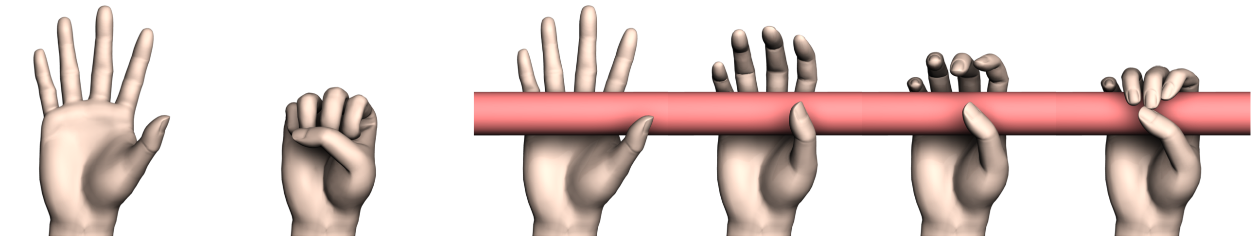
A Blendshape Model that Incorporates Physical Interaction

Wan-Chun Ma[1] Yi-Hua Wang[2] Graham Fyffe[1] Jernej Barbič[1] Bing-Yu Chen[2] Paul Debevec[1]

[1]University of Southern California [2]National Taiwan University



Source shape Target shape Shape blending while interacting with an object

1 Concept

We present a new technique for physically-plausible shape blending by interpolating the *spring rest length* parameters of a mass-spring system. This blendshape method begins by constructing two consistent mass-spring systems (i.e., with vertex-wise correspondence and the same topology) for source and target shapes, respectively, and setting the two systems as in their static states. In other words, their edge lengths equal to the rest lengths of the springs. To create an intermediate pose, we generate a new mass-spring system consistent with the source and target ones and set its rest lengths as linearly interpolated between source and target based on an interpolation factor $\alpha \in [0, 1]$. The new pose is then synthesized by computing the equilibrium given the interpolated rest lengths. In addition, the mass-spring system may interact with other objects in the environment by incorporating collision detection.

2 The Blendshape Model

We make the assumption that the source and target shapes have vertex-wise correspondence. To preserve their volumes, we first add some additional corresponding vertices inside the two shapes. Then a common topology is created for the structure of the volumetric mass-spring systems based on 3D Delaunay triangulation. Similar to [Choi and Ko 2002], we used a combination of structure springs and bending springs, which model the elastic properties of the object's surface and the bending and flexural properties of the material.

Spring-Space Shape Interpolation. For a given interpolation factor α, the interpolated shape is the equilibrium state of the mass-spring system with $r'_{ij} = (1-\alpha)r^s_{ij} + \alpha r^t_{ij}$, where r_{ij} is the rest length of a spring that connects vertices v_i and v_j, and r'_{ij}, r^s_{ij}, and r^t_{ij} indicate the interpolated, source, and target spring rest lengths, respectively. In an equilibrium state, the total force $f(v_i)$ acting on each vertex v_i in the mass-spring system equals zero:

$$f(v_i) = \sum_{j \in \mathbf{n}(i)} k_i(\|v_i - v_j\| - r'_{ij}) \frac{v_i - v_j}{\|v_i - v_j\|} = 0, \quad (1)$$

where $\mathbf{n}(i)$ is the set of neighbors that are adjacent to v_i, and k_i is the spring constant. Currently we set all the spring constant of all springs to be uniform. Note that we ignore mass and velocity since our formulation is based solely on quasi-static states. We solve the system using the Newton–Raphson method to determine the first order approximation of the optimal shape positions:

$$f(x_{t+1}) \approx f(x_t) + J(x_t)\Delta x_t, \quad (2)$$

where x_t and x_{t+1} are the shape position vectors from the previous frame and current frame, respectively, and $x_{t+1} = x_t + \Delta x_t$. $J(x_t) = \frac{\partial f}{\partial x}(x_t)$ is the global stiffness (Jacobian) matrix of f evaluated at the current positions of vertices x_t. Since we are solving the equilibrium state, we require $f(x_{t+1}) = 0$. Hence, Eq. (2) becomes a sparse linear system: $J(x_t)\Delta x_t = -f(x_t)$. Solving for Δx_t determines x_{t+1}, and then we iterate it until $\|\Delta x_t\|$ is smaller than a threshold.

Collision Detection and Handling. Our model can also interact with other objects during the interpolation. For each interpolation iteration, we perform collision detection between the mass-spring system and objects. Once a collision is reported, we move the intersecting vertices to the surface of the object. Next, we add all the intersecting vertices as additional boundary conditions, and then recompute the equilibrium state. This produces a result which preserves the original shape's features as much as possible, while respecting the collision constraints.

3 Results

The proposed method is capable of producing fully automatic shape interpolation, with low distortion of surface area and volume. While our method does not guarantee volume preservation, it still maintains its volume as much as possible from the support of internal vertices and springs. Blending between multiple shapes in spring space can also be easily done by writing the rest length as a convex linear combination of the spring rest lengths from n_p input shapes: $r'_{ij} = \sum