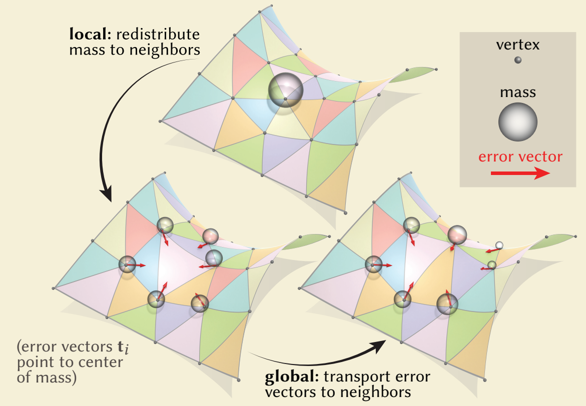
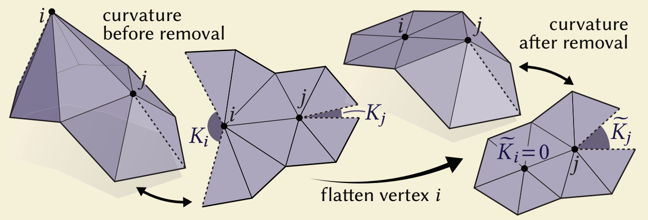
Surface Simplification using Intrinsic Error Metrics

Hsueh-Ti Derek Liu — University of Toronto & Roblox
Mark Gillespie — Carnegie Mellon University
Benjamin Chislett — University of Toronto
Nicholas Sharp — University of Toronto & NVIDIA
Alec Jacobson — University of Toronto & Adobe Research
Keenan Crane — Carnegie Mellon University

Intrinsic Curvature Error

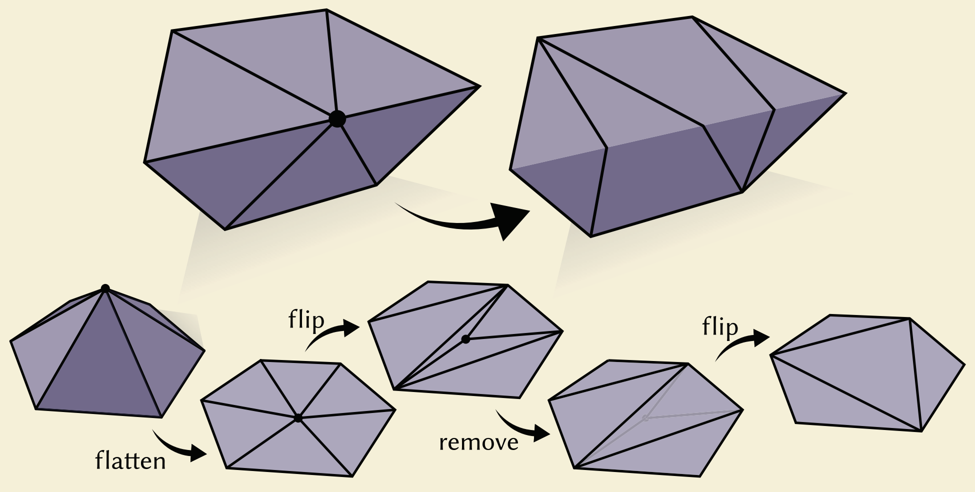


Our method constructs a coarse triangulation over a fixed geometric domain. In each **local** step we redistribute curvature or other quantities from a removed vertex to its neighbors. From step to step we also accumulate **global** information about error via tangent vectors pointing to the approximate center of mass of the decimated vertices.



Flattening a vertex i changes the angle sums Θ at neighboring vertices j, effectively redistributing the discrete curvature $K = 2\pi - \Theta$. We use the change in curvature to guide simplification.

