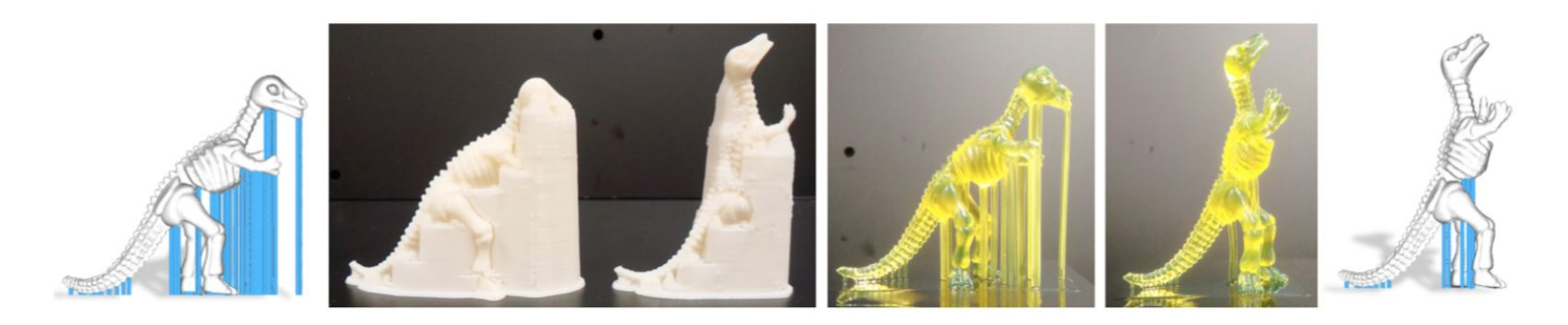
Fast skeleton-based deformation to reduce 3D printing support

Julia Gilenko, Peiqi Wang, Eris Zhang

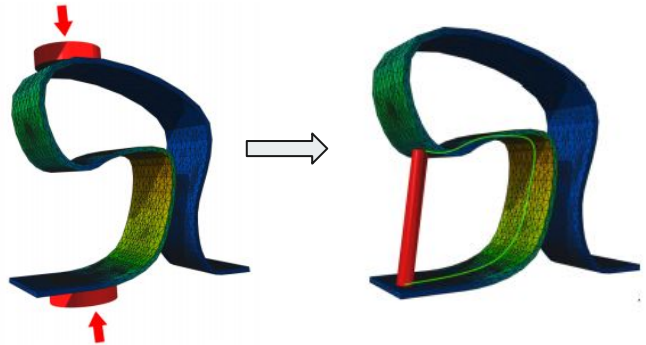

WARNING
HOT SURFACE
MakerBot
Replicator 2
Build from SD
Preheat
Utilities
MakerBot Replicator™2
DESKTOP 3D PRINTER

