A Future of Polygon Reduction

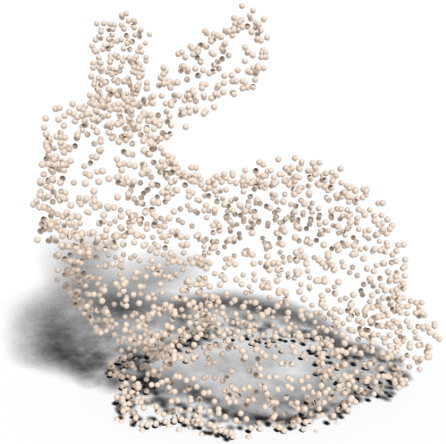
Changkun Ou
changkun.de/s/polyred4us
@changkun

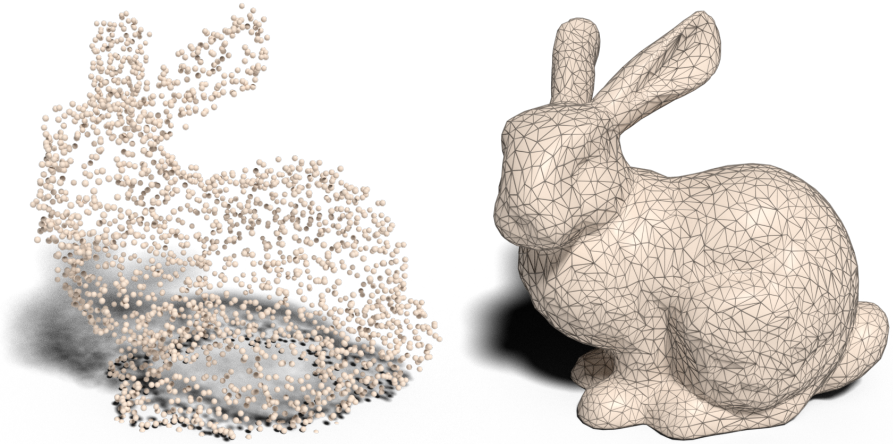
IDC 2020 Autumn
Venice, Italy
Oct 7, 2020

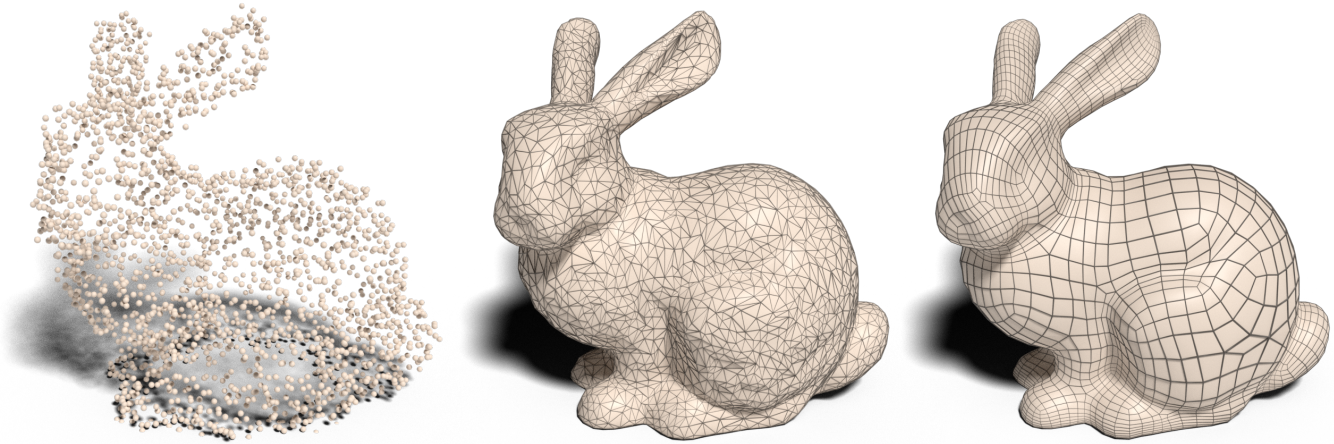
Previously on Polygon Reduction (Polyred)...

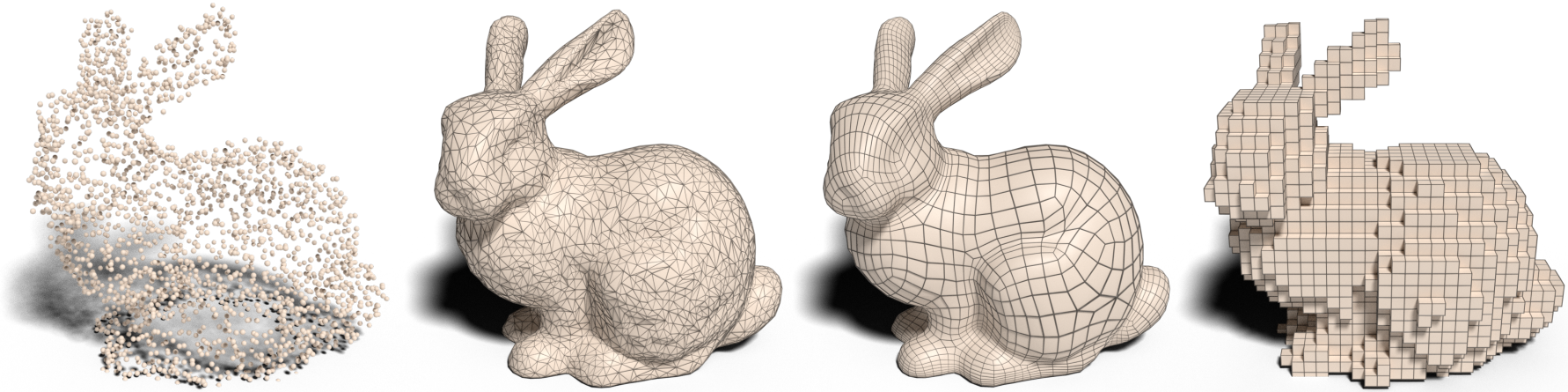
Previously on Polygon Reduction (Polyred)...

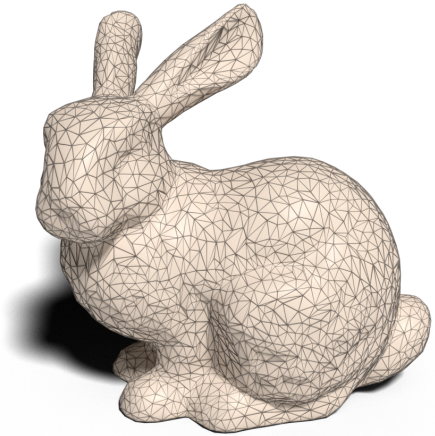
changkun.de/s/polyred1step (2019a)