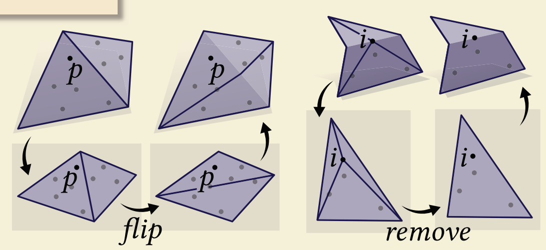
Intrinsic Vertex Removal

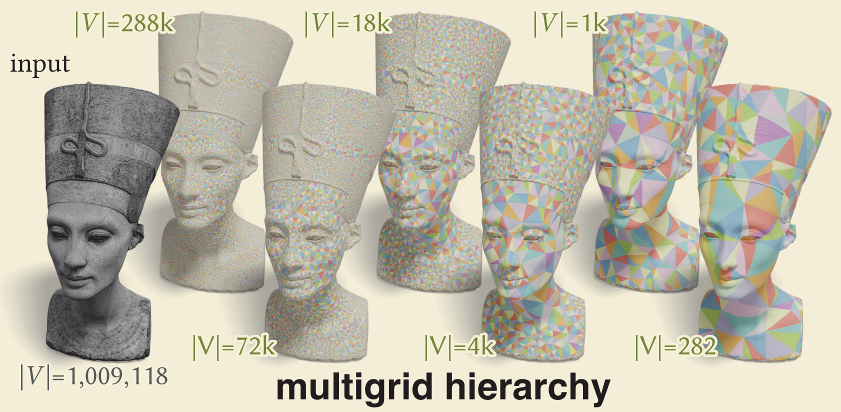


We decimate an interior vertex by intrinsically flattening it, flipping to degree 3, removing it from the mesh, then flipping back to an intrinsic Delaunay triangulation.

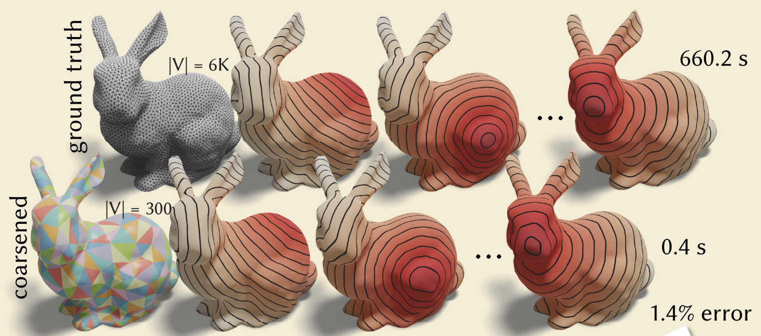
Pointwise Mapping

To map any point p on the fine mesh to a point p' on the coarse mesh, we track its barycentric coordinates through local coarsening operations (namely: edge flips, vertex flattenings, and vertex removals).



