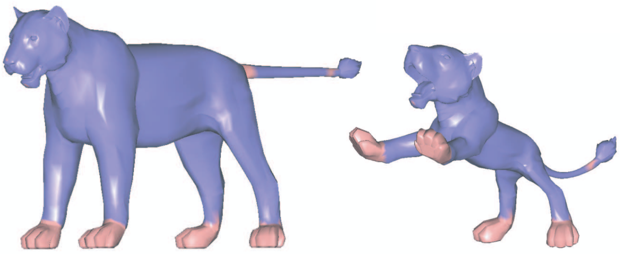
Mesh Editing with Curvature Flow Laplacian

Hong Kong University of Science and Technology
Oscar Kin-Chung Au (oscarau@ust.hk)
Chiew-Lan Tai
Hongbo Fu

Zhejiang University
Ligang Liu



- **Differential coordinates are essentially vectors encoded in the global coordinate system. In differential-coordinates-based mesh editing, they must somehow be transformed to match the desired new orientations, otherwise distortion like shearing and stretching will occur.**
- **We present an iterative Laplacian-based editing framework to solve this transformation problem.**
  - Positions of the handles are the only required input
  - No local frames are required
  - Supports point handle editing
- **Our iterative updating process finds the best orientations of local features, including the orientations at the point handles.**

Iterative Updating Framework

- To reduce distortion, the edited mesh should retain:
  - Parameterization information (shapes of triangles)
  - Geometry information (sizes of local features)
- Our system is based on Laplacian coordinates (LCs) $\mathbf{l}$ defined on mesh vertex position $\mathbf{V}$:

$$\mathbf{l}_i = \sum_{j \in i^*} w_{ij}(\mathbf{v}_j - \mathbf{v}_i),$$

In matrix form : $\mathbf{l} = \mathbf{L}\mathbf{V},$

We propose:
  - The magnitudes of LCs as geometry information
  - The coefficients of $\mathbf{L}$ as parameterization information
- We adopt the cotangent weighting scheme, because this LC approximates the curvature normal.

$$w_{ij} = \cot\alpha_{ij} + \cot\beta_{ij},$$

$$\mathbf{l}_i = \sum_{j \in i^*} w_{ij}(\mathbf{v}_j - \mathbf{v}_i) = 4\, Area_i \kappa_i \mathbf{n}_i,$$

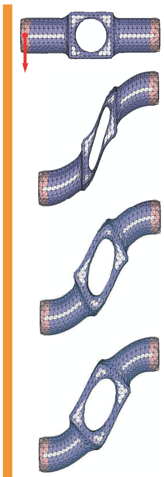
mean curvature; one ring triangles area; unit normal

- The system iteratively improves the vertex positions $\mathbf{V}$ and the LCs $\mathbf{l}$, minimizing parameterization and geometry distortions progressively.

Algorithm. Let $\mathbf{v}_i^t$ and $\mathbf{l}_i^t$ be the vertex positions and the LCs at time t, respectively, and let $\mathbf{v}_i^0 = \mathbf{v}_i$ and $\mathbf{l}_i^0 = \mathbf{l}_i$.

Step 1. Update the vertex positions
We use the current $\mathbf{l}_i^t$ and solve the normal equations (with the current handle positions as constraints) to compute the vertex positions $\mathbf{v}_i^{t+1}$.

Step 2. Update the Laplacian coordinates
We update the LCs to match the current deformed surface; that is, we fix the vertex positions $\mathbf{v}_i^{t+1}$ and compute the mean curvature normals as the new LCs $\mathbf{l}_i^{t+1}$, but scale the magnitudes to $|\mathbf{l}_i^0|$, in order to keep the original feature sizes.



(a) Input model with handles at two ends.

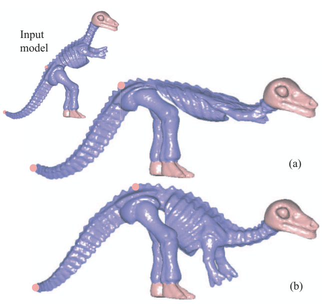
(b) without updating LCs; triangles are sheared and stretched.