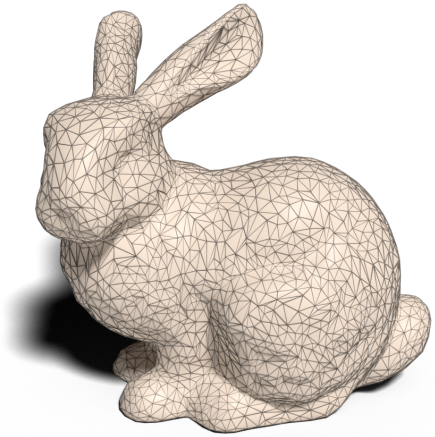


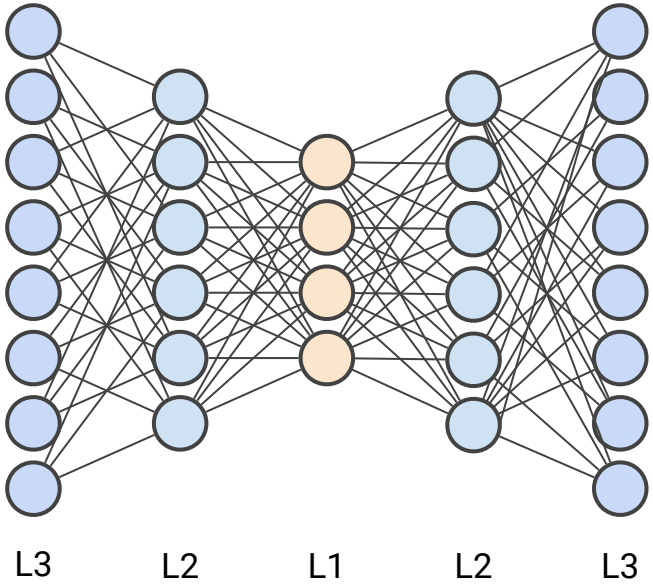


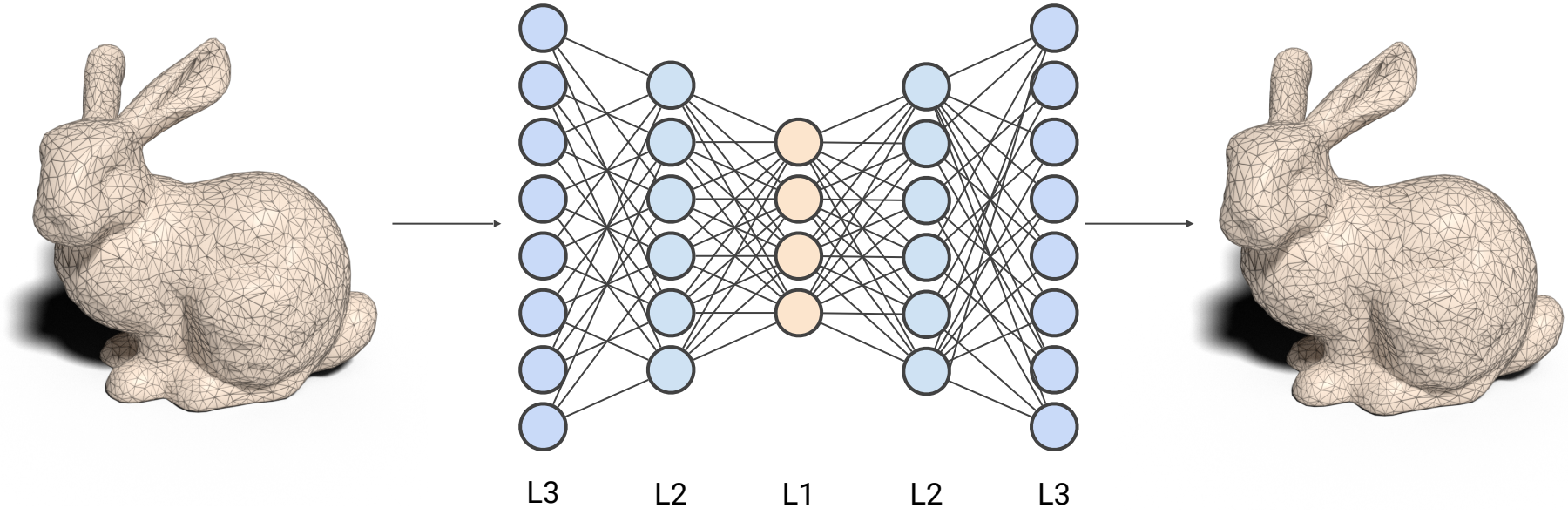


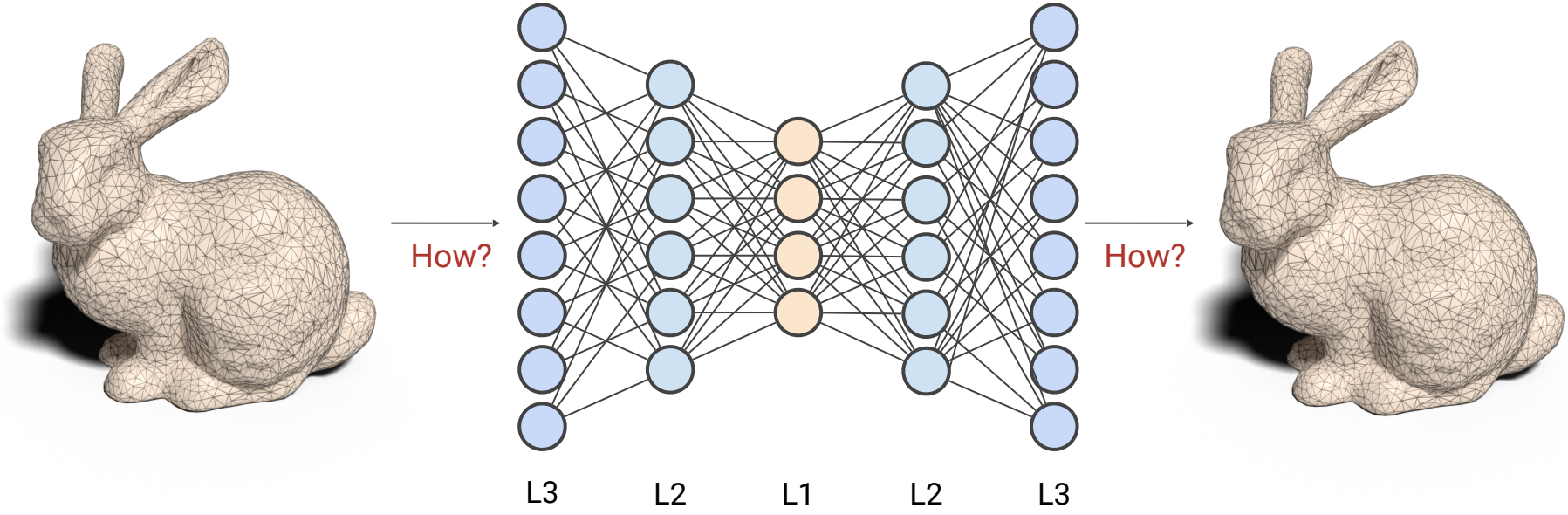






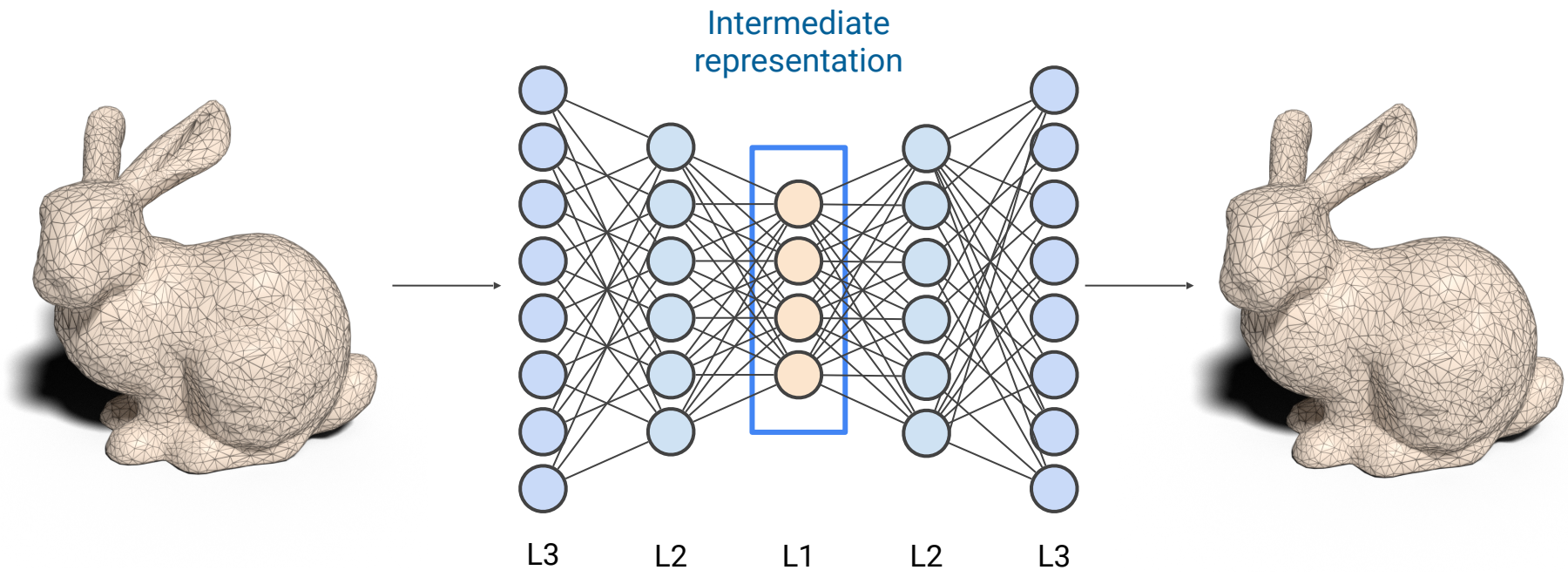

L3
L2
L1
L2
L3

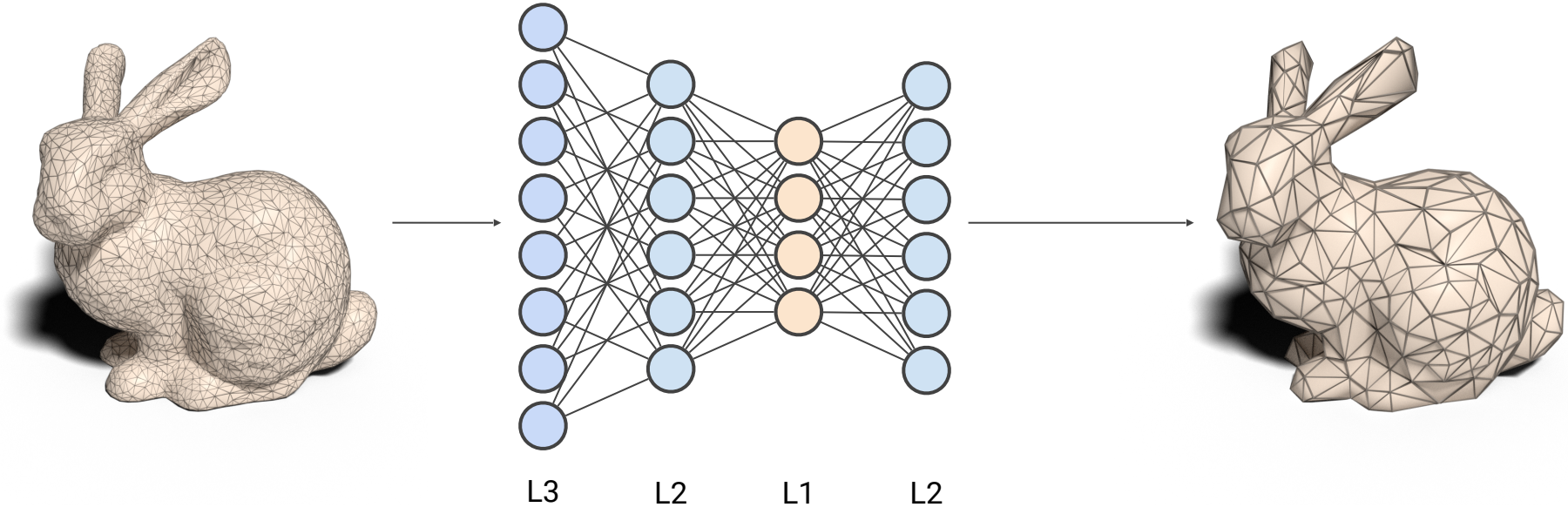

L3
L2
L1
L2
L3


How?
How?
L3
L2
L1
L2
L3

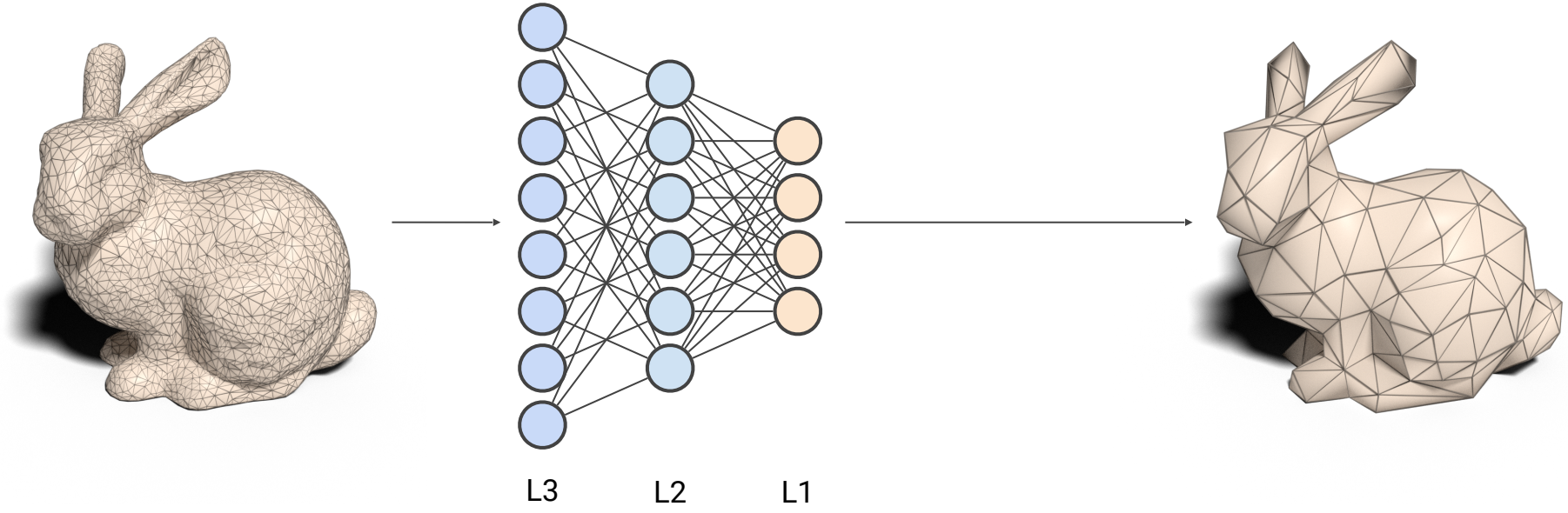
Previously on Polygon Reduction (Polyred)...

changkun.de/s/polyred1step (2019a)