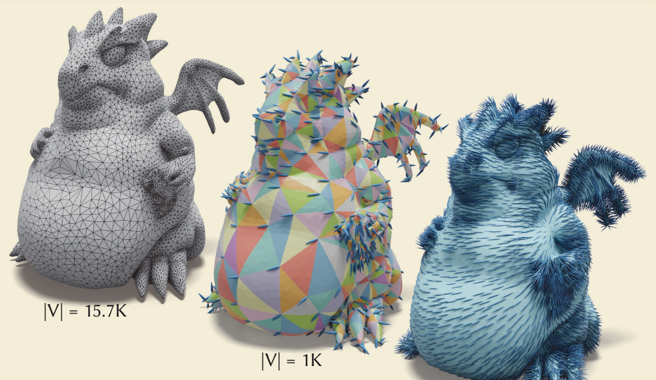


multigrid hierarchy

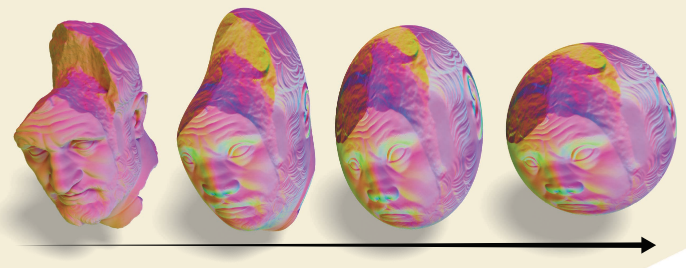


approximate all pairs geodesic distance matrix

1650x faster

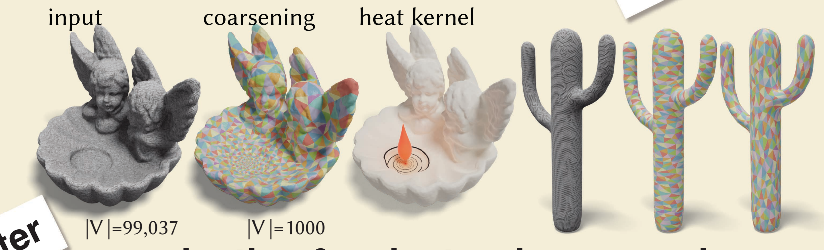


vector field design

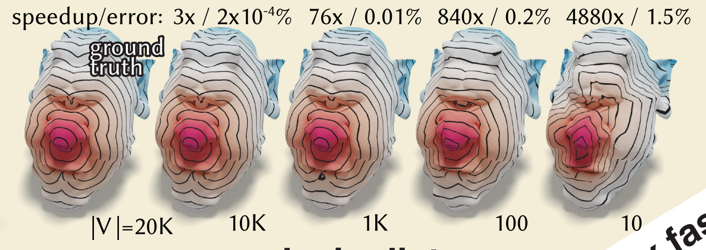


extrinsic curvature flow

20x faster

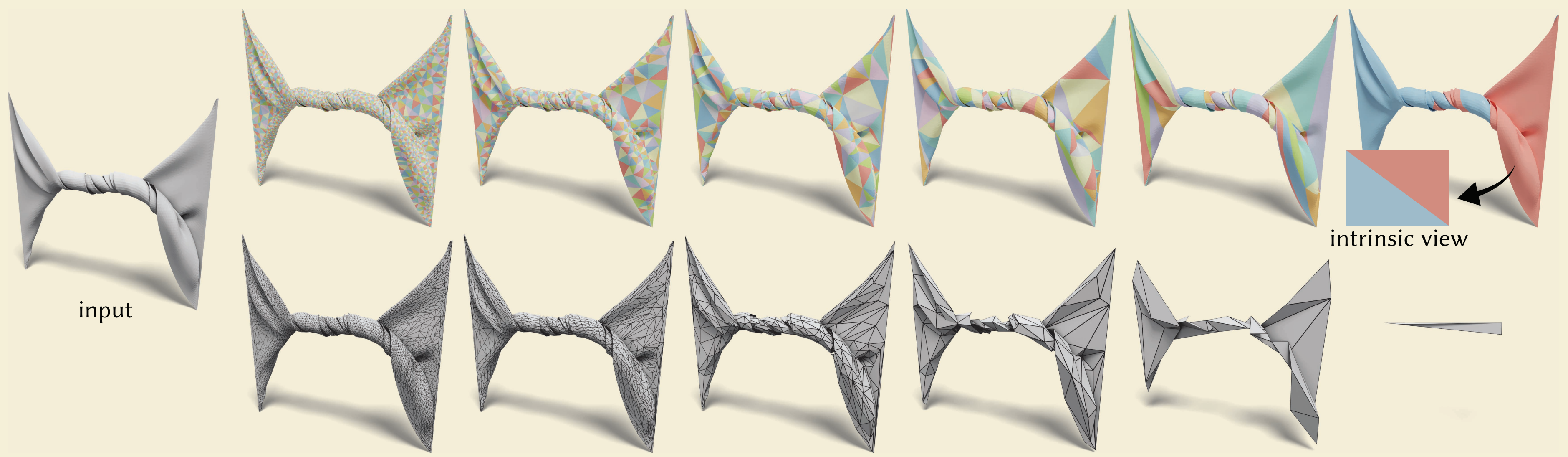


adaptive & anisotropic coarsening



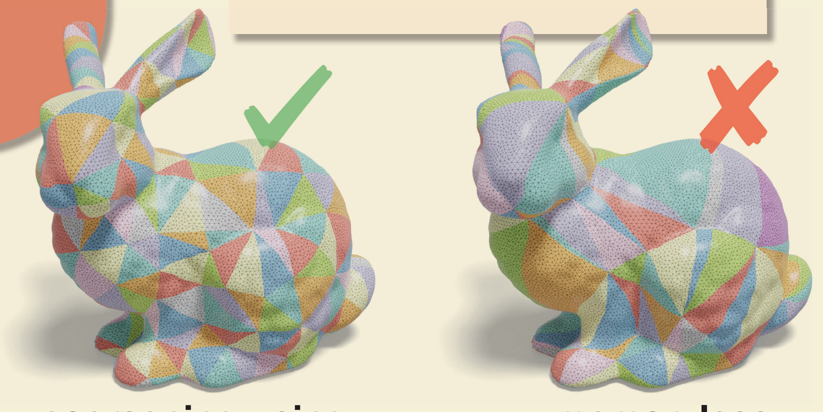
geodesic distance

4880x faster

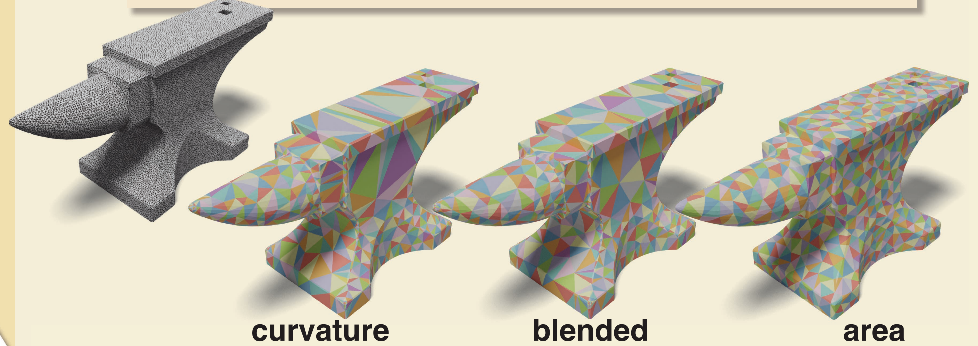


Whereas traditional extrinsic simplification *(bottom row)* must simultaneously juggle element quality and approximation error, triangles produced by our intrinsic scheme *(top row)* can wrap around the original