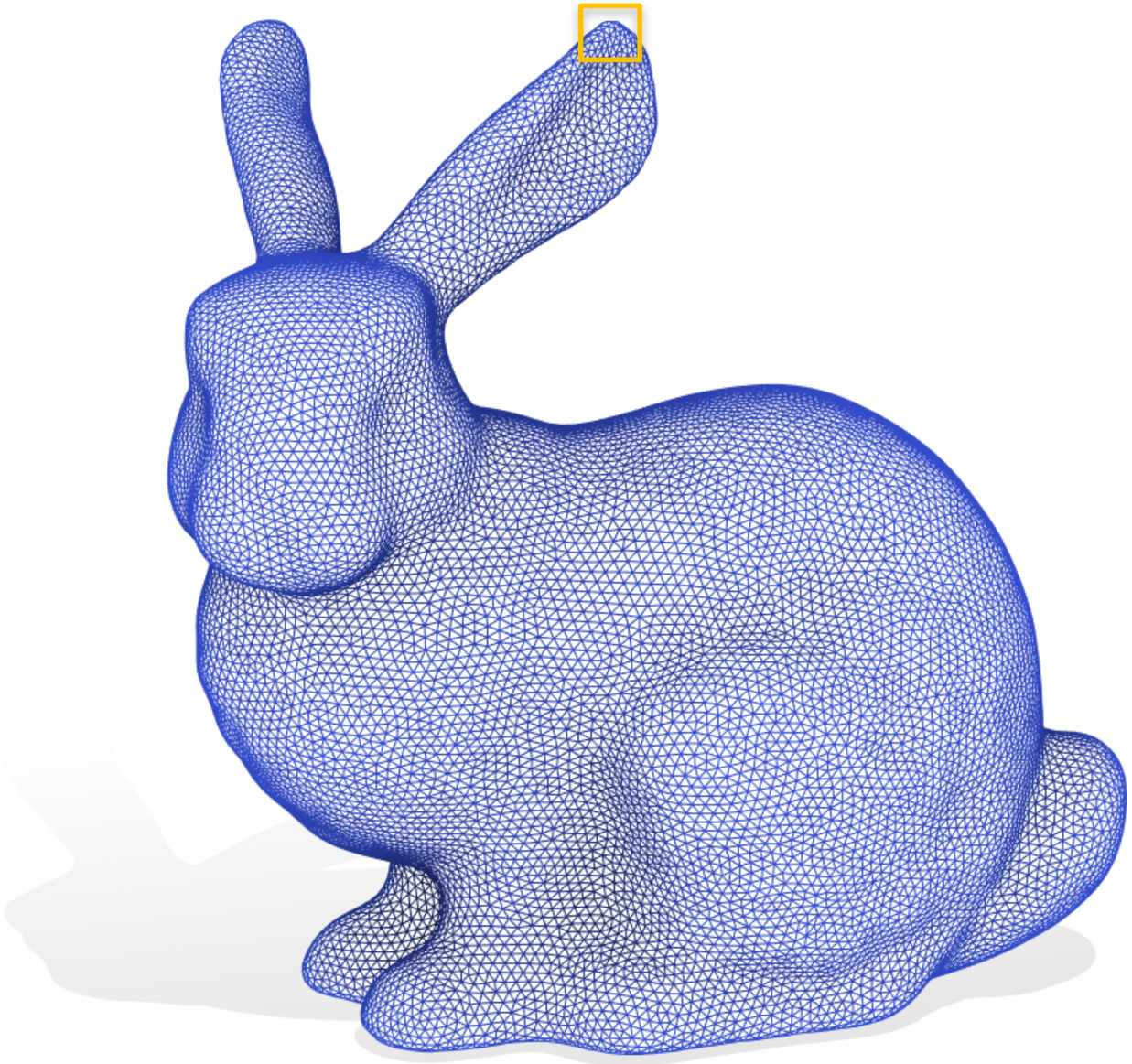


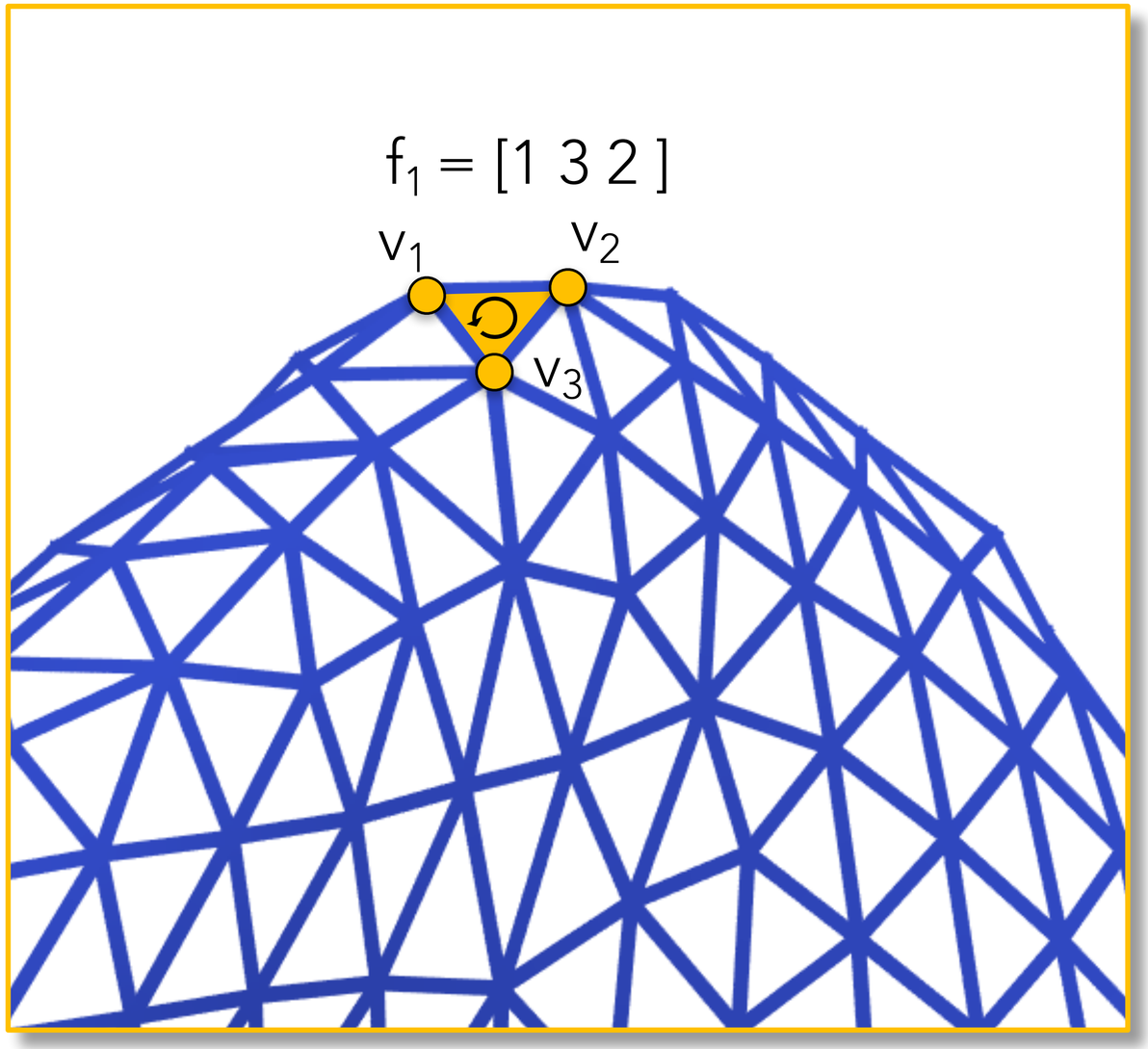


Triangle meshes discretize surfaces…

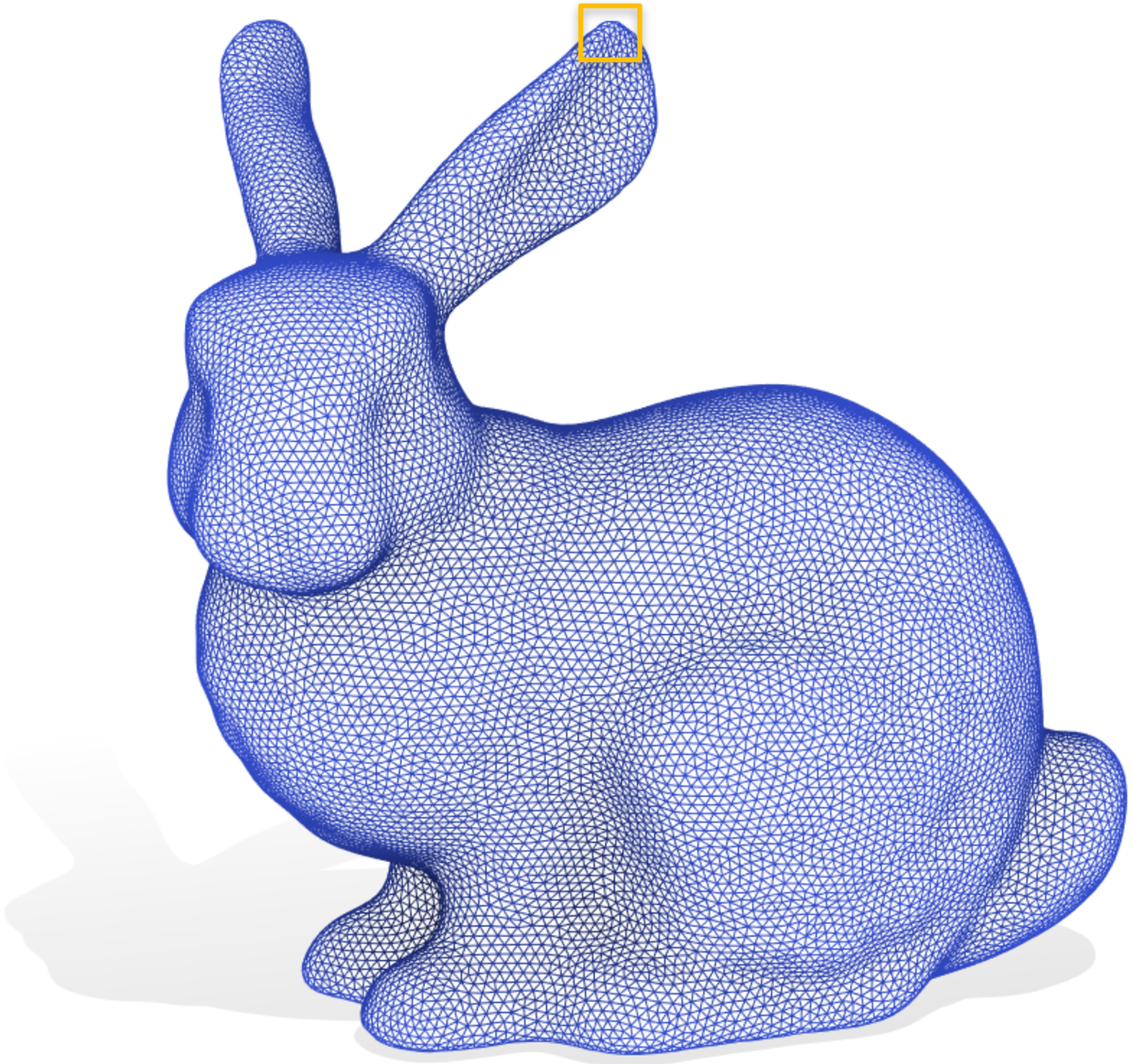


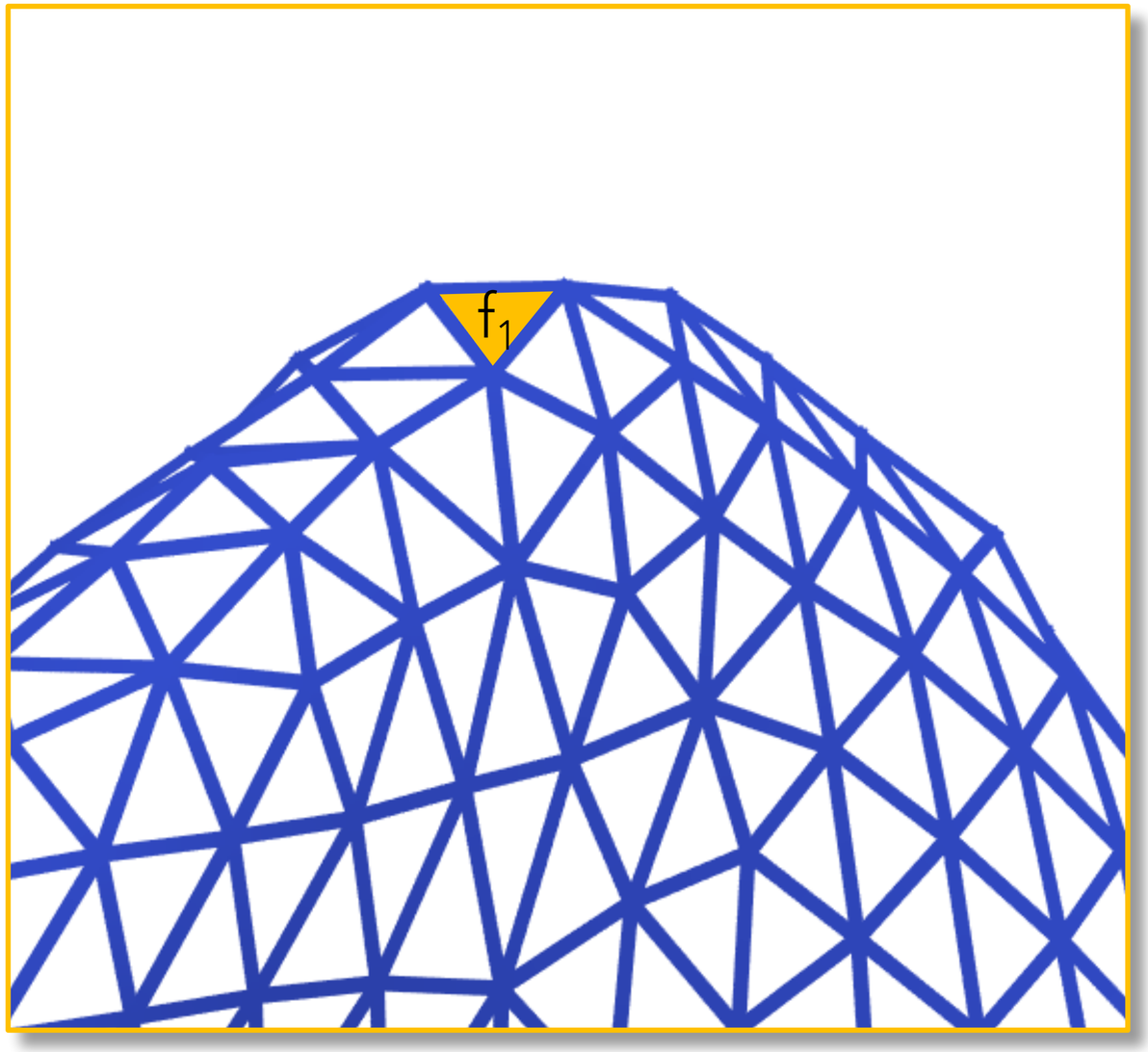
Triangle meshes discretize surfaces…