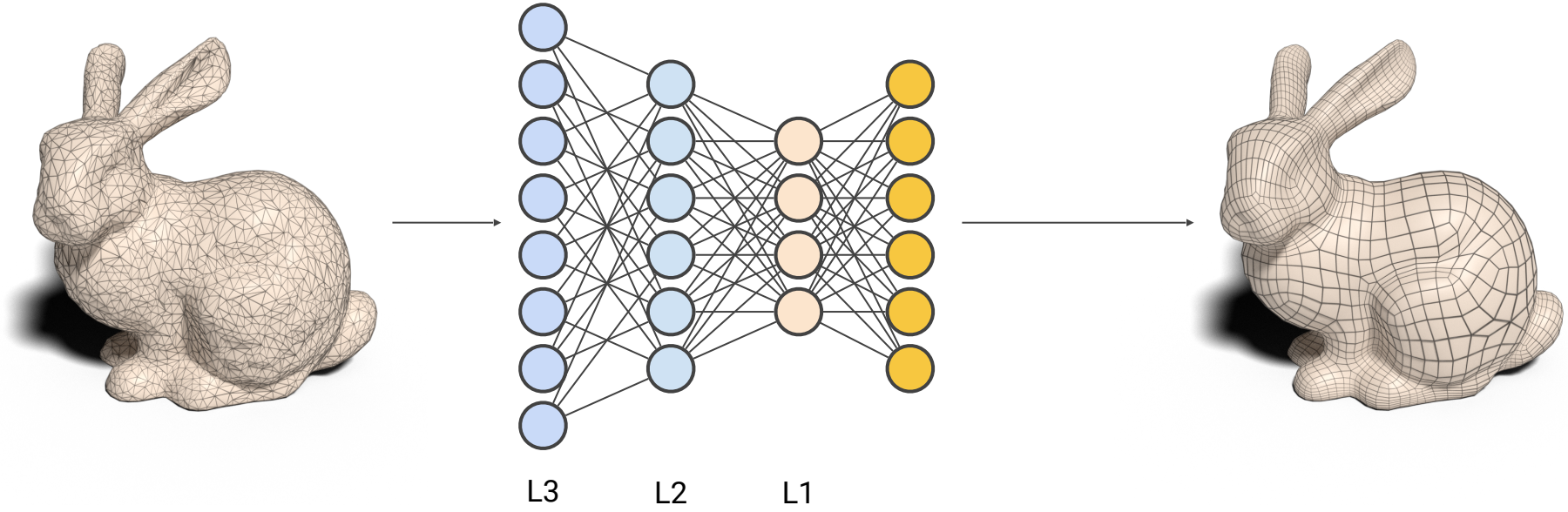

Intermediate
representation
L3
L2
L1
L2
L3


L3
L2
L1
L2

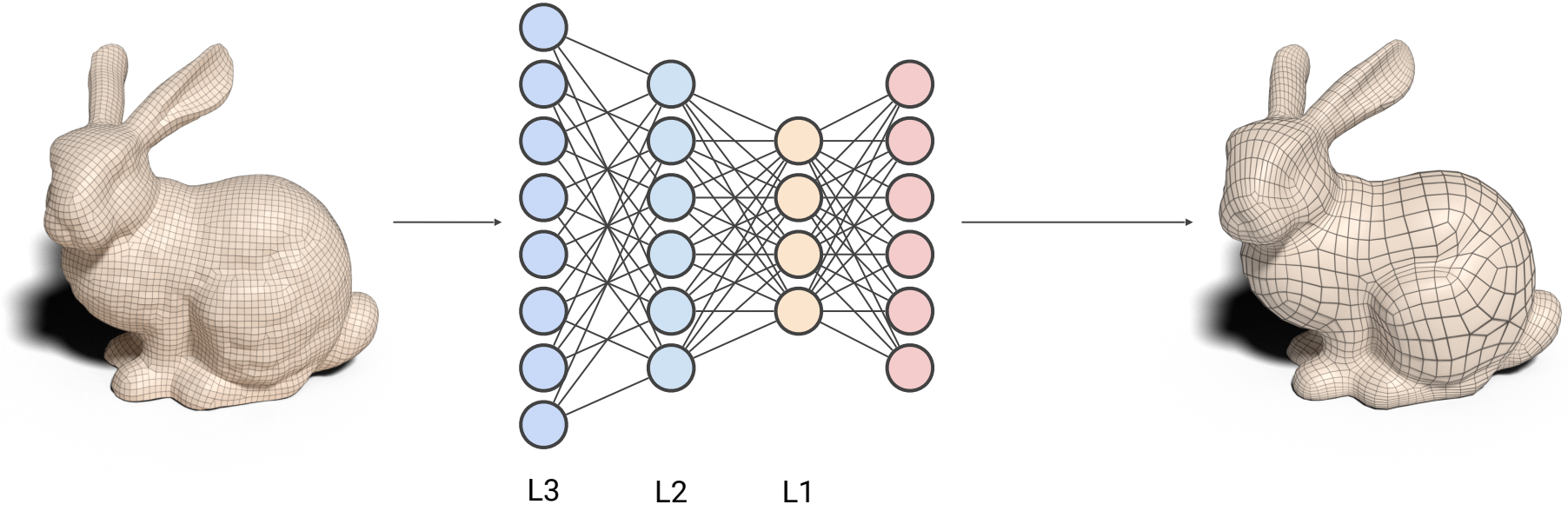
Tri2simNet



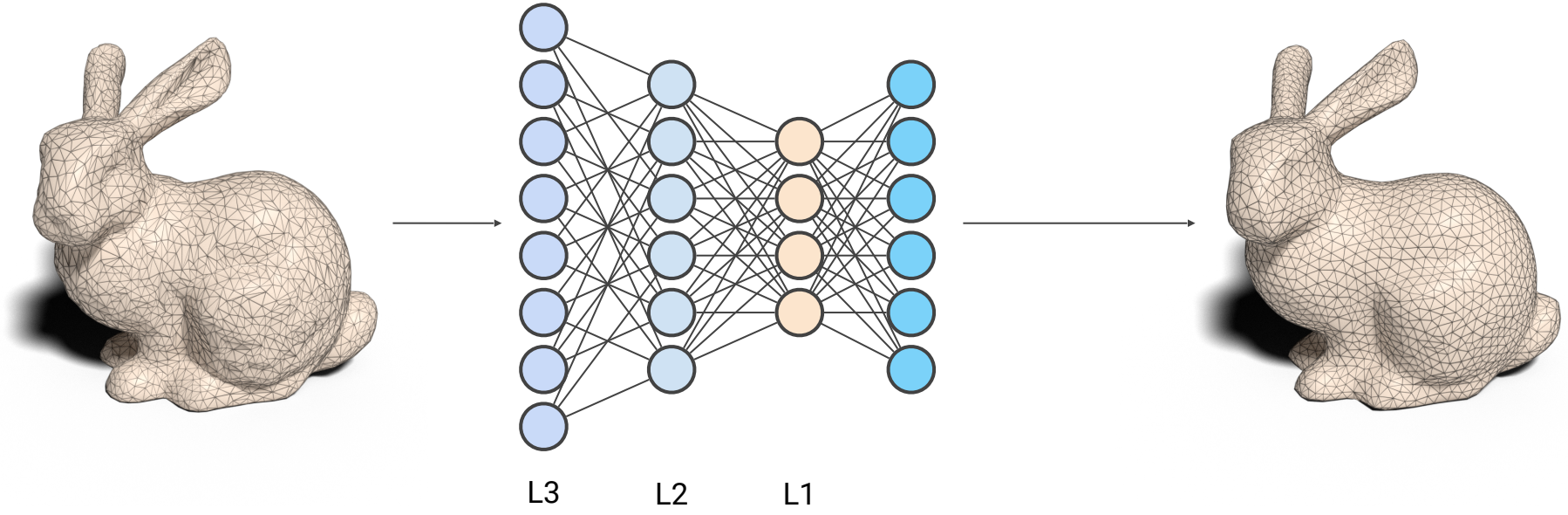
Tri2quadNet



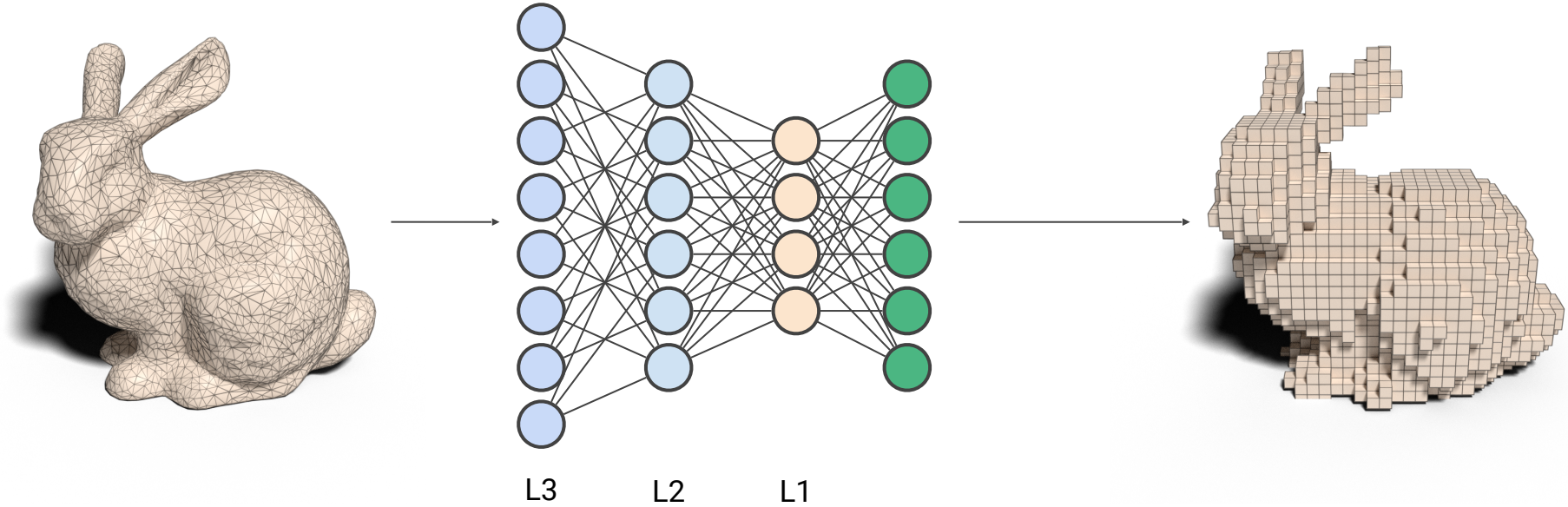
Quad2simNet



Tri2DelauneyNet



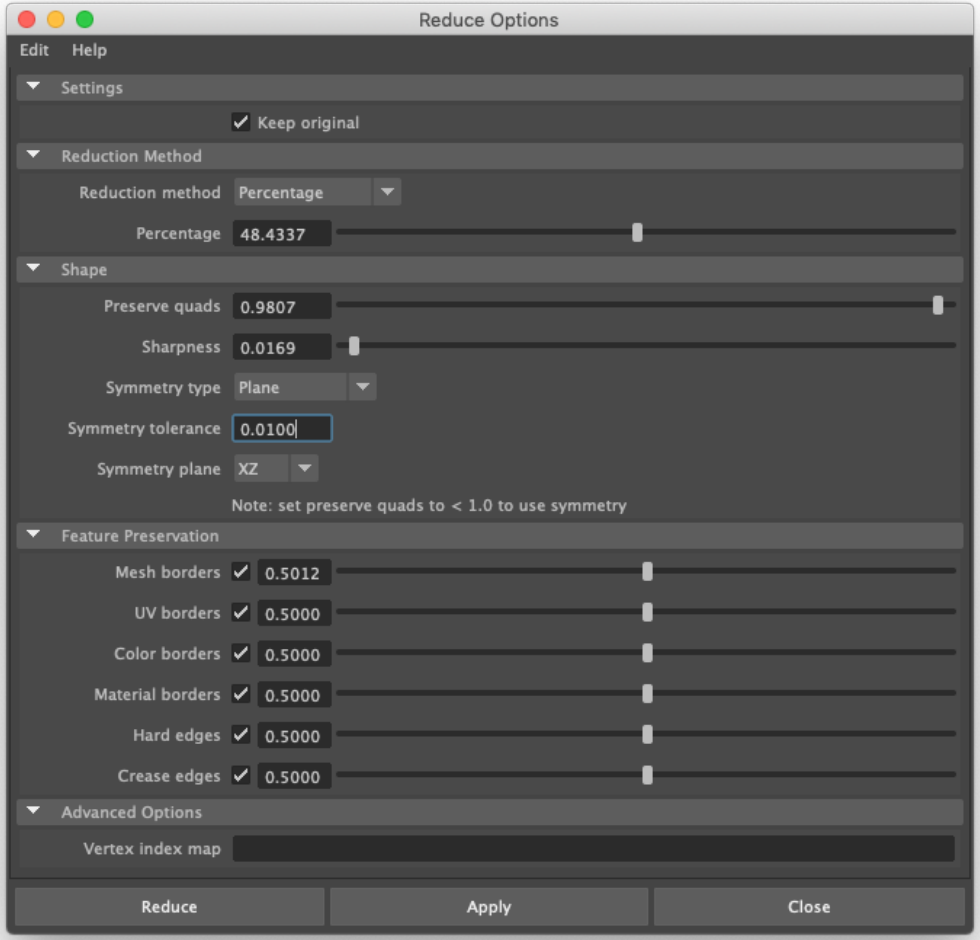
Tri2gridNet



Previously on Polygon Reduction (Polyred)...

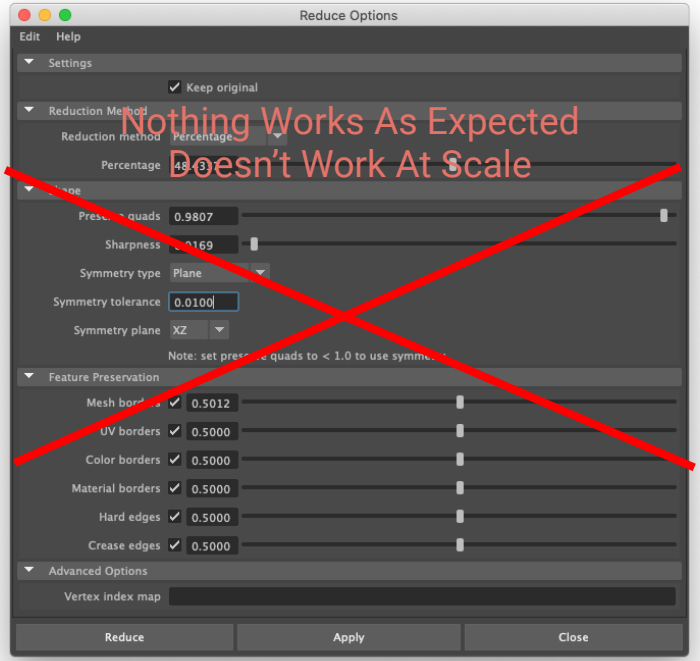
changkun.de/s/polyred2what (2019b)

MAYA


Reduce Options
Edit
Help
Settings
Keep original
Reduction Method
Reduction method
Percentage
Percentage
48.4337
Shape
Preserve quads
0.9807
Sharpness
0.0169
Symmetry type
Plane
Symmetry tolerance
0.0100
Symmetry plane
XZ
Note: set preserve quads to < 1.0 to use symmetry
Feature Preservation
Mesh borders
0.5012
UV borders
0.5000
Color borders
0.5000
Material borders
0.5000
Hard edges
0.5000
Crease edges
0.5000
Advanced Options
Vertex index map
Reduce
Apply
Close

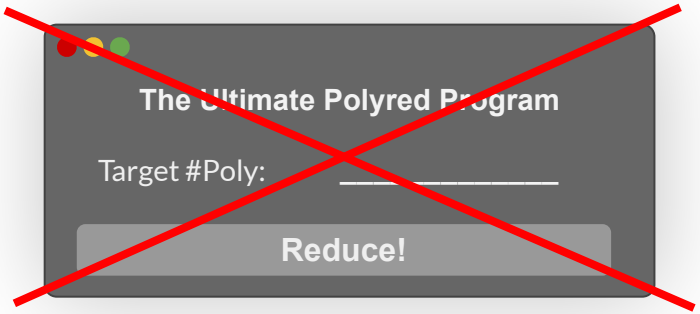
No Free Lunch: Simplicity is complicated

$$\sum_{\text{models}} \text{cost}(\text{models}, \text{methods}, \text{target}) = \text{constant}$$

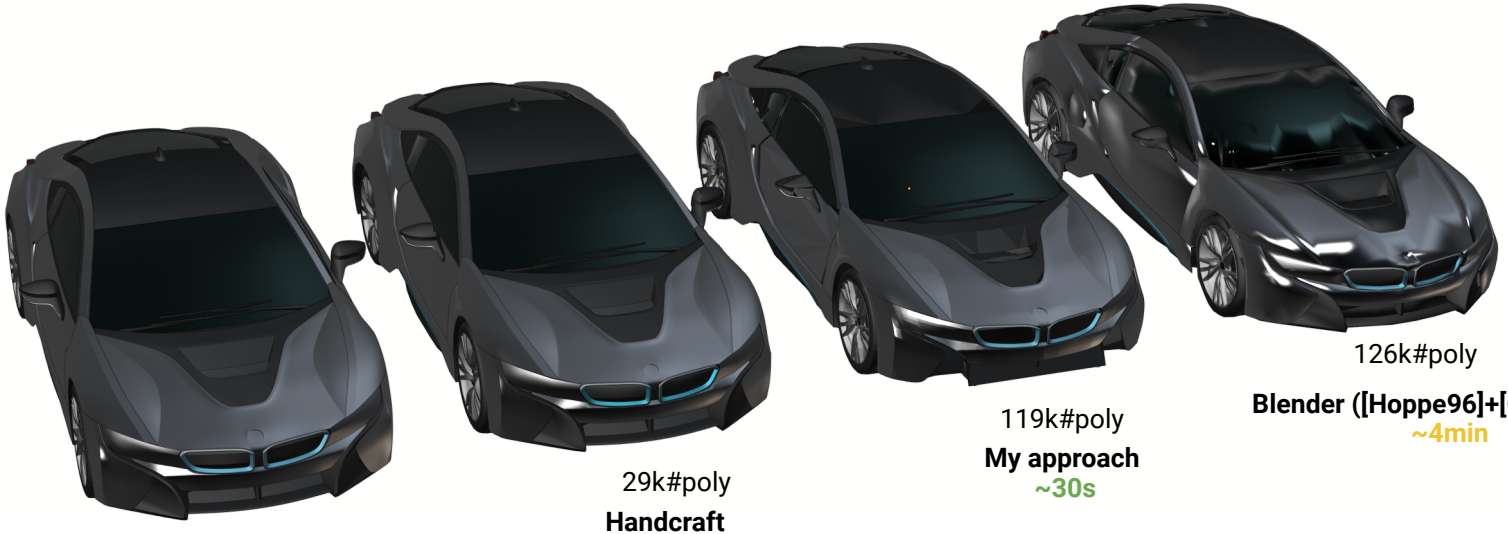


VS

Impractical



"Polyred is solved!!"



640k#poly

Ground truth

29k#poly

Handcraft

40 hours+

119k#poly

My approach

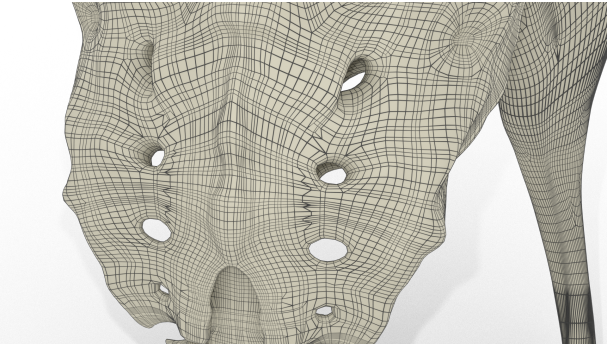
~30s

126k#poly

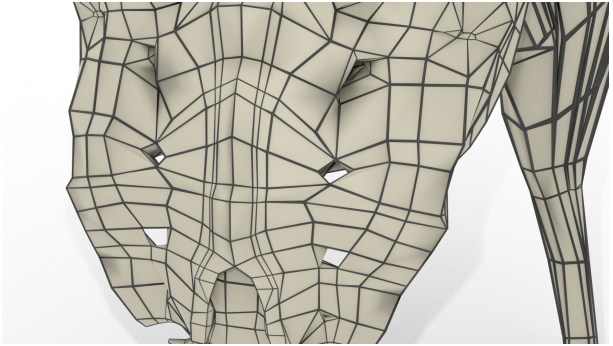
Blender ([Hoppe96]+[Garland97])

~4min

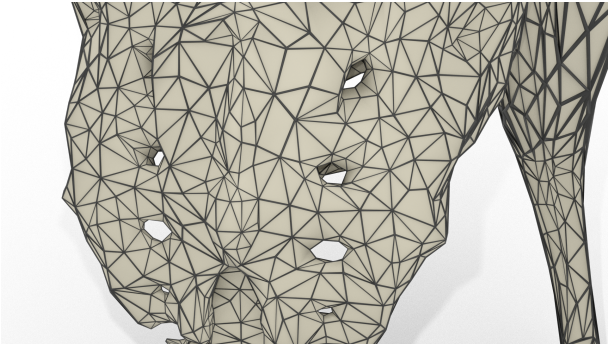
"Polyred is solved!!”