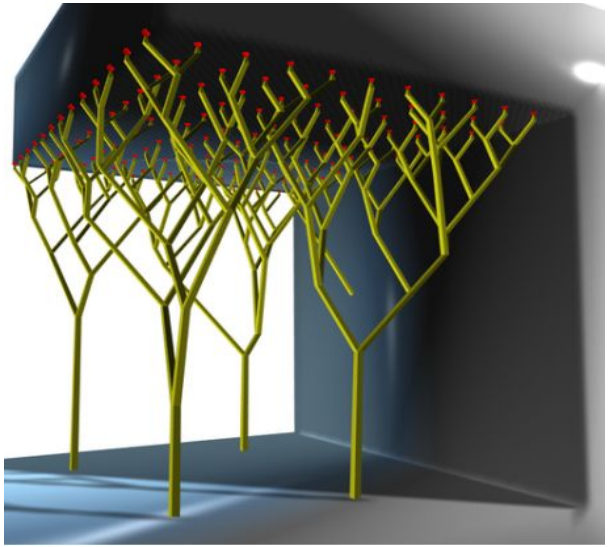
Related Work

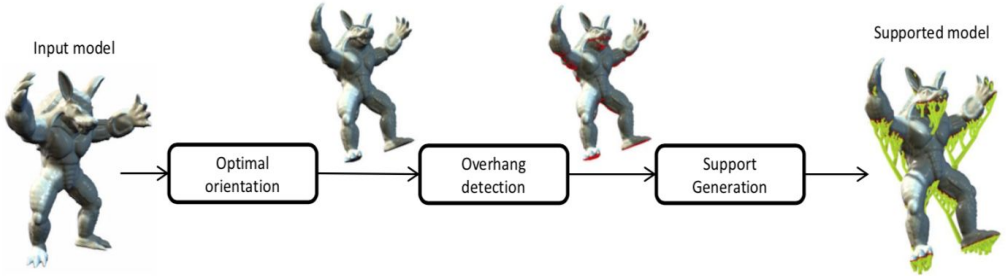
- Support Slimming for Single Material Based Additive Manufacturing K.Hu et al.
- Clever Support: Efficient Support Structure Generation for Digital Fabrication J.Vanek et al.