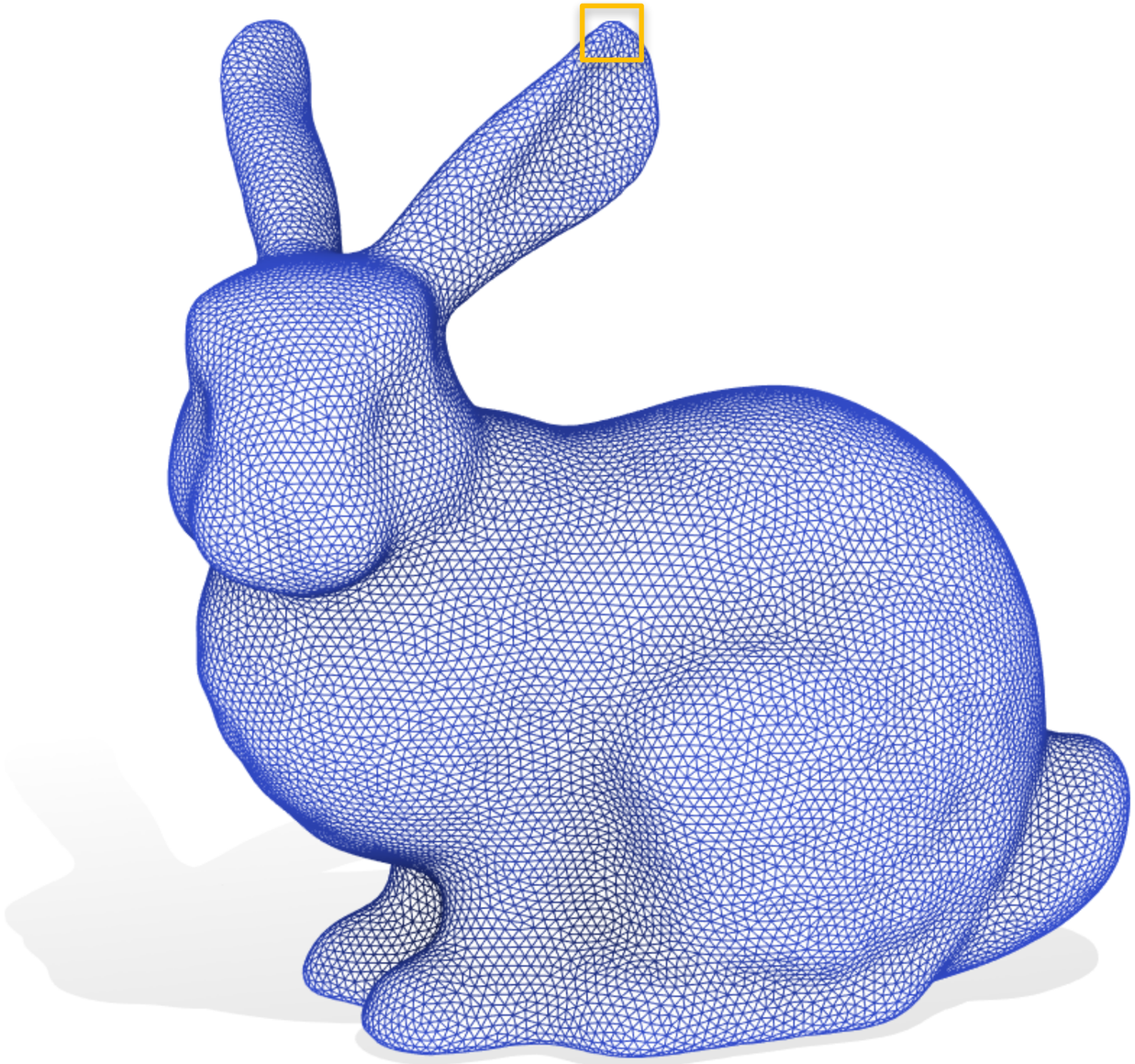


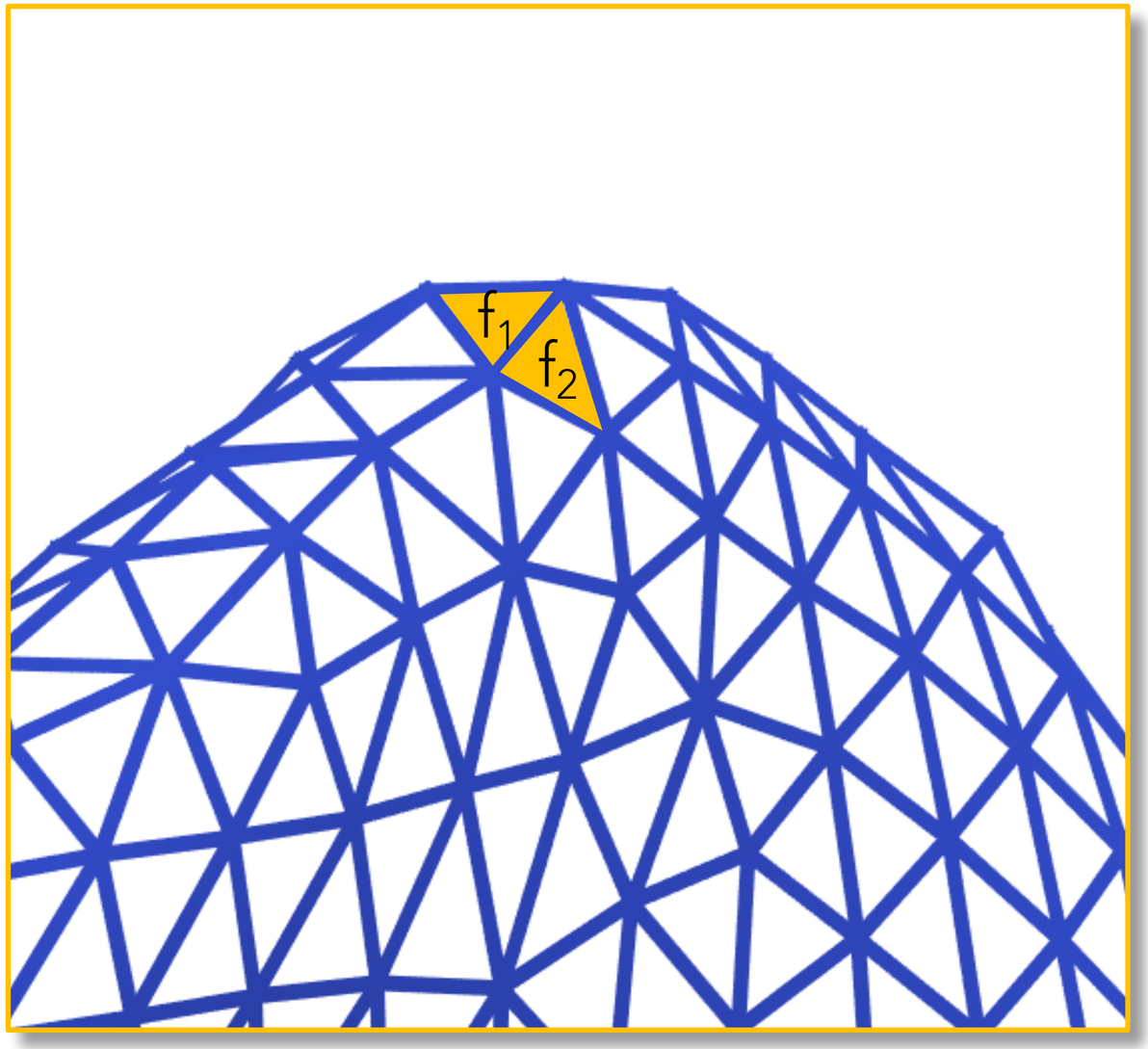
Triangle meshes discretize surfaces…



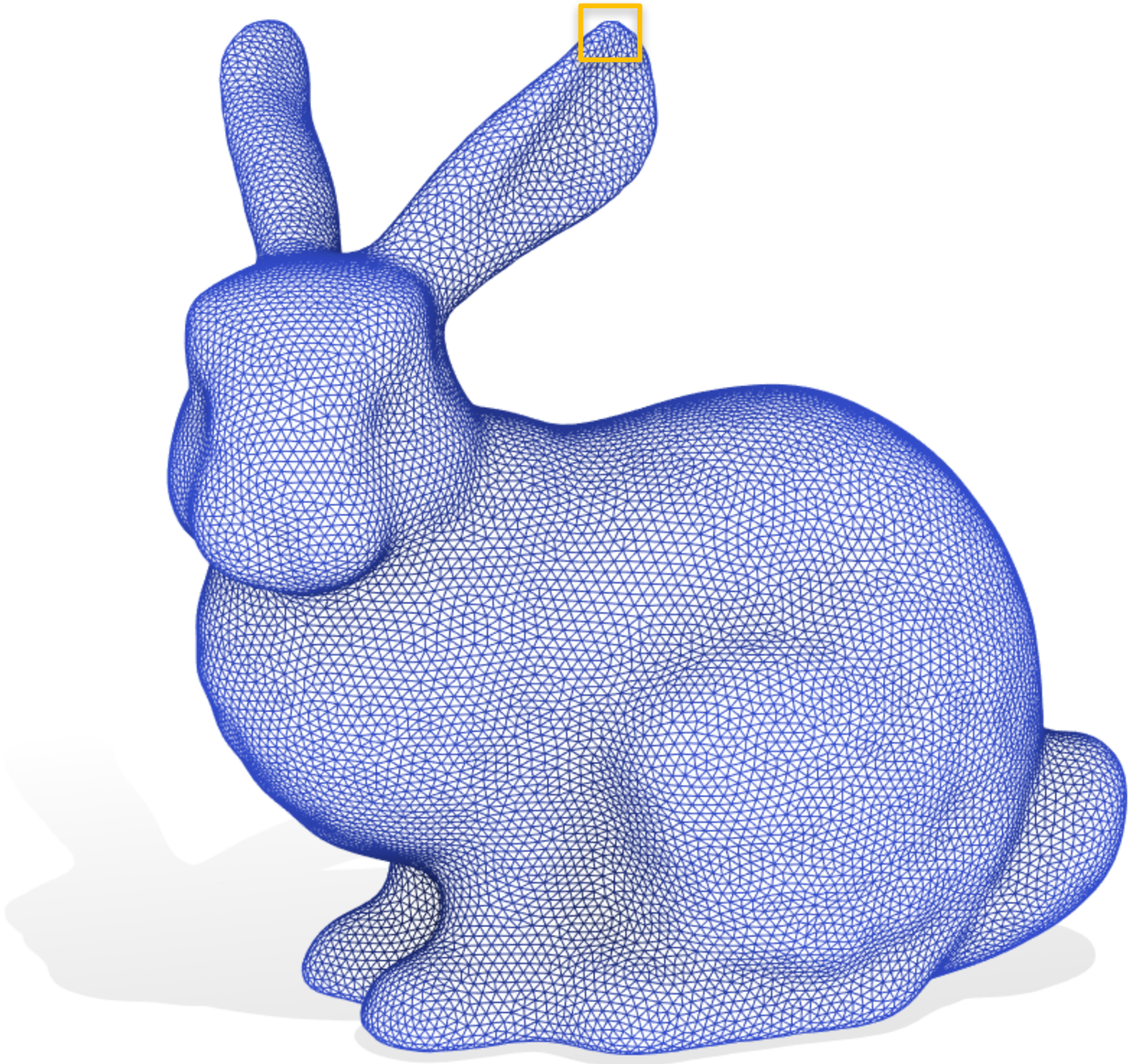


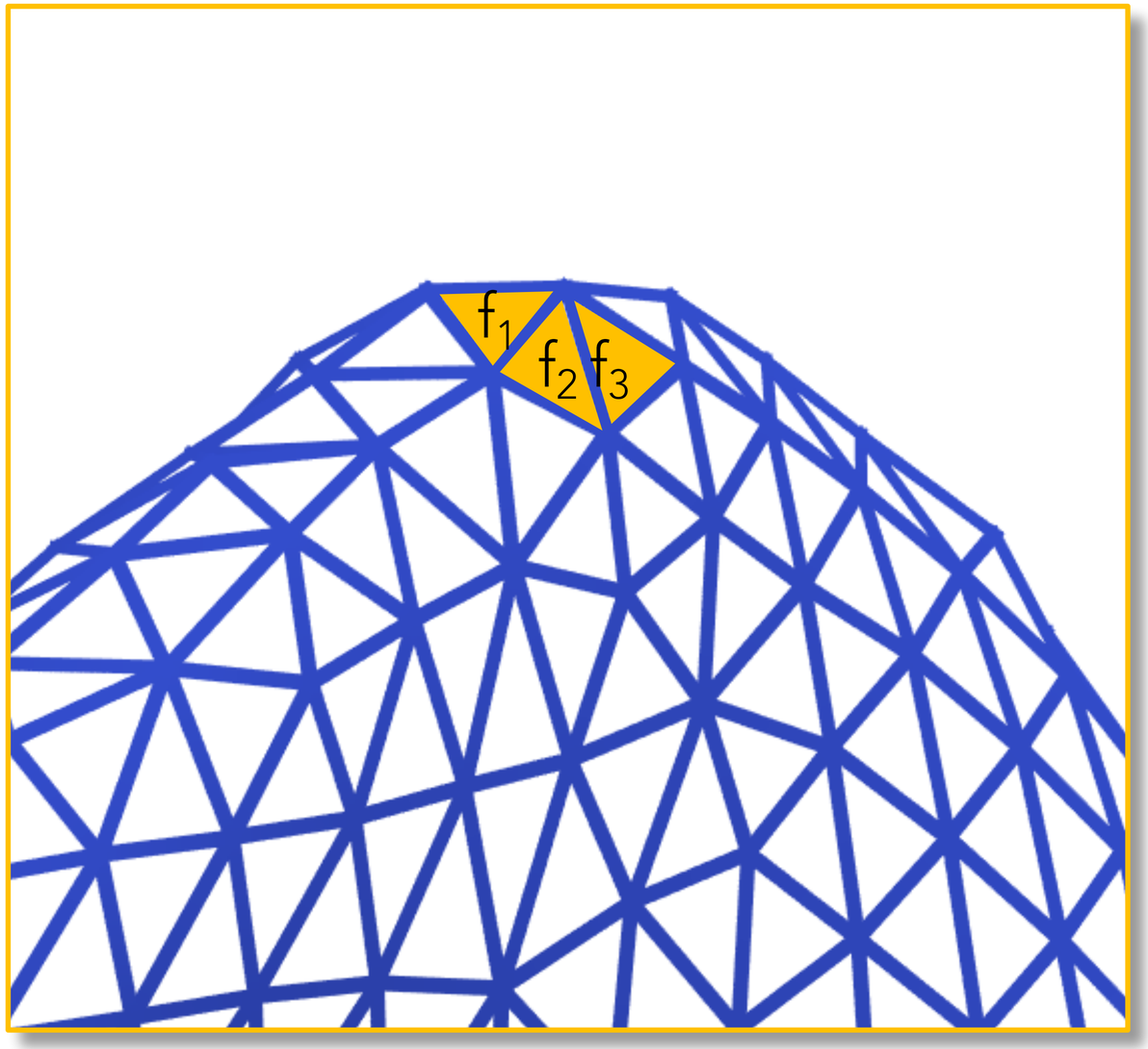
Triangle meshes discretize surfaces…