Ground truth

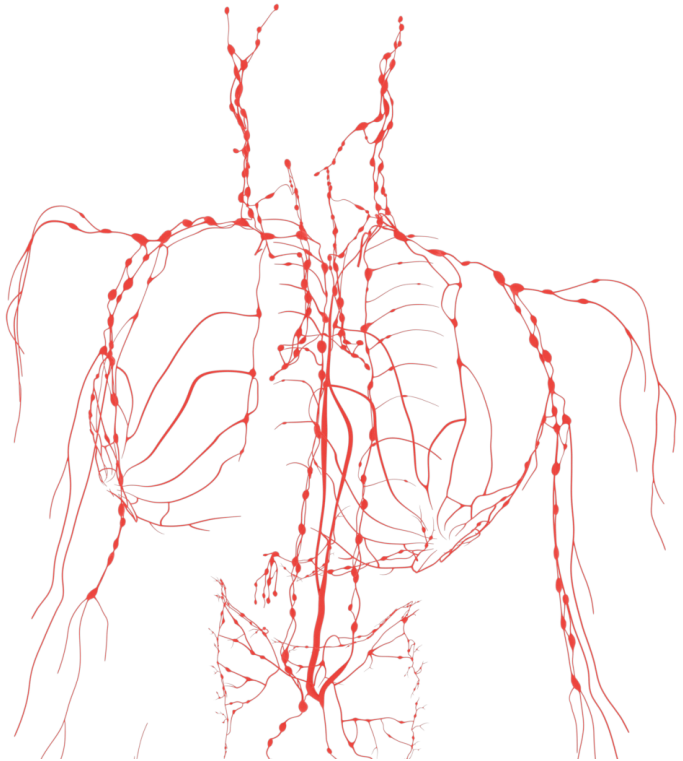


My approach



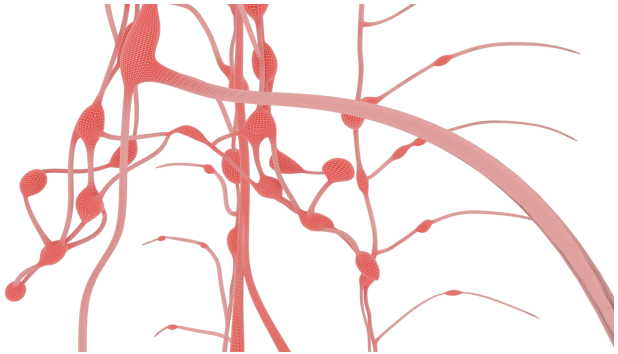
Blender

"Polyred is solved at scale!!"

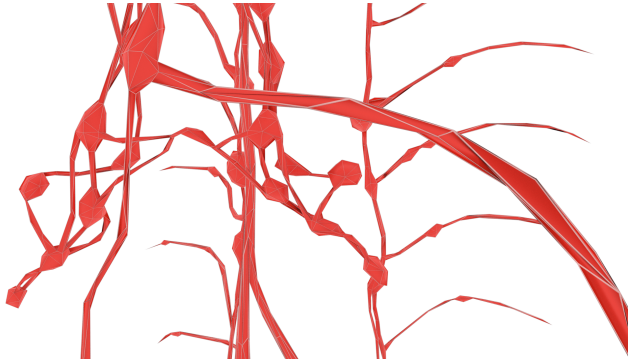


Blood Circulation System

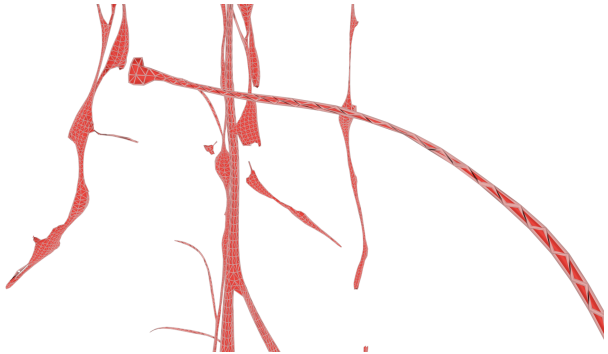
"Polyred is solved at scale!!"



Ground truth

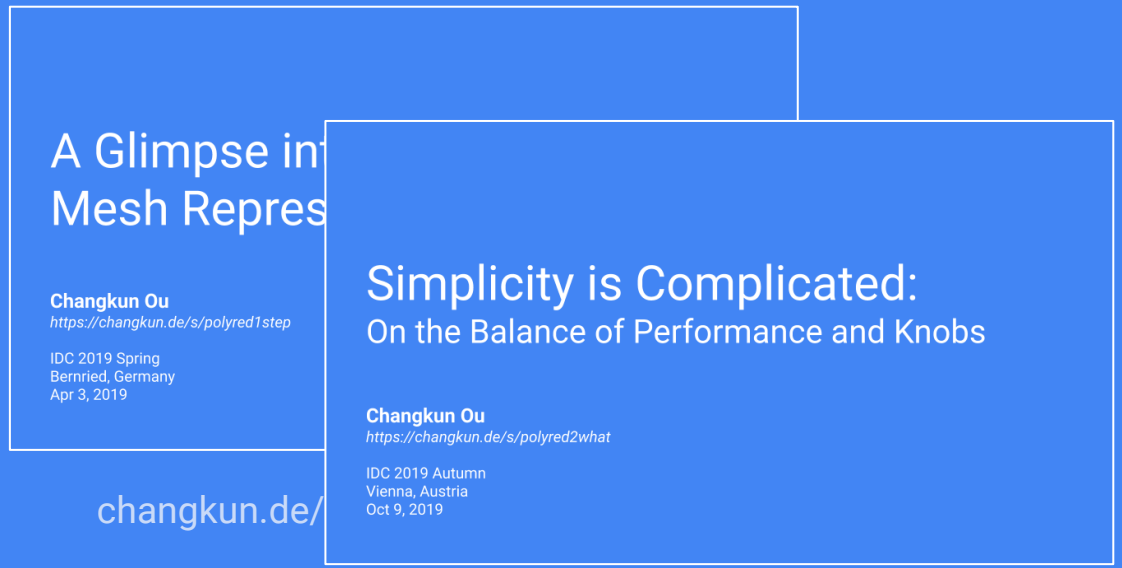


My approach



Blender

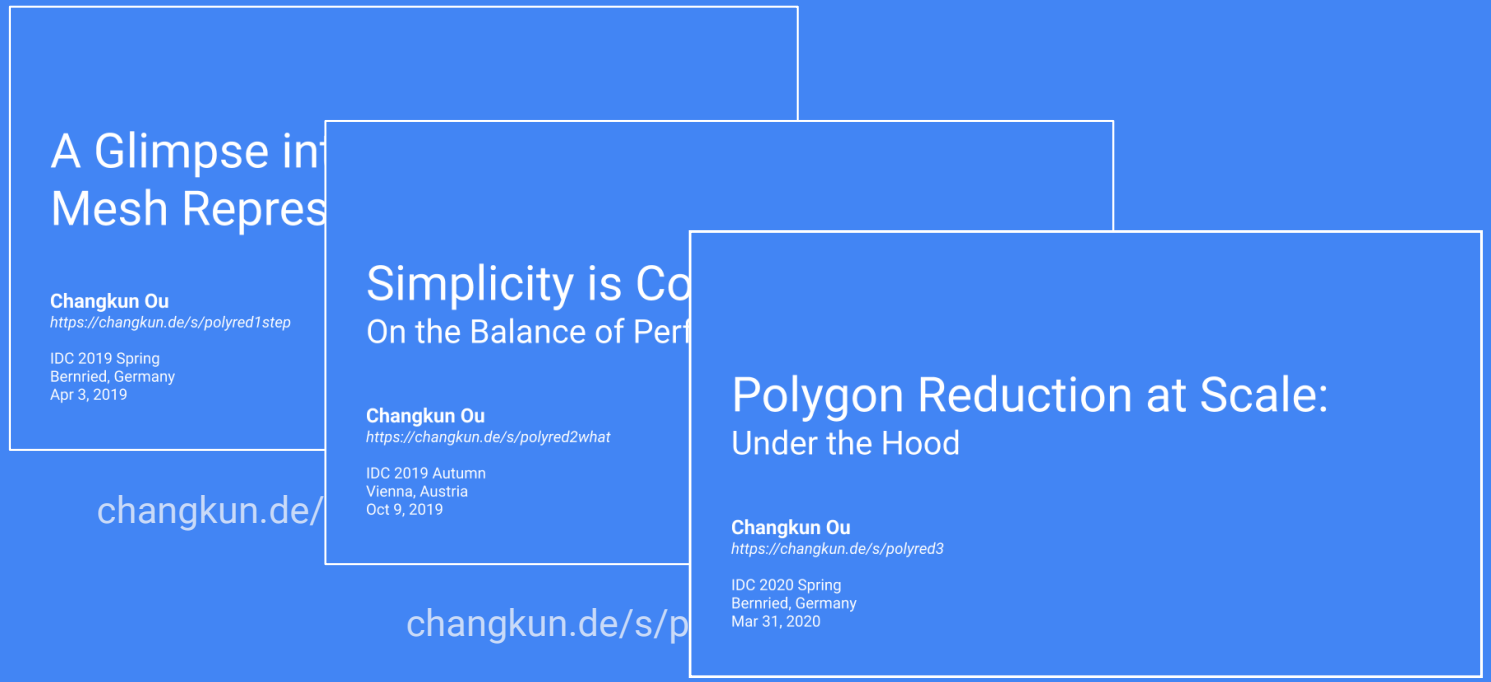
Previously on Polygon Reduction (Polyred)...



changkun.de/

changkun.de/s/polyred2what (2019b)

Previously on Polygon Reduction (Polyred)...



changkun.de/s/polyred3 (2020a)

Worker Multiplexing 🔥🔥🔥

Naïve Polyred

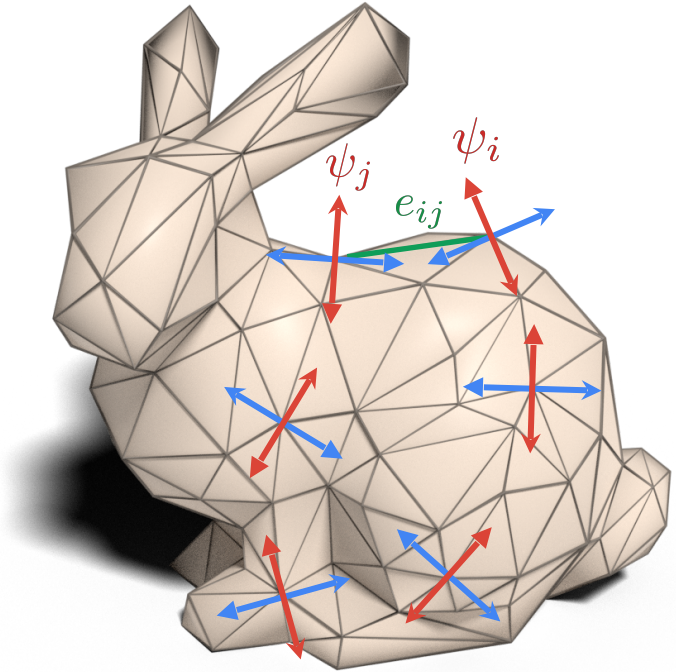
```
// Polyred reduces number of polygons
// while preserving local topologies.
func Polyred(m *Mesh, c *Criteria) {

      for !m.Eval(c) {
            local := m.Pick()
            local.Simplify()
      }

}
```

Inspiration

```
// SGD implements mini-batch
// stochastic gradient descent.
func SGD(m *Model, d *Dataset) {

      for !m.Converge() {
            miniB:= d.Batch()
            m.GradientDescent(miniB)
      }

}
```

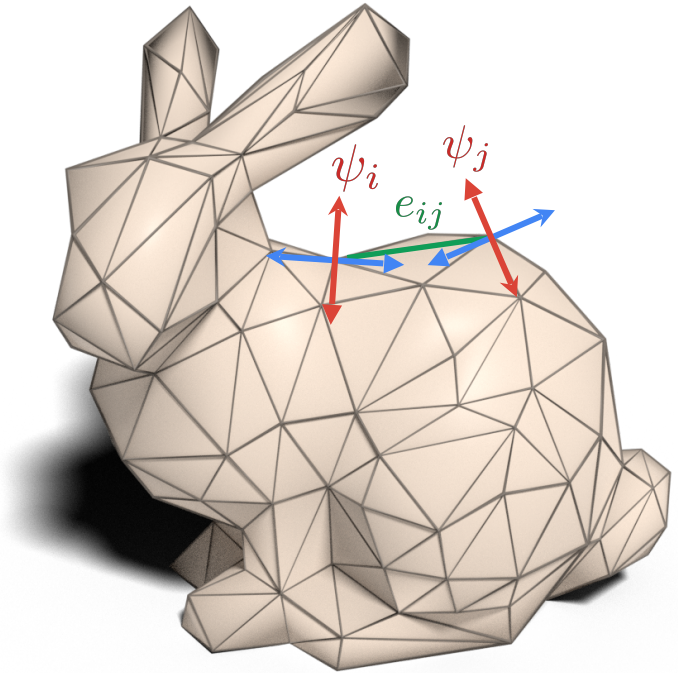
```
func Polyred(m *Mesh, c *Criteria) {
      // SPEEDUP1: build workQ concurrently
      for local := m.Pick(); local != nil; {
            sched.Submit(func() {
                  quality, ok := local.Eval(c)
                  if ok {
                        workQ.Push(quality, local)
                  }
            })
            local = m.Pick()
      }
      sched.Wait() // sync barrier

      // SPEEDUP2: run workQ concurrently
      for w := workQ.Pop(); w != nil; {
            sched.Submit(w.Simplify)
            w = workQ.Pop()
      }
      sched.Wait() // sync barrier
}
var sched Sched // M:N work-steal scheduling
func (s *Sched) Submit(f func()) { … }
```

Under the Hood



Under the Hood: Mesh Energy Minimization

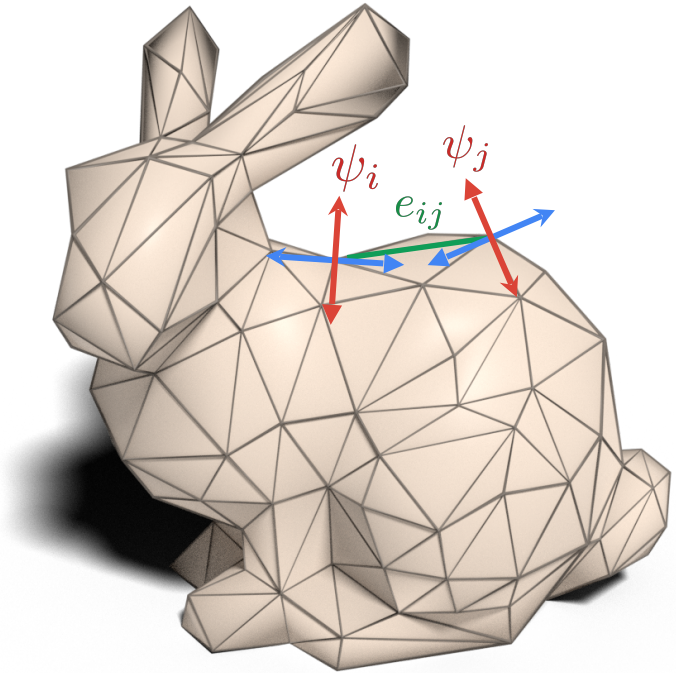
$$\min_{\psi} \sum_{e_{ij} \in \mathcal{S}} |\psi_j - \psi_i|^2$$