surface—nicely approximating the underlying function space without changing the geometry. Coarse meshes or hierarchies produced by this scheme can be used in "black box" fashion to accelerate solvers without changing user inputs/outputs

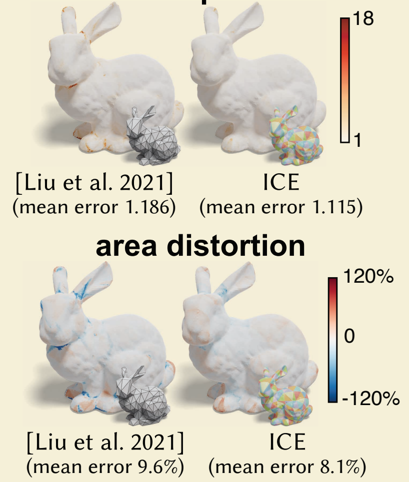
Accumulated Error Estimates



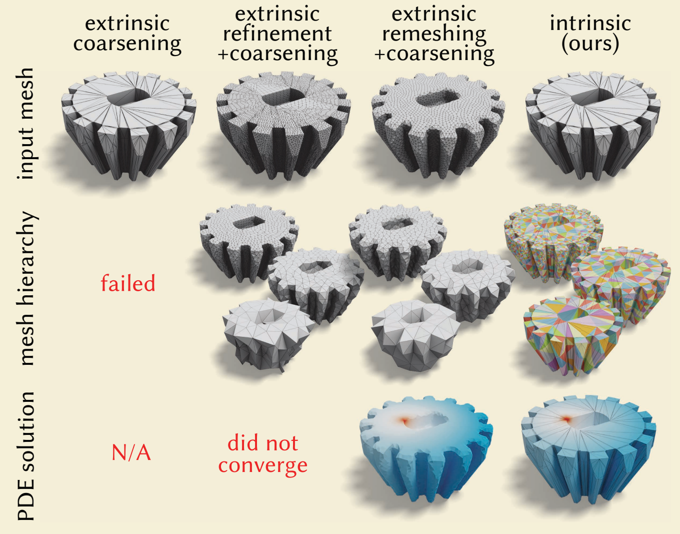
Using Other Transport Costs



Distortion



Robust Mesh Hierarchies



Acknowledgements

NSF, NSERC CRSNG, CRC Canada, The David & Lucile Packard Foundation, Alfred P. Sloan Foundation, Vector Institute, Fields, Data Sciences Institute, Adobe, Reality Labs, Google, Ontario, dgp Dynamic Graphics Project University of Toronto www.dgp.toronto.edu, Geometry Collective