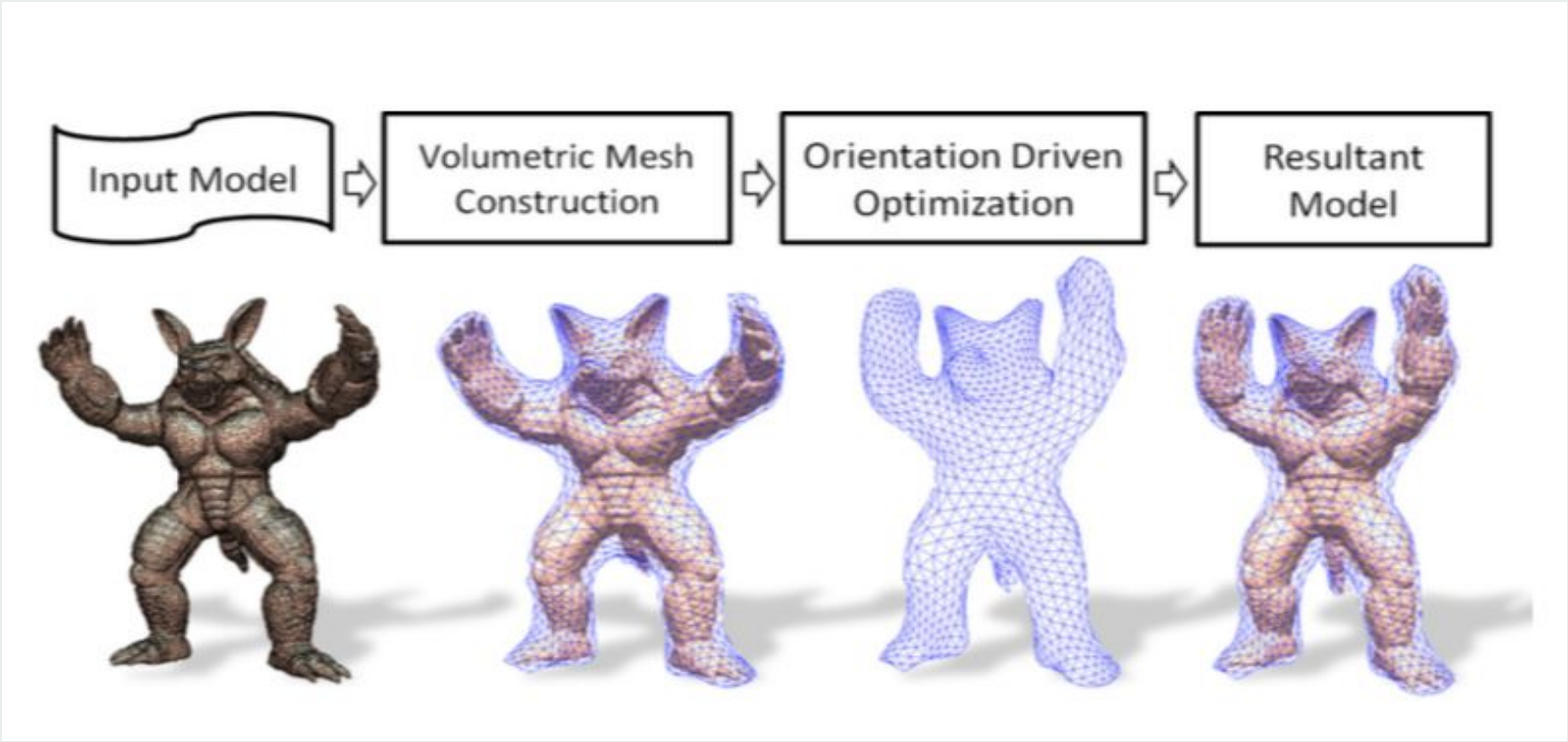
Related Work







Related Work



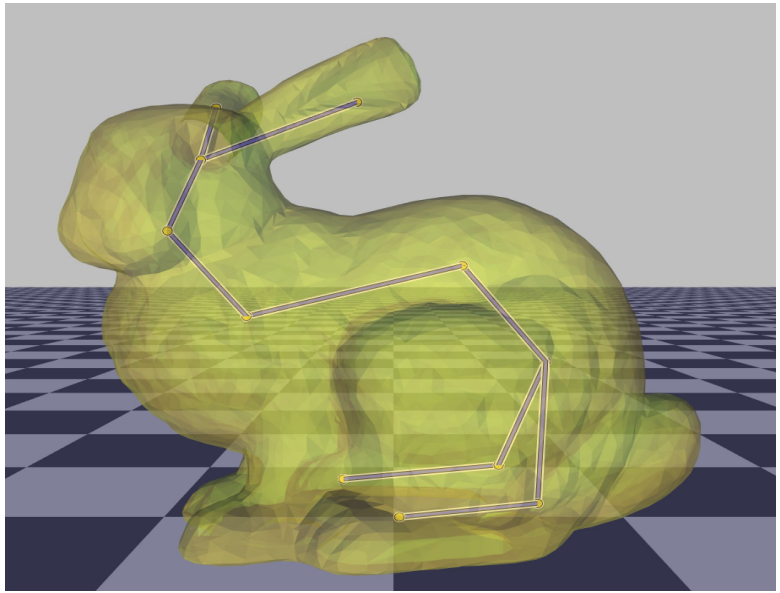
Our Method

- Specify skeleton handles, and compute bounded biharmonic weights $\mathbf{W}$
- Sample random rotations $\mathbf{T}$ for each bone, propagate via forward kinematics.
- Use linear blend skinning to find vertex positions $\mathbf{V}'$

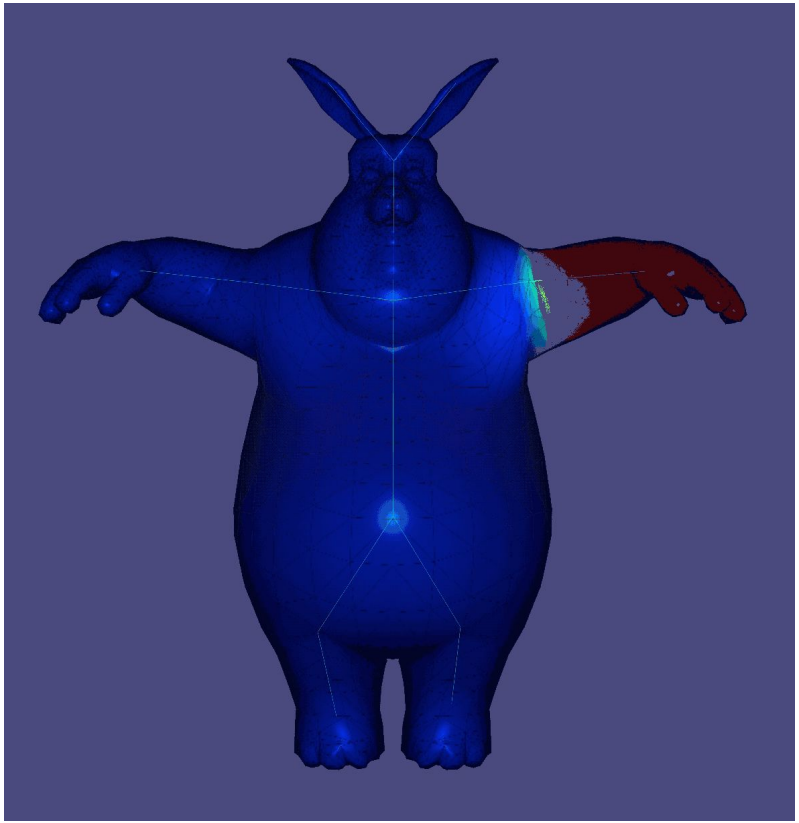
$$\mathbf{v}_i' = \sum_{j=1}^{m} w_j(\mathbf{v}_i)\mathbf{T}_j \begin{pmatrix} \mathbf{v}_i \\ 1 \end{pmatrix}$$