Under the Hood: Mesh Energy Minimization

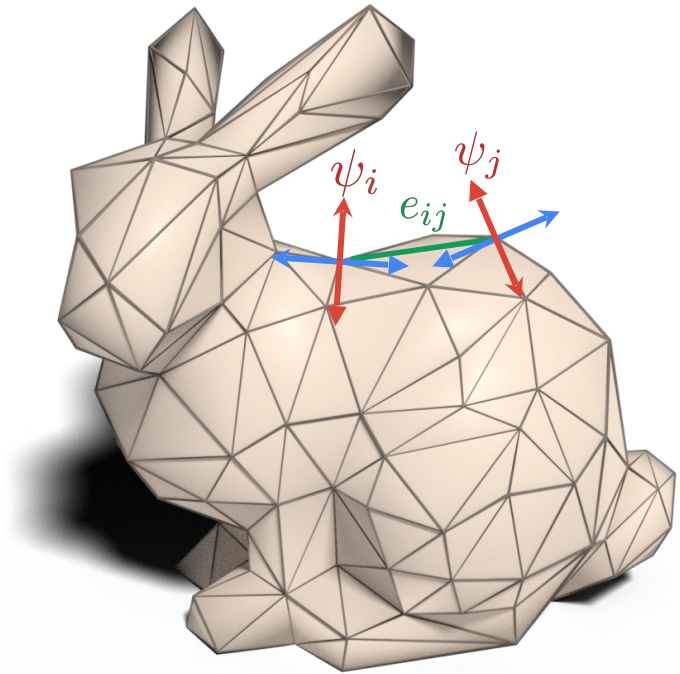
$$\min_{\psi} \sum_{e_{ij} \in \mathcal{S}} |\psi_j - \psi_i|^2$$

$$\min_{\psi} \int |\nabla \psi|^2$$ **(Dirichlet Energy)**



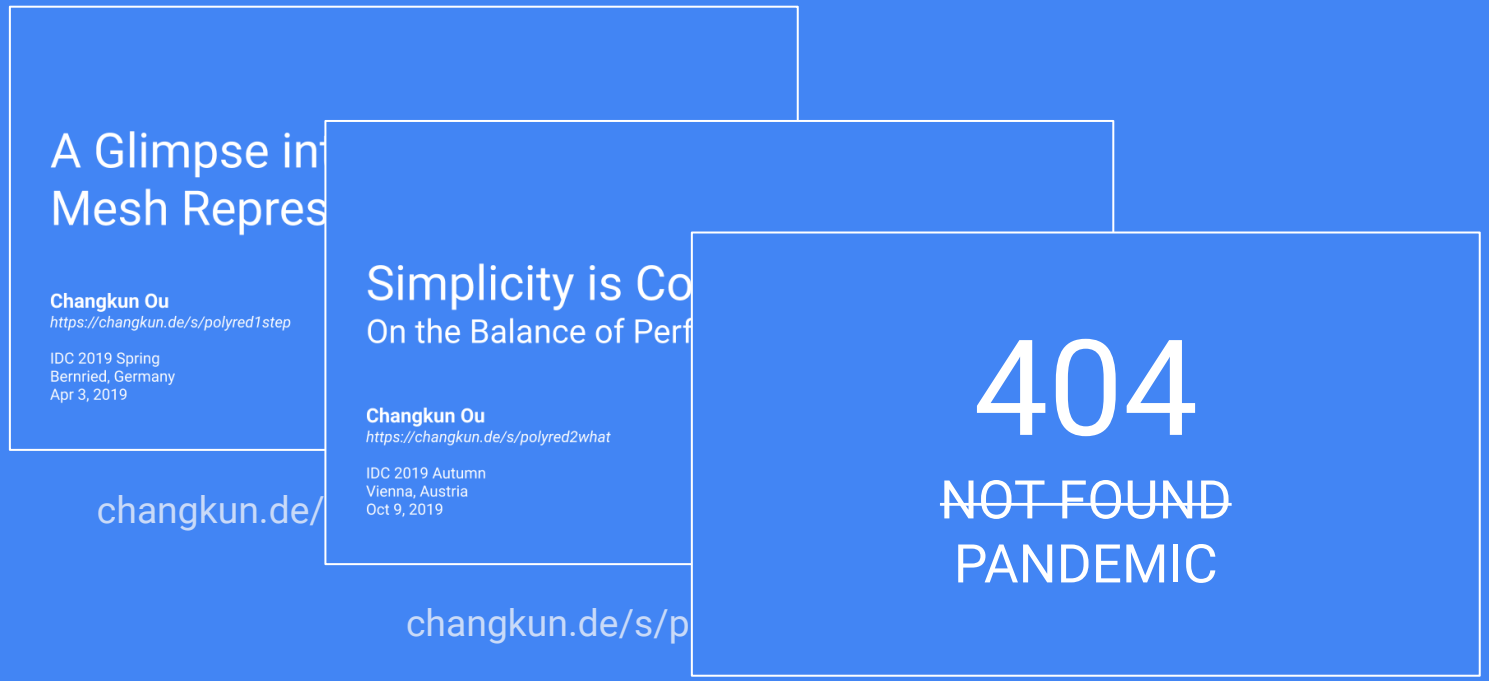
Under the Hood: Mesh Energy Minimization

$$\Delta\left(\quad\right) = 0$$



Under the Hood: Mesh Energy Minimization

Previously on Polygon Reduction (Polyred)...



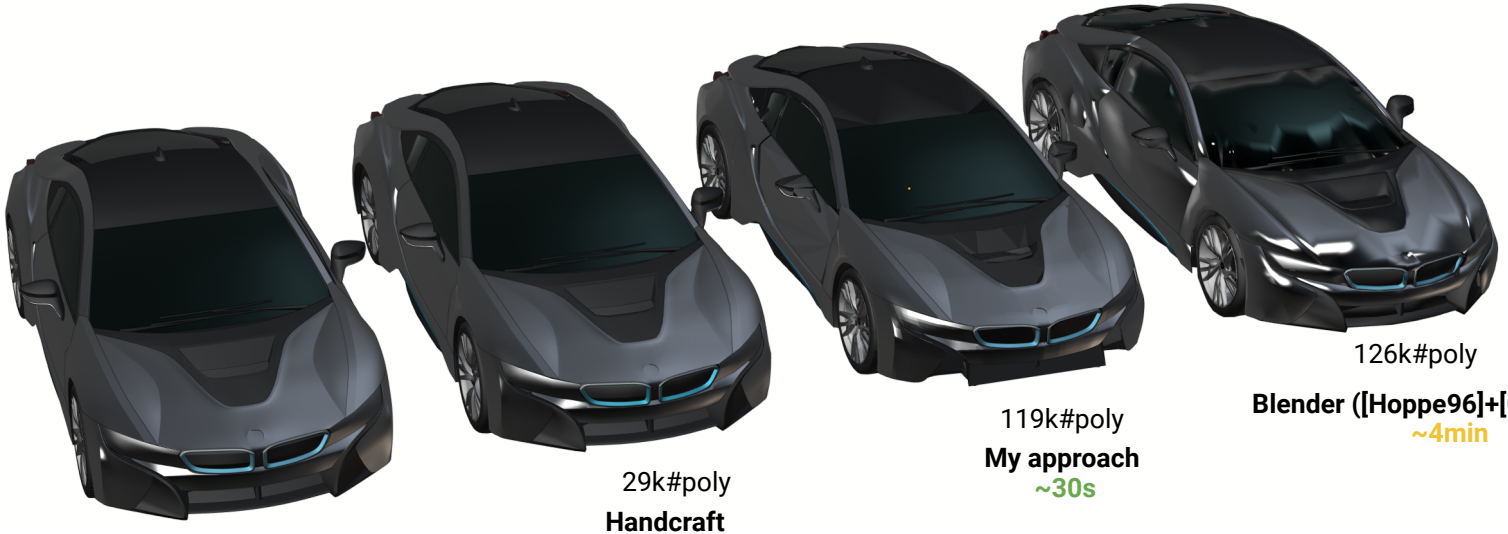
~~changkun.de/s/polyred3 (2020a)~~

A Future of Polygon Reduction

Changkun Ou
changkun.de/s/polyred4us
@changkun

IDC 2020 Autumn
Venice, Italy
Oct 7, 2020

"Polyred is solved at scale!!"



640k#poly

Ground truth

29k#poly

Handcraft

40 hours+

119k#poly

My approach

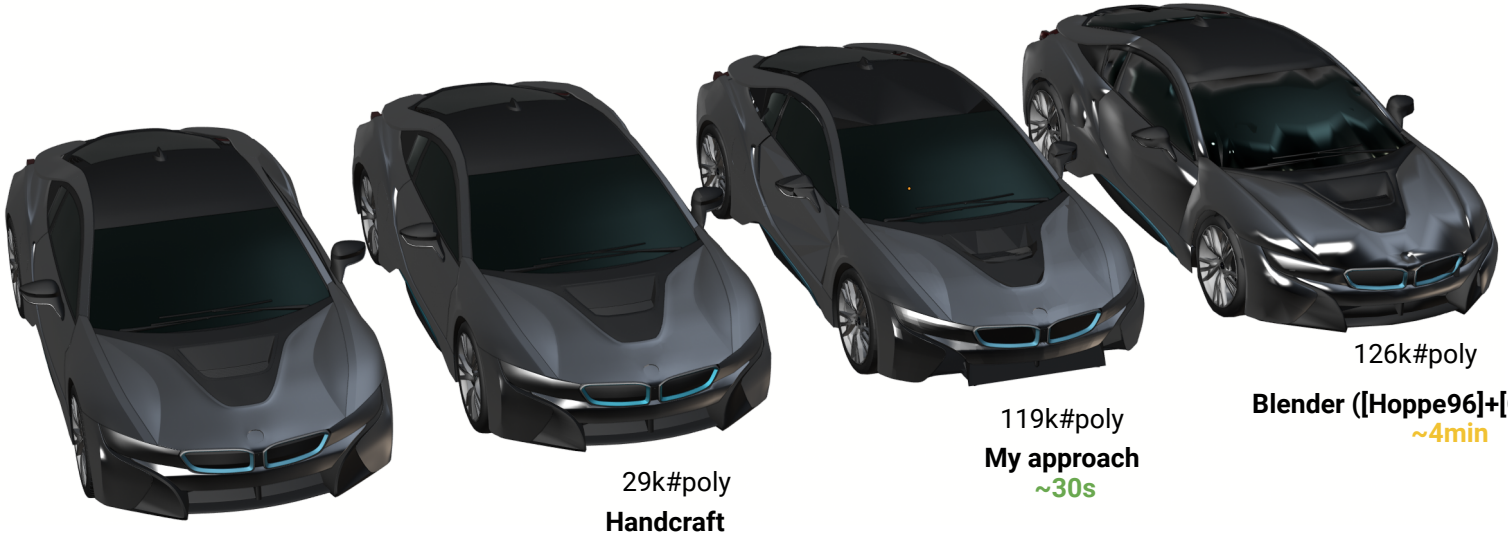
~30s

126k#poly

Blender ([Hoppe96]+[Garland97])

~4min

"Polyred is solved at scale!!"



640k#poly

Ground truth

29k#poly

Handcraft

40 hours+

119k#poly

My approach

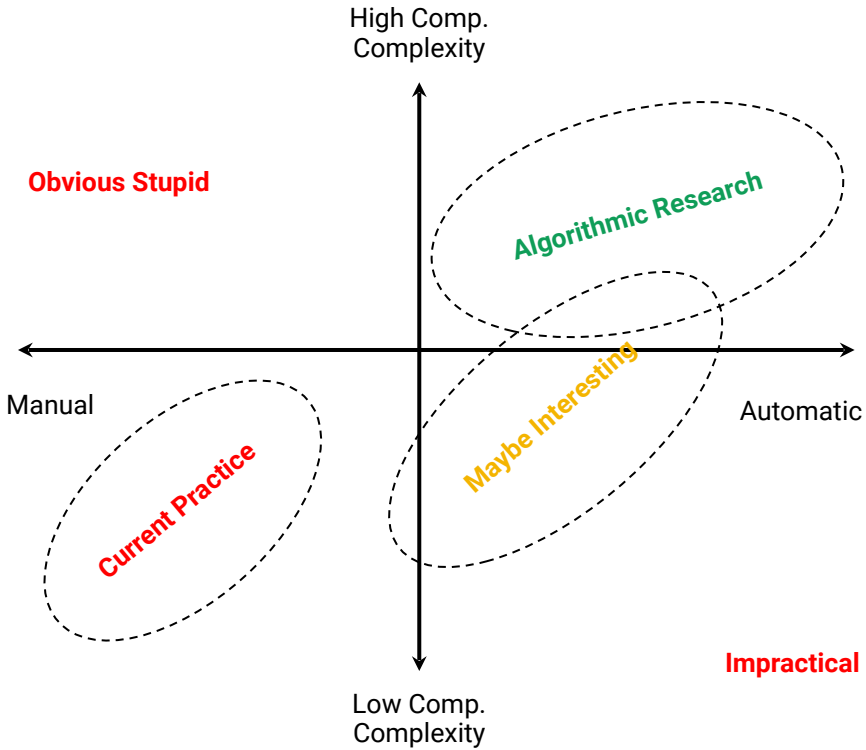
~30s

126k#poly

Blender ([Hoppe96]+[Garland97])

~4min

Automation, Speed, Quality Tradeoff

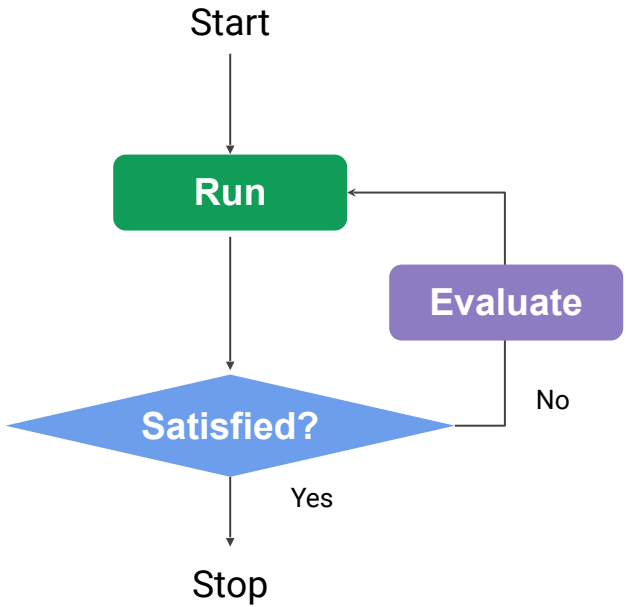


A First Thought on Human Cost

Time Spent | # Attempts | Decision Time | Polyred Time

$$T = \sum_{a \in A} (DT(\mathbf{x}) + P(\mathbf{m}))$$

Iteration Complexity | Model Complexity | UI Complexity




```
interface Polyred {
    Upload(m *Model) (sessID string)
    Run(sessID string, n int) (opIDs []OpID) // 1m #poly ≈ n minutes → n models
    Eval(opIDs []OpID, scores []int)
    Download(OpID string) (m *Model)
}
```