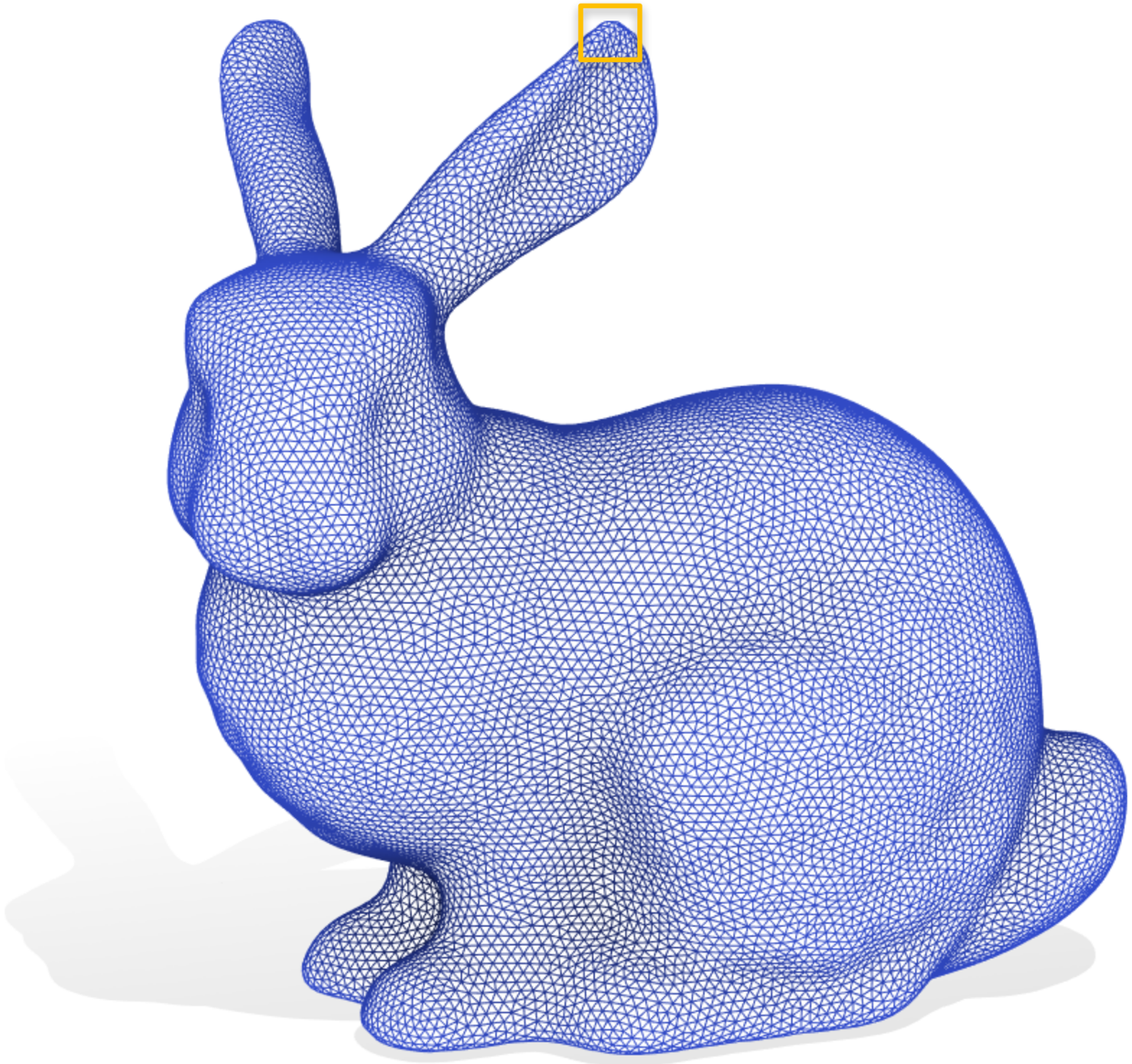


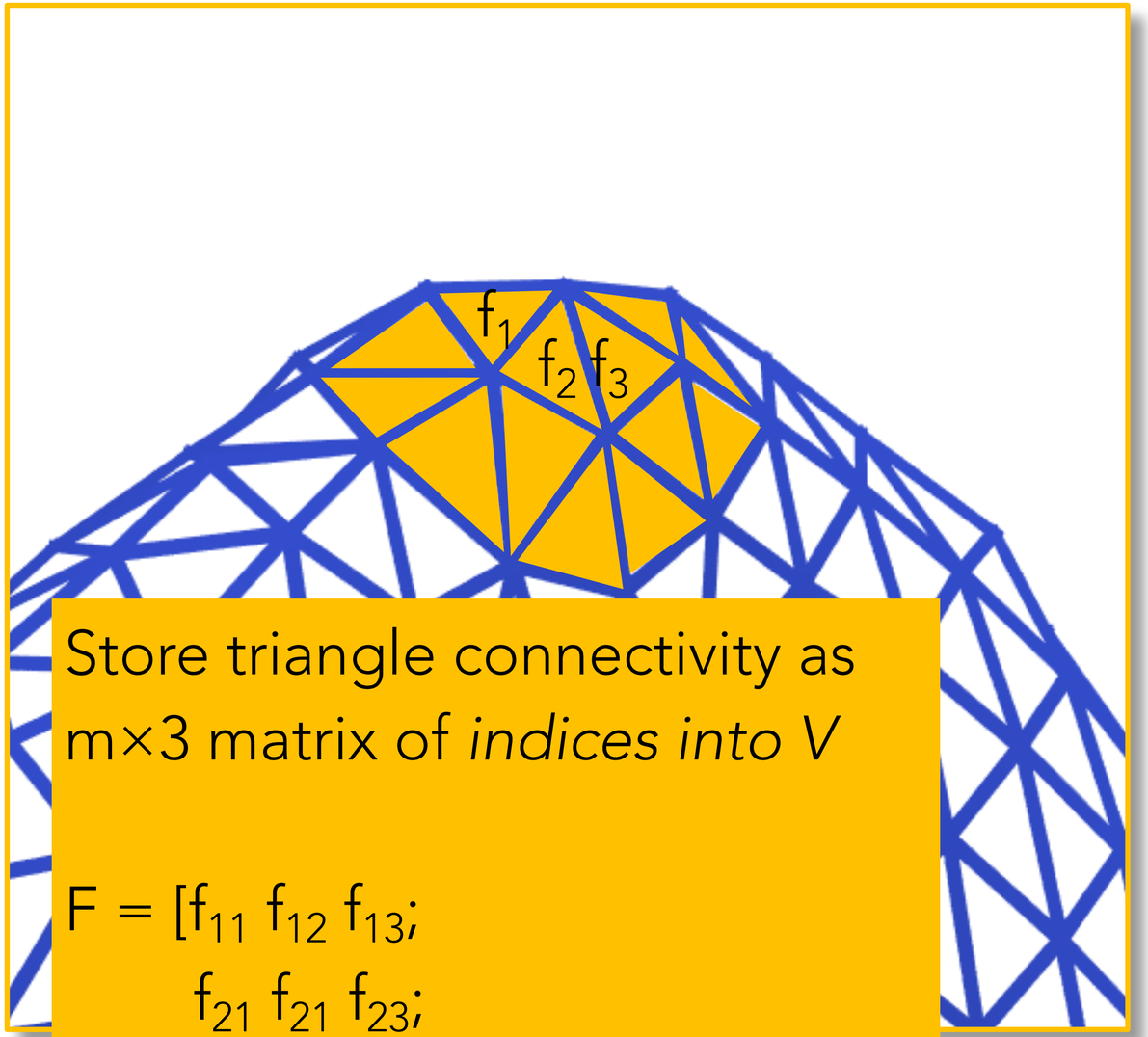
Triangle meshes discretize surfaces…





Triangle meshes discretize surfaces…





Store triangle connectivity as m×3 matrix of *indices into V*

$$F = \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{21} & f_{23} \\ \dots & & \\ f_{n1} & f_{n2} & f_{n3} \end{bmatrix}$$

Why RAW matrices?

- Memory efficient and cache friendly
- Indices are simpler to debug than pointers
- Trivially copied and serialized
- **Interchangeable** with other libraries: **numpy**, pyTorch, Tensorflow, Scipy, MATLAB, OpenCV

Getting Started

https://geometryprocessing.github.io/geometric-computing-python/

Closing Remarks

Roadmap

- We are developing this infrastructure to support our own research, we hope to reach a wide coverage of **wildmeshing** and **libigl** within the next 6 months

- **Polyfem** is in development, and new features will be exposed to python as they are added to the C++ version

- We are working on a physically-based rendering module for **meshplot**

Please let us know if you use it!

- Please cite the course notes in your papers if you use any of these python libraries.
- If you want to contribute, PRs are very welcome!
- For questions/suggestions/complaints please open issues on the corresponding GitHub repo

Black-Box Analysis:

From Theory to Practice

Teseo Schneider

NSF
Adobe
FNSNF
Swiss National Science Foundation
nTopology

NYU COURANT INSTITUTE OF MATHEMATICAL SCIENCES