- Compute an objective function $E(\mathbf{T}) : \mathbb{R}^n \to \mathbb{R}$ to be optimized by a global optimizer

$$E = \omega_1 E_{arap} + \omega_2 E_{overhang} + \omega_3 E_{intersect}$$



Specify skeleton handles

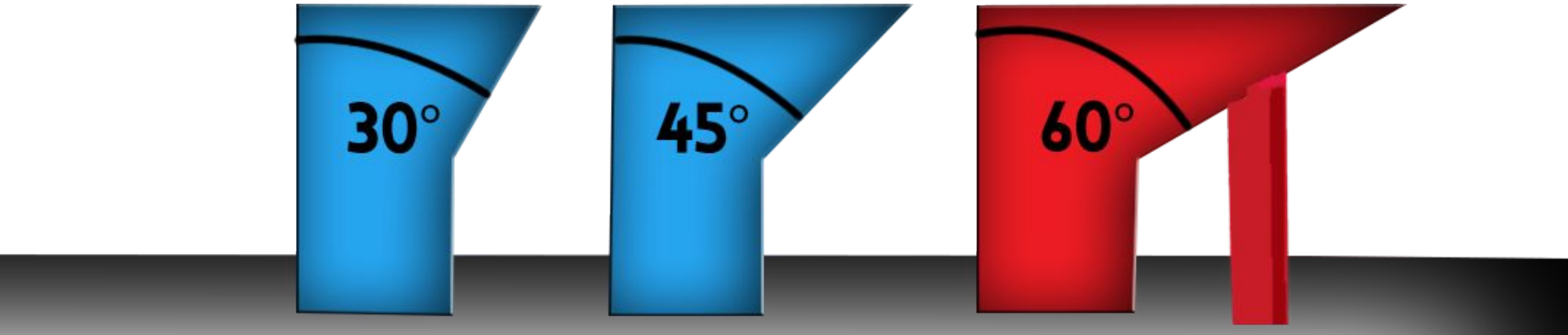


Bounded biharmonic weights

ARAP energy

$$E_{arap}(\mathbf{V}', \mathbf{R}) = \frac{1}{2} \sum_{f \in \mathbf{F}} \sum_{(i,j) \in f} c_{ij} ||(\mathbf{v}'_i - \mathbf{v}'_j) - \mathbf{R}(\mathbf{v}_i - \mathbf{v}_j)||^2$$

Overhang energy



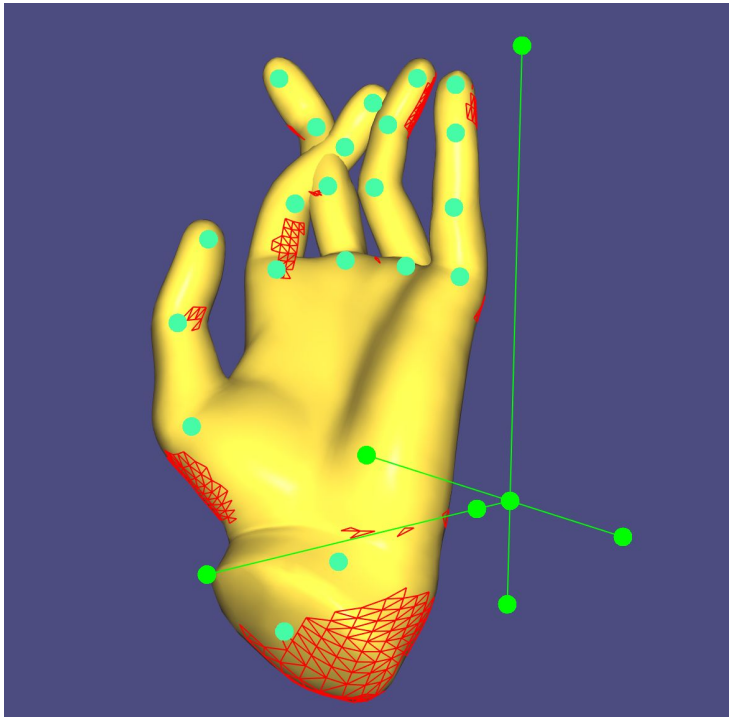
$$E_{overhang}(\mathbf{V}') = \sum_{f \in \partial \mathcal{M}} \lambda(f)$$