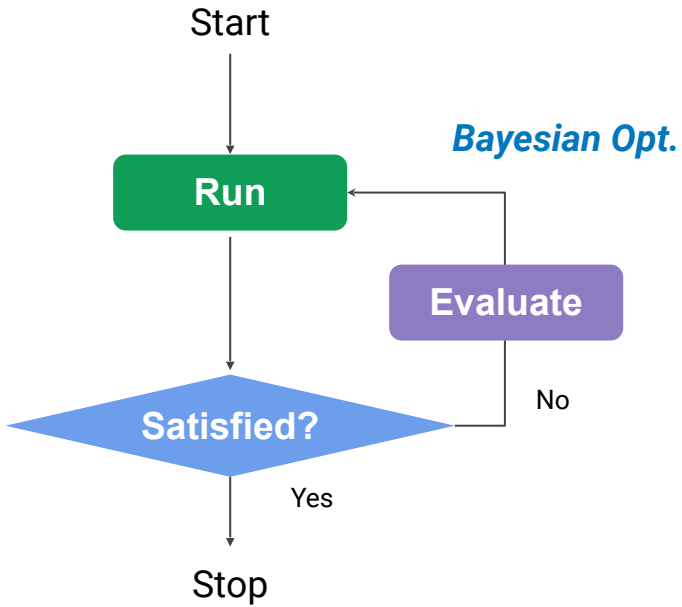
*Talk to me for web APIs access at **https://poly.red***

A First Thought on Human Cost

Time Spent | # Attempts | Decision Time | Polyred Time

$$T = \sum_{a \in A} (DT(\mathbf{x}) + P(\mathbf{m}))$$

Iteration Complexity | Model Complexity | UI Complexity




```
interface Polyred {
    Upload(m *Model) (sessID string)
    Run(sessID string, n int) (opIDs []OpID) // 1m #poly ≈ n minutes → n models
    Eval(opIDs []OpID, scores []int)
    Download(OpID string) (m *Model)
}
```

*Talk to me for web APIs access at **https://poly.red***

Yet Another Ultimate Polyred Program?

Rate from 1~5 for each model
(1=worst, 5=best)

Run

Yet Another Ultimate Polyred Program?

(1/5) Rate from 1~5 for each model
(1=worst, 5=best)

-99.5%

-90.0%

-50.0%

-20.0%

Evaluate

Yet Another Ultimate Polyred Program?

(1/5) Rate from 1~5 for each model
(1=worst, 5=best)

-99.5%

1

-90.0%

3

-50.0%

4

-20.0%

2

Evaluate

Yet Another Ultimate Polyred Program?

(2/5) Rate from 1~5 for each model
(1=worst, 5=best)

-99.0%

-95.0%

-80.0%

-60.0%

Evaluate

Yet Another Ultimate Polyred Program?

(5/5) Optimal Reduced Model:

-89.4%

Done

How Good Is the Design? Will it Success at Scale?

Looks like a similar and successful idea proven in XYZ, will it also success (at scale) for polyred?

How Good Is the Design? Will it Success at Scale?

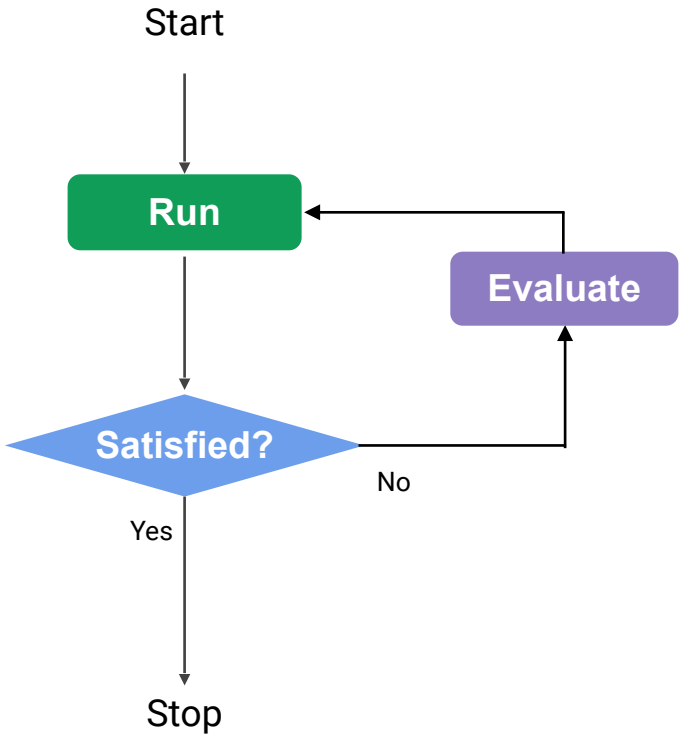
- Initially it indeed draws a lot of attention from the industrial partner
- Later proven this is a really just a look-fancy but horrible design
- Reasons:
-

Human Cost: What did I miss??

Time Spent ***#Attempts*** ***Decision*** **Polyred**

$$T = \sum_{a \in A} (DT(\mathbf{x}) + P(\mathbf{m}))$$

Preference Elicitation ***UI Complexity*** ***Model Complexity***

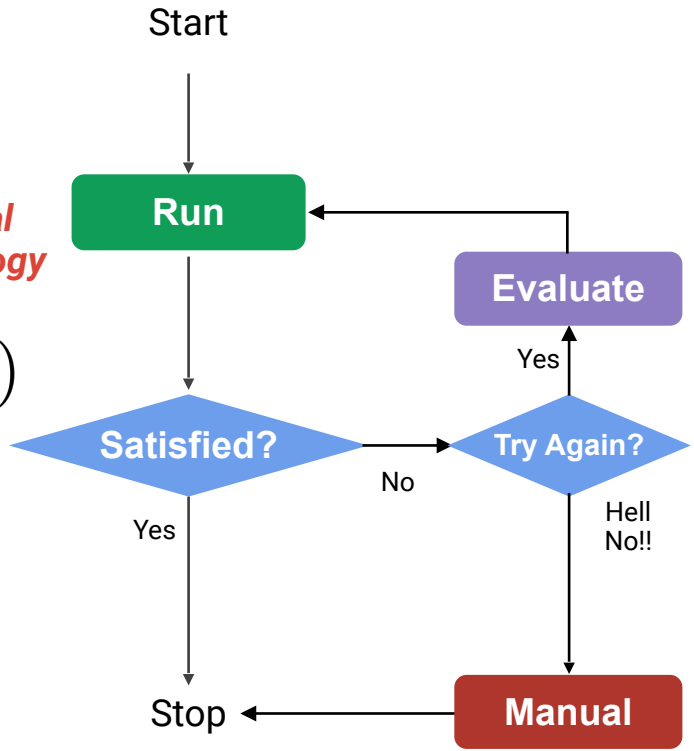


Rethinking Human Cost In Reality

Time Spent ***#Attempts*** ***Decision*** **Polyred** ***Abandon Probability*** ***Manual Retopology***

$$T = p\sum_{a \in A}(DT(\mathbf{x}) + P(\mathbf{m})) + (1-p)H(\mathbf{m})$$

Iteration Complexity ***UI Complexity*** ***Model Complexity*** ***Algorithm Boundedness & Robustness***



Crowd Wisdom: Can we “bring insights from XAI research”?

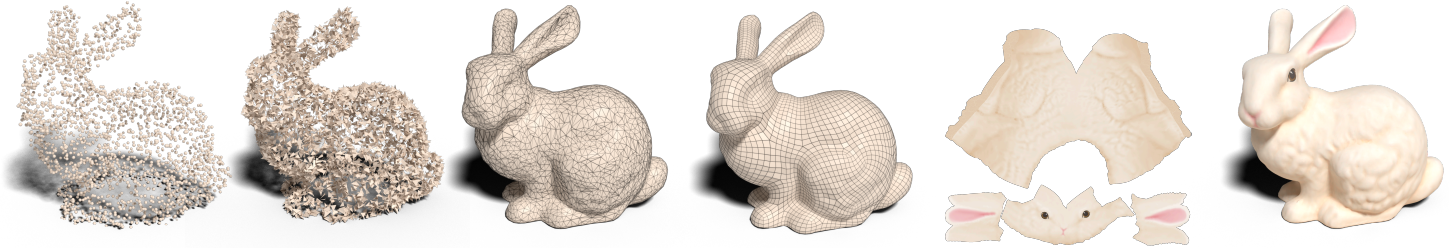
Crowd Wisdom: Can we “bring insights from XAI research”?

“Maybe you just need address the trustworthy issue, that let human believe this is a good result”

Crowd Wisdom: Can we “bring insights from XAI research”?

“Maybe you just need address the trustworthy issue, that let human believe this is a good result”

Revisit the modeling pipeline:

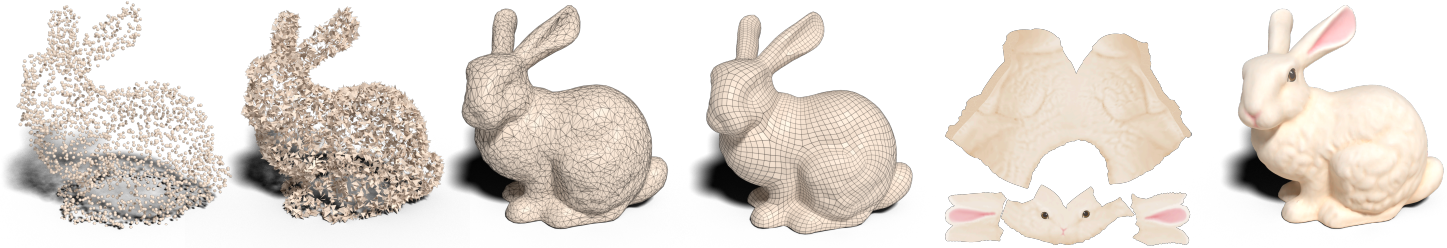


……

Crowd Wisdom: Can we “bring insights from XAI research”?

“Maybe you just need address the trustworthy issue, that let human believe this is a good result”

Revisit the modeling pipeline:



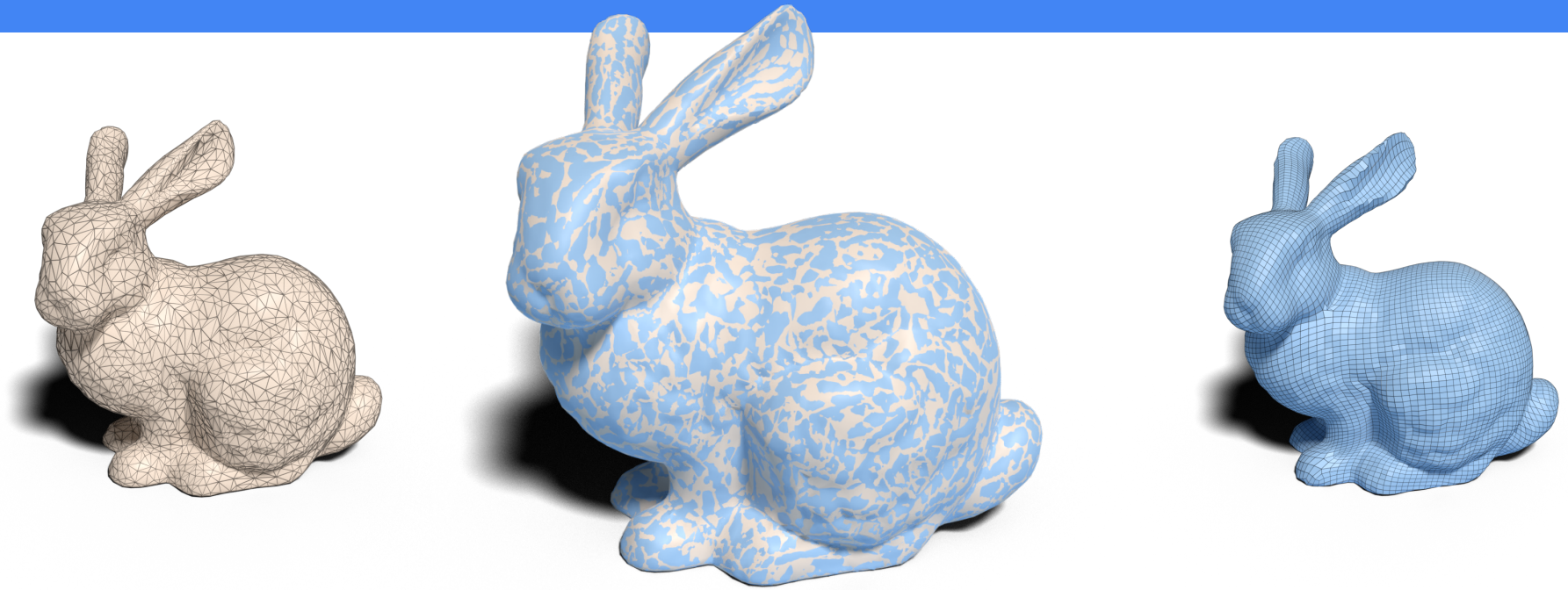
……

- Tons of work after the initial modeling phase
  - Subdivision
  - Surface parameterization
  - Tuning materials
  - Stretching for animation

Revisit: CG101 About ***Good*** Meshing

A good approximation of positions?