(c) After one iteration.

(d) After 10 iterations.

Shearing effect is now reduced and the features are deformed naturally

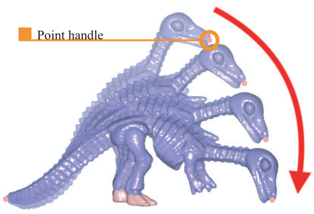
Editing with Handle Translation



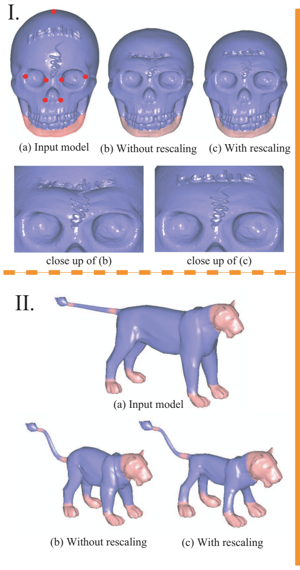
- (a) Only translation is specified at handles, thus no transformation change can be propagated.
- (b) Our iterative framework can reduce distortion caused by handle translation.

Editing with Point Handles

- The local orientation at the point handle is automatically decided by our system.
- Local frames at point handles are not required as input.



Rescaling of LCs



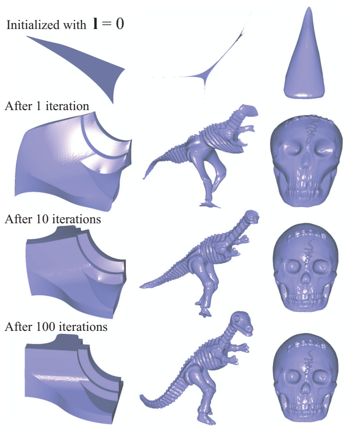
- When distances between handles are changed drastically, stretching or squashing distortion occurs. Merely reorientating the LCs cannot produce deformation with small parameterization error.
- Our system provides an option to rescale the LCs by the ratio of average edge lengths to reduce such anisotropic scaling:

$$d_i^{t+1} = \left\| \ell_i^0 \right\| \sqrt{Area_i^{t+1} / Area_i^0},$$

where $Area_i^0$ and $Area_i^{t+1}$ are the sums of the triangle areas sharing the vertex $\mathbf{v}_i$ in the original mesh and the mesh at time $t+1$, respectively.

I. Our system automatically rescales the LCs to eliminate undesired distortion, which is dependent on the geometry complexity and thus it is difficult for user to design a scaling field for the LCs.

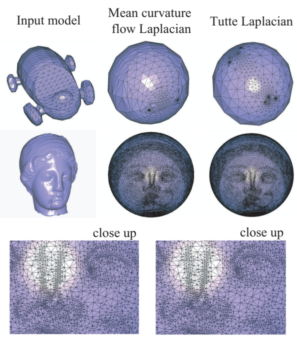
II. Editing example where the global feature is too big if the LCs are not rescaled.



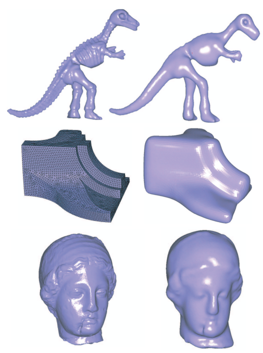
Reconstruction without Directional Information

- Our representation of geometry and parameterization information basically is the decomposition of the global vertex positions into local scalar information.
- We can reconstruct the models without using the directions of original LCs: only the magnitudes of the LCs (geometry information) and the Laplacian coefficients (parameterization information) are used.

Application : Spherical Mapping



- Spherical parameterizations using:
  - Mean curvature flow Laplacian
  - Tutte Laplacian
- This is done by setting the curvature field of the LCs to a constant value, and constructing the spherical mapping with our updating method.



Application : Non-shrinking Smoothing

- Smoothed models produced by smoothing the curvature field of the LCs and then reconstructing the models using our updating method.
- The curvature fields of all the examples were smoothed with 10 iterations.