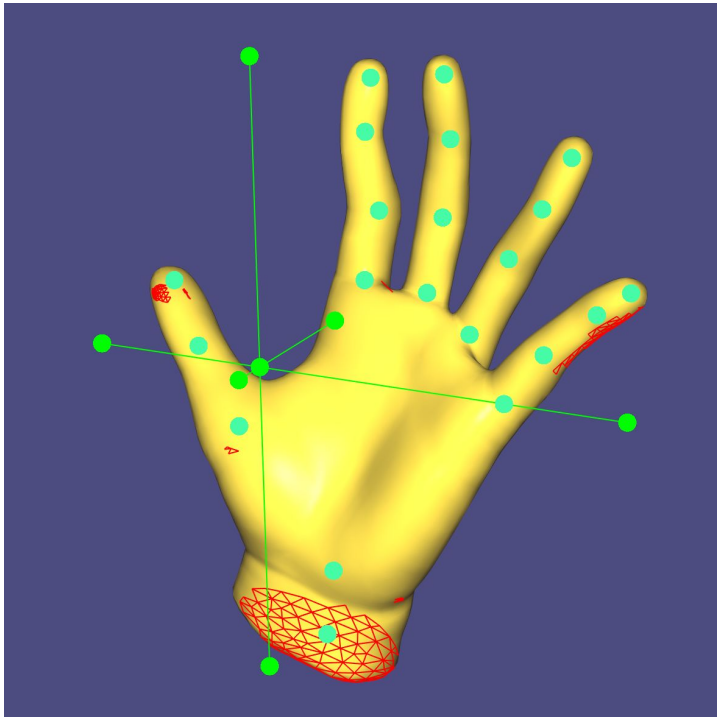
$$\lambda(f) = \begin{cases} A_{base} h = A_f |\mathbf{n}_f \cdot \mathbf{d}_p| (\mathbf{c}_f \cdot \mathbf{d}_p) & \text{if } \mathbf{n}_f \cdot \mathbf{d}_p < \tau \\ 0 & otherwise \end{cases}$$

- Overhang energy as volume underneath risky faces
- Volume under a single face approximated by a rectangular prism

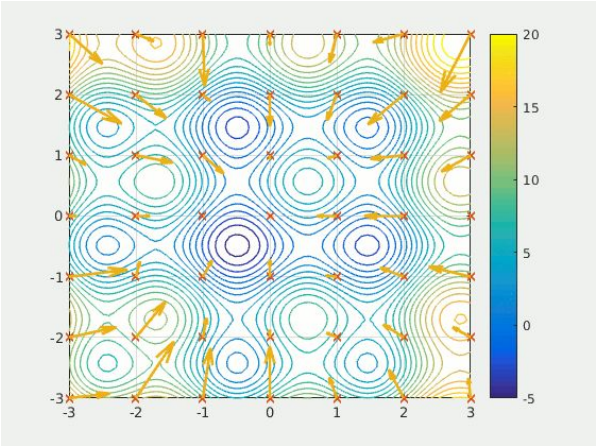
Self-intersection energy

- Define a grid below the model and draw a ray up from every grid point
  - Energy = sum of volumes of all self-intersecting regions
  - Finer grid = more accurate approximation of volumes, but slower
- More efficient: examine each face and determine which grid point(s) are beneath it