Example: Meshes with Equal Hausdorff Distance to A Reference



$$\mathcal{H}(\mathcal{M}, \mathcal{M}') = \sqrt{\frac{1}{\mathcal{S}} \iint_{v \in \mathcal{S}} d(v, \mathcal{S}')^2 \mathrm{d}\mathcal{S}}$$

What We Talk About When We Talk About ***Good*** Meshing

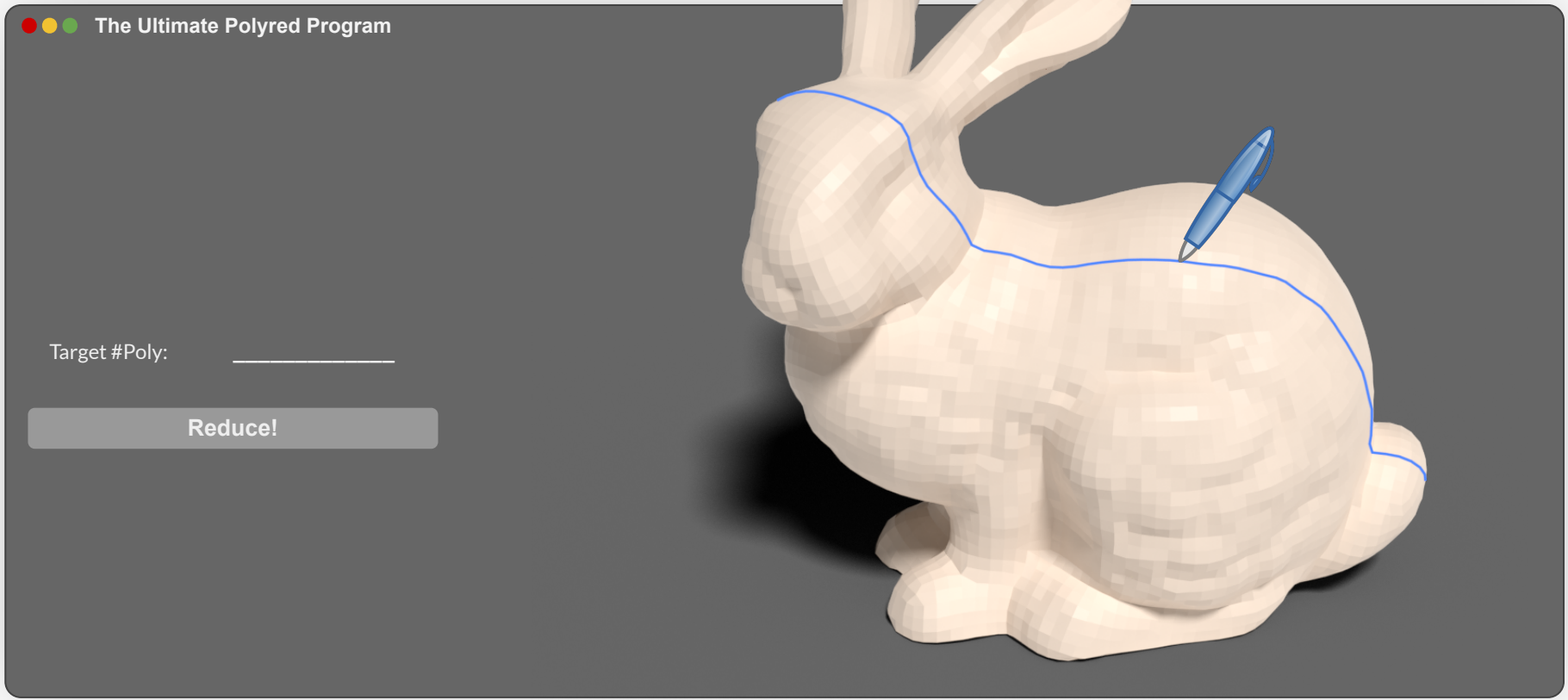
- Approximation of position is not enough
  - Can still have wrong appearance, normals, curvatures, …

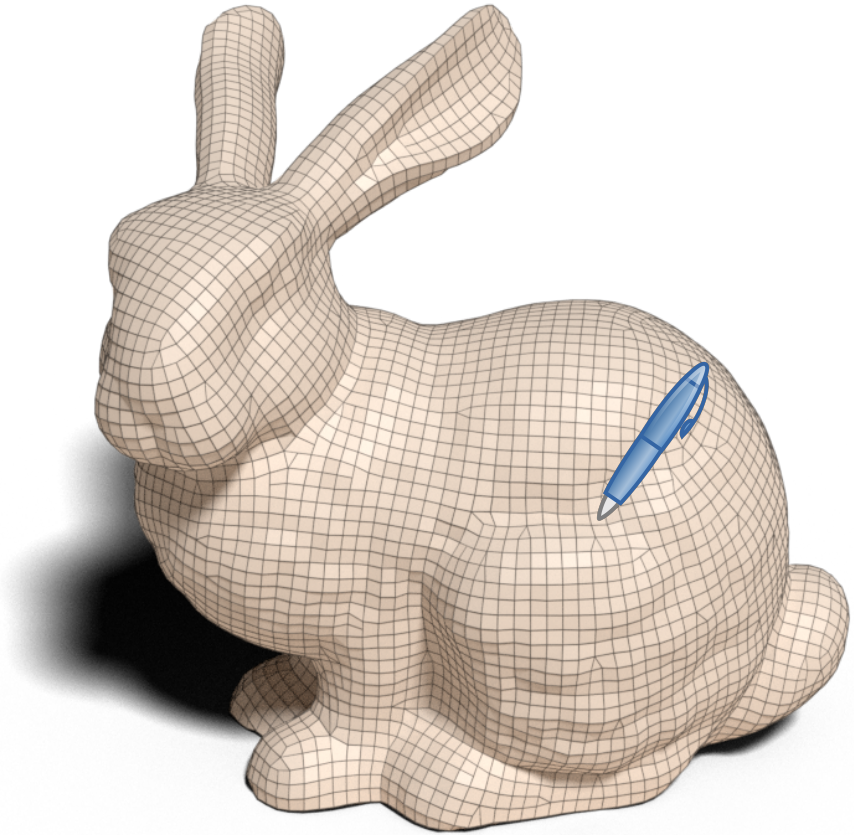
- Rule of thumb today
  - Polygon shape, e.g. Delaunay
  - Vertex degree (Tri6, Quad4)
  - Adaptive sampling
  - Conformality

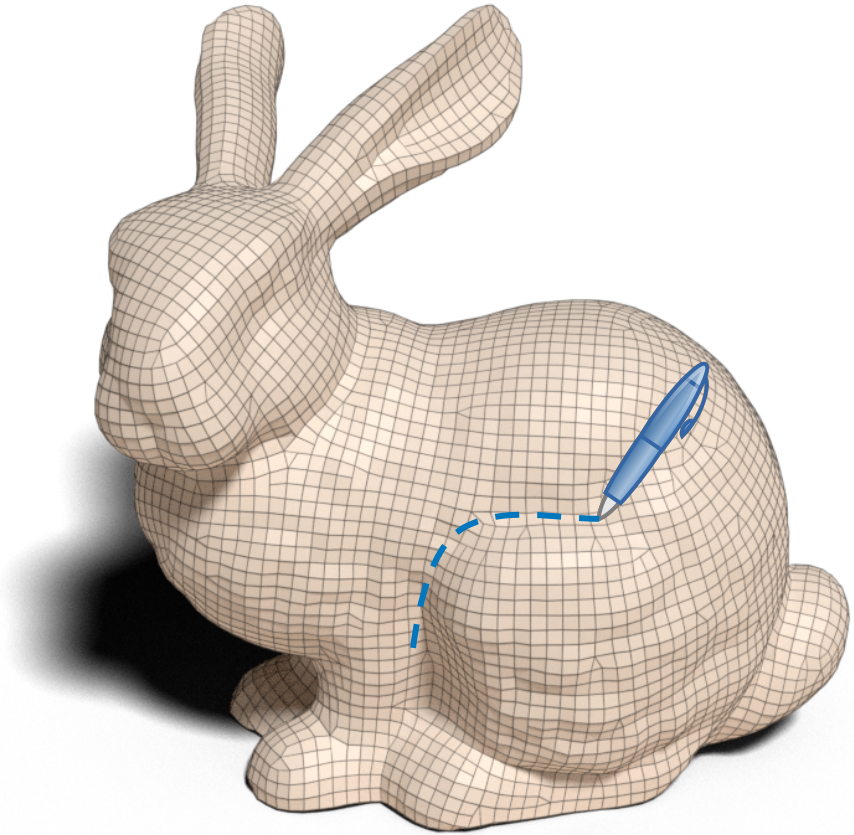
- Sounds trivial but harder to realize
- Even when you talk with domain experts

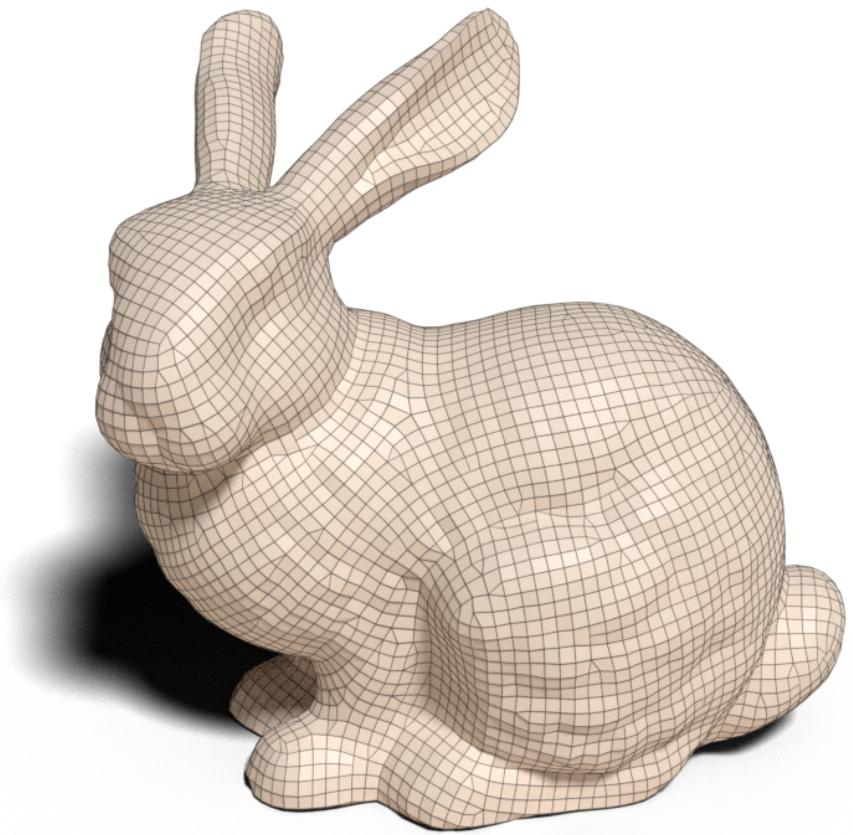
Here should add many good visualizations but I just don't have the time to render it. Sorry.

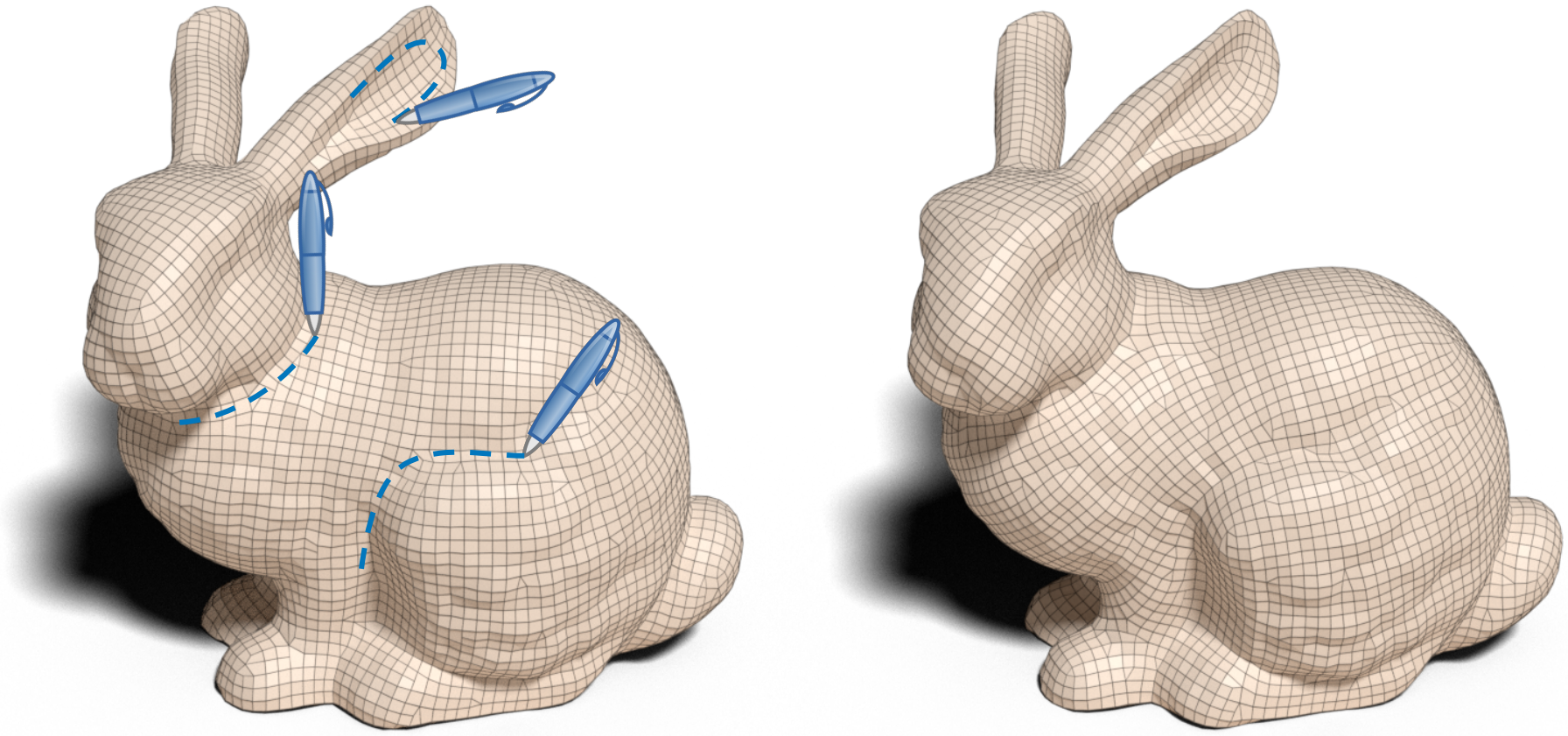
How much control knobs is enough?