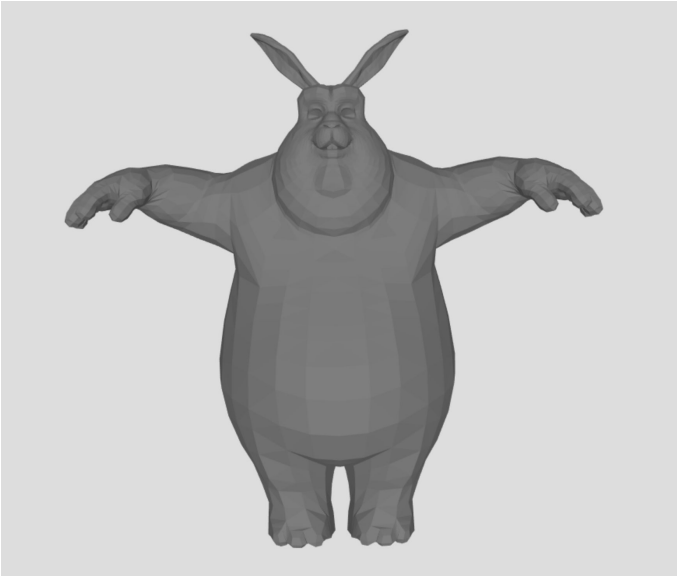


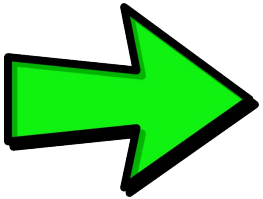
Particle Swarm Optimization

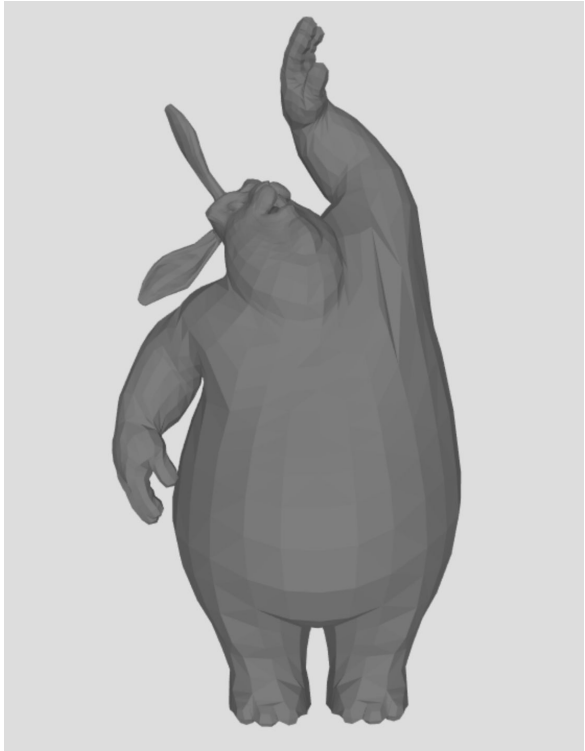


- Black box optimizer without the need for gradient
- Particle movement guided by local best and global best solutions