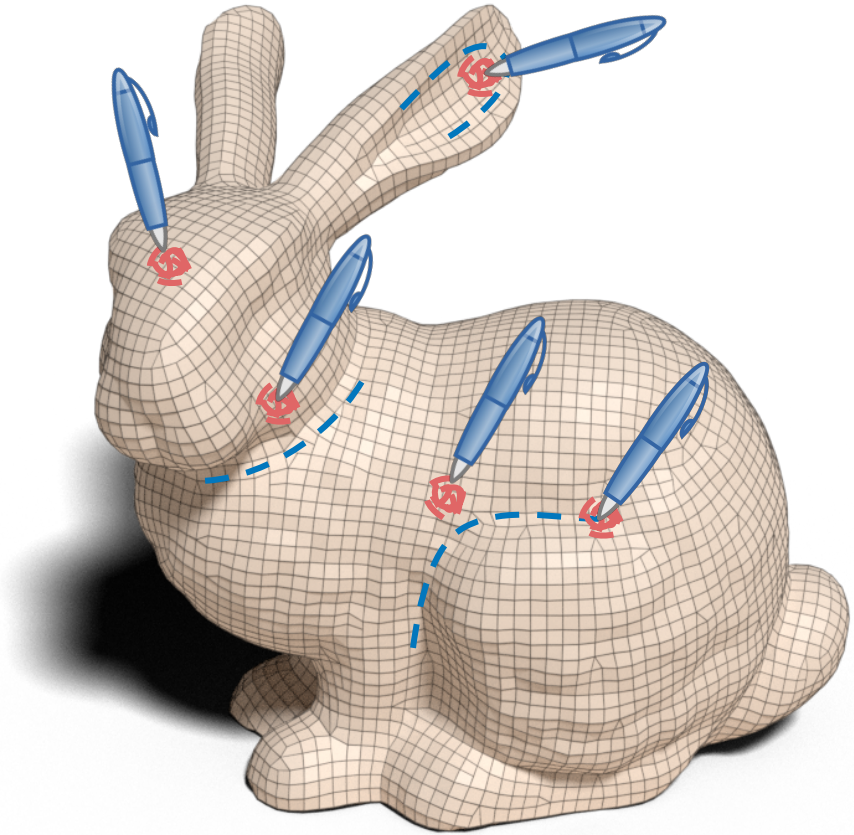


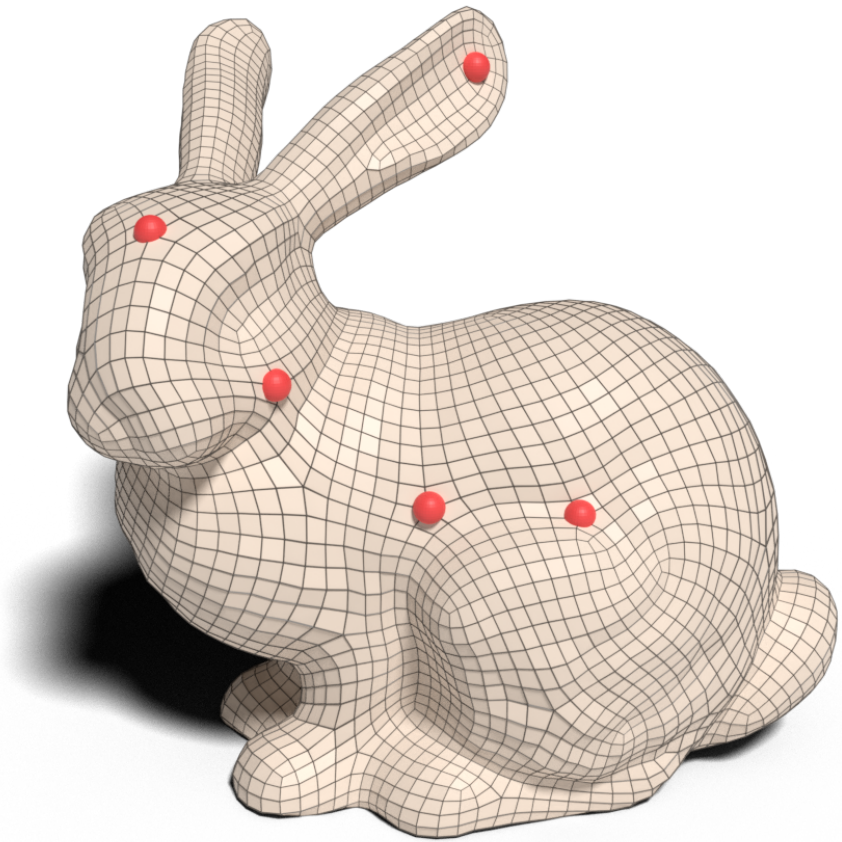






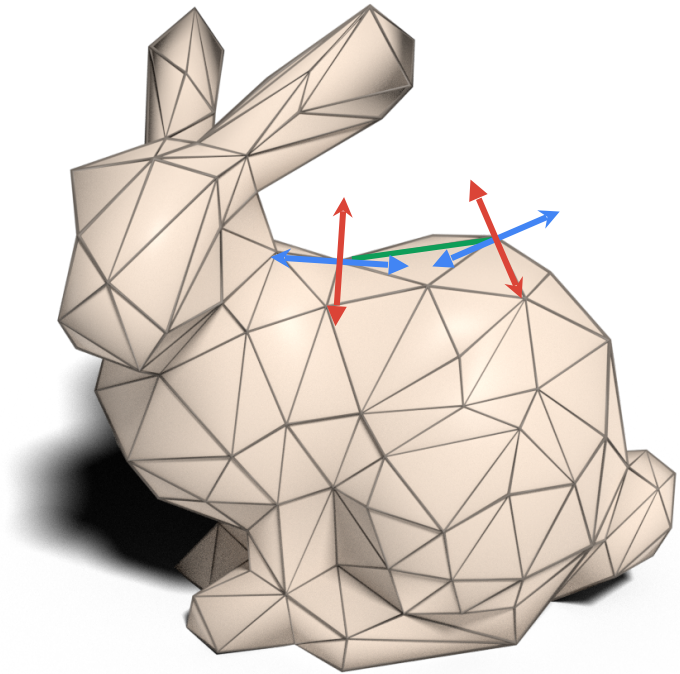




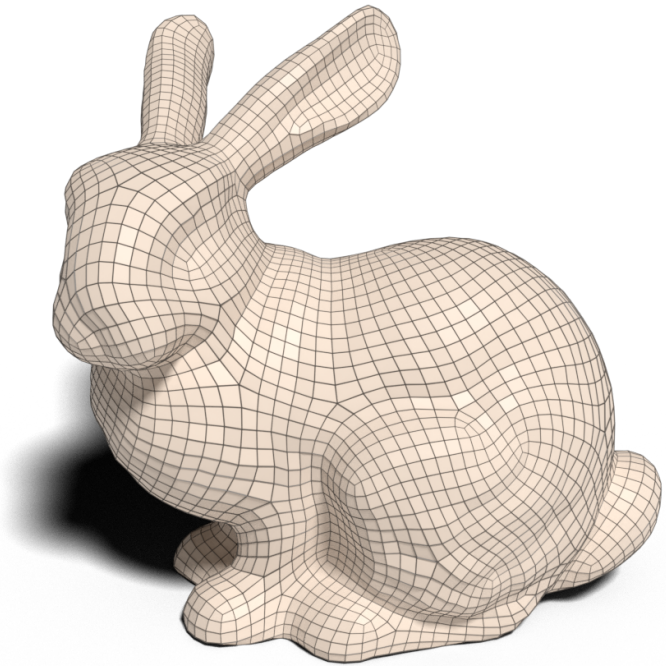


Mesh Energy Minimization (Revised)

$$\Delta\left(\quad\right) = 0$$



Subject to "pen strokes" as boundary condition



A Beautiful Mesh

Do we really want care about meshing?

“If everything looks correct, then it is correct.”

“If everything looks correct, then it is correct.”

-20.0%

-89.4%

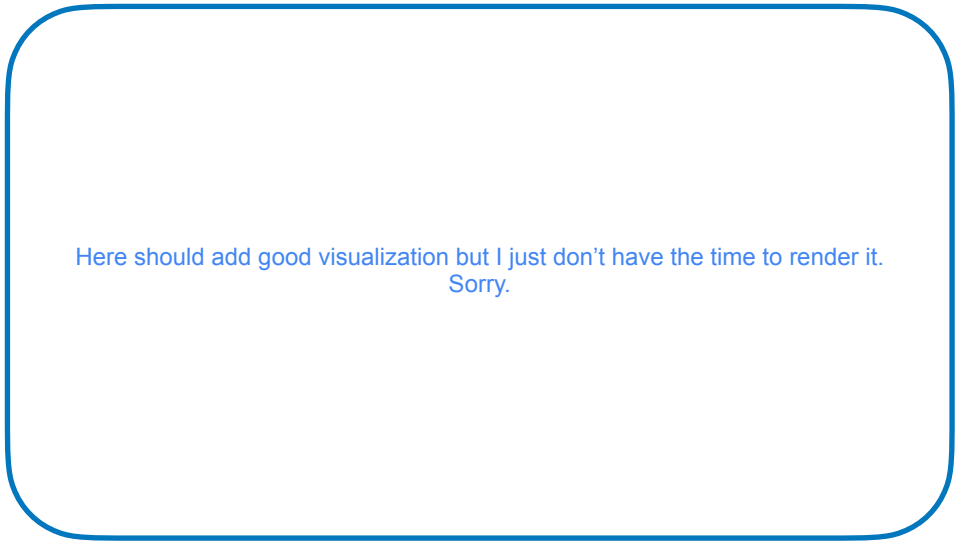
Still Beautiful

Even More Beautiful

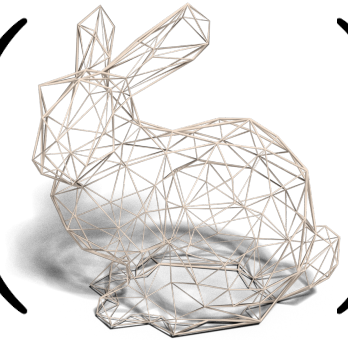
Render Settings

Settings that constitute the final rendered object:

- Rendering algorithm: Rasterization, path tracing, photon mapping, …
- BRDF parameters
- Light sources
- Camera parameters
- Procedural parameters
- …

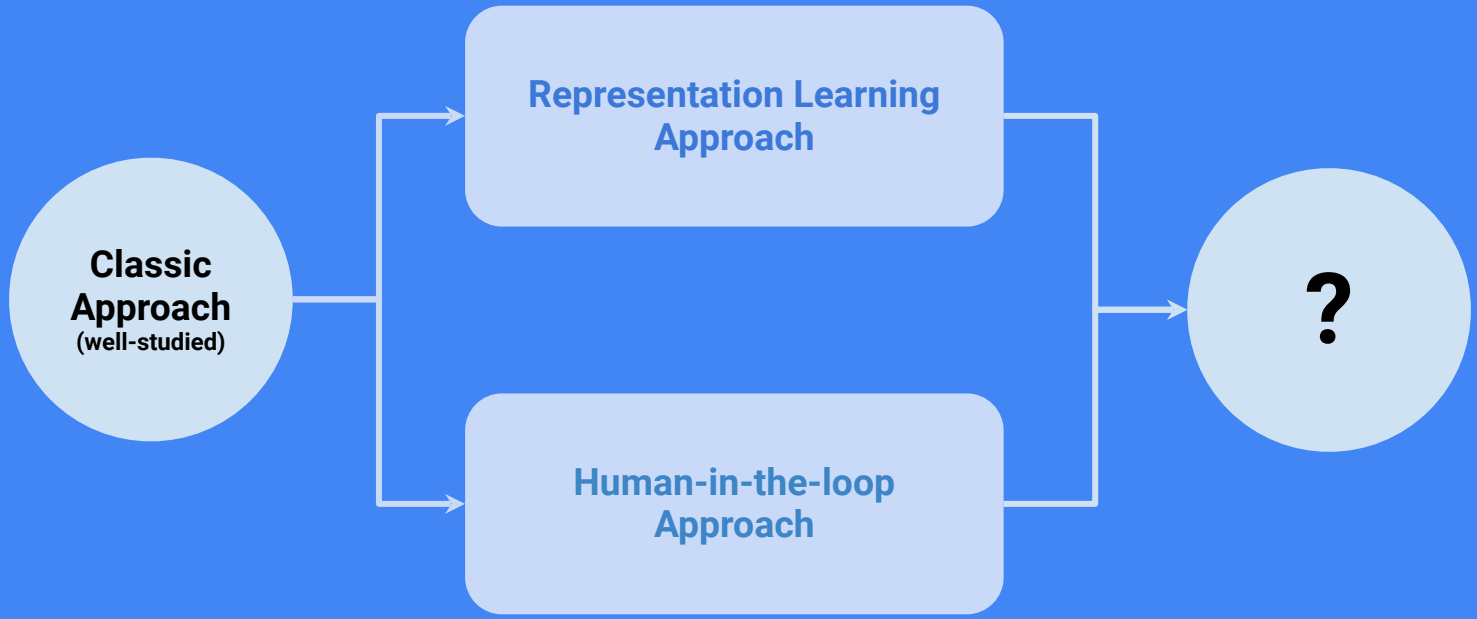


“Differential” Rendering: The Objective

$$L\left(\text{  }\right) = \left\| \text{  } - \text{  } \right\|^2$$

$$\underset{\mathbf{s}\in\mathcal{S}}{\textbf{minimize}}\ L(M, r(\mathbf{s}))$$

Objective *Model* *Renderer* *Render Settings*


Classic Approach
(well-studied)
Representation Learning Approach
Human-in-the-loop Approach
?

Challenges

Image source: Division

Image source: Division

The Cancer in 3D Modeling Pipeline

- **Reusability**
  - More than 90% of the art works are served as one time use purpose
- **User education**
  - Artists are trained strictly, and quite comfortable with their current modeling workflow, whereas
  - General users won't even notice the low quality meshes (analogy as in "color science")
  - **Poly counts is only an issue with rasterization, but not an issue with ray tracing family**
- **…**

Priceless Lessons

3D artists are “educated” and “trained” to produce adaptive sampled parallel flow lines with fewer singularities

Solving edge cases is much harder than the problem itself, and utterly hard to maintain consistency

Fancy algorithms are slow when N is small, and N is usually small

Always rethink the problem whether it is a “bug” that worth a fix

Respect design work and technical details

Robust preference elicitation replaces mesh manipulation

Intermediate representation replaces polygon meshes eventually

Professional end users produce unique understanding the original purpose of a feature

Don’t trust domain experts when you start touching their area, just become one of them

…

What was missed?

What else can be simplified?

What else can be done better?

What works what doesn't?

Report your experience

...

A Future of Polygon Reduction

Changkun Ou
changkun.de/s/polyred4us

IDC 2020 Autumn
Venice, Italy
Oct 7, 2020

What was missed?

What else can be simplified?

What else can be done better?

What works what doesn't?

Report your experience

...

The Death
~~A Future~~ of Polygon Reduction

Changkun Ou
changkun.de/s/polyred4us

IDC 2020 Autumn
Venice, Italy
Oct 7, 2020

What was missed?

What else can be simplified?

What else can be done better?

What works what doesn't?

Report your experience

...

The Death
~~A Future~~ of Polygon Reduction

Changkun Ou
changkun.de/s/polyred4us

IDC 2020 Autumn
Venice, Italy
Oct 7, 2020