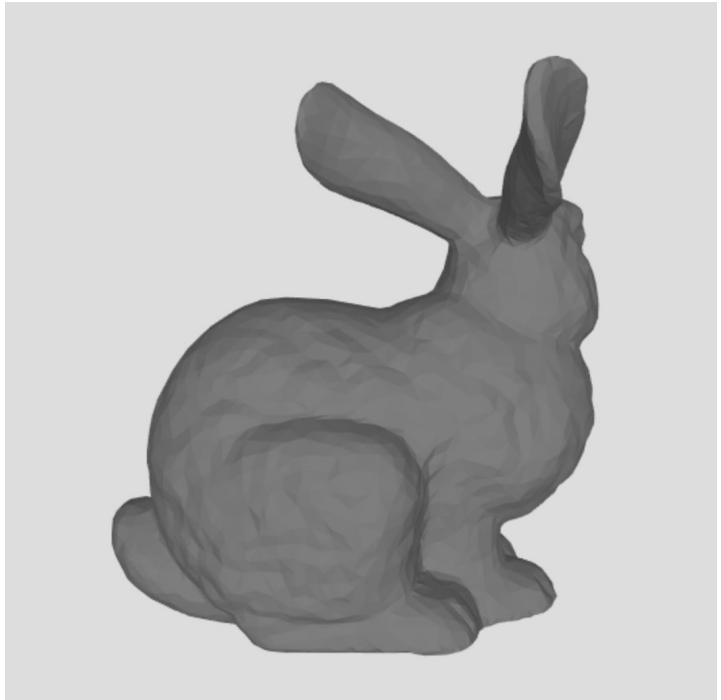
Results

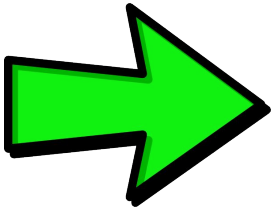


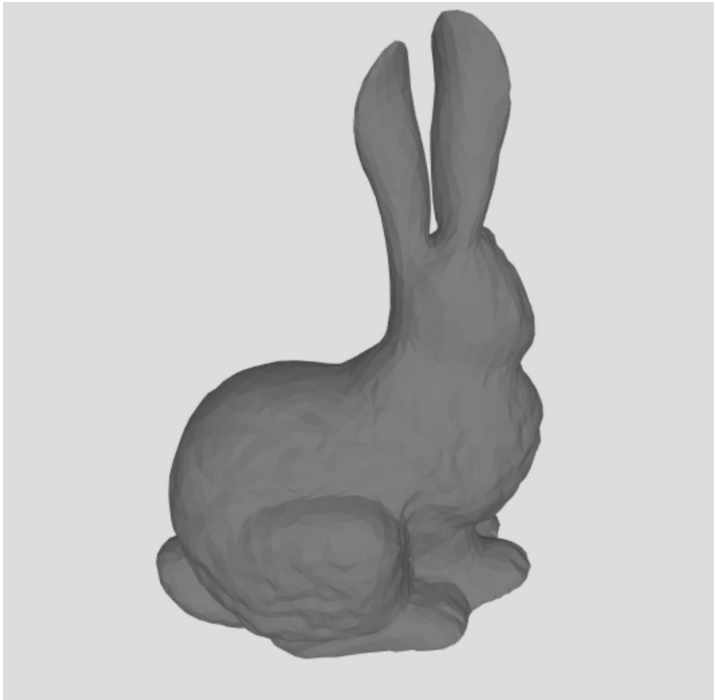




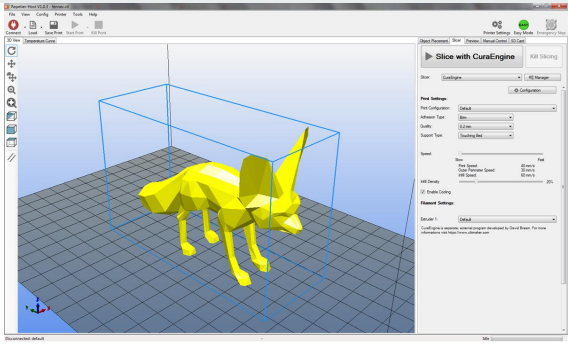

Support reduced by 15%





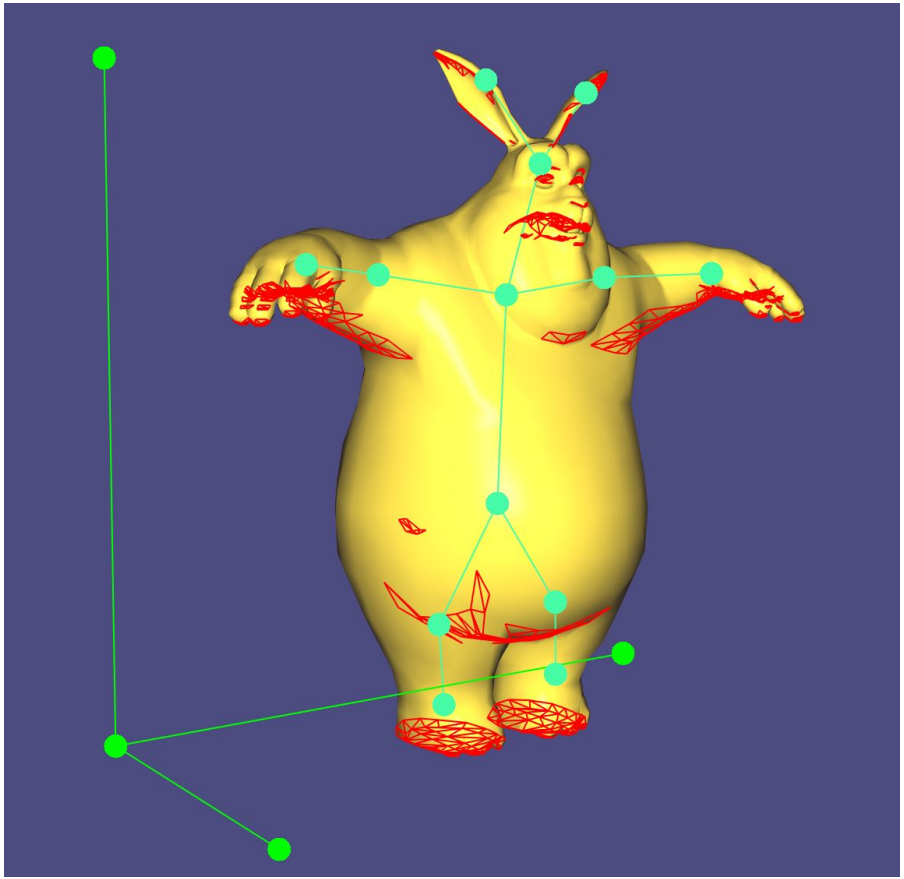


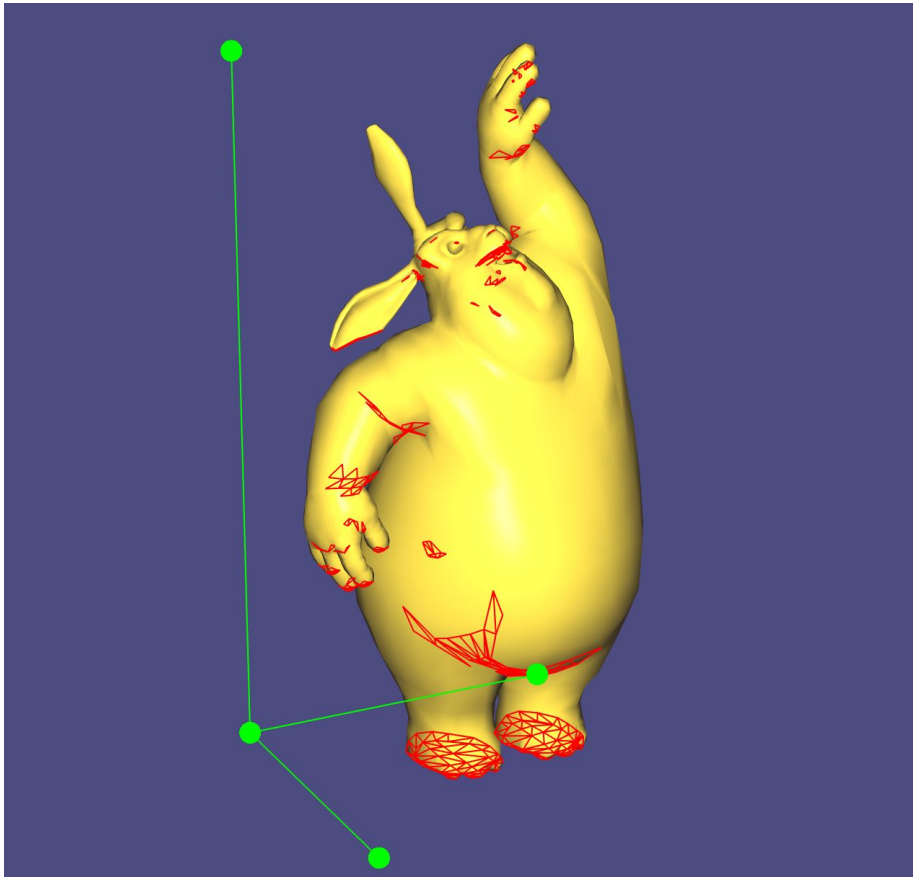

Support reduced by 37%



Future work

- Make it interactive
  - Optimize the pipeline with OpenGL for real-time computation
- Account for center of mass for balancing
  - *Make It Stand: Balancing Shapes for 3D Fabrication* (SIGGRAPH 2013)
- Account for support generation algorithm
  - Currently, the overhang energy does not reflect the reality
- Multi-objective optimization often conflicting (distortion vs. overhang)
  - Need more investigation on better methods





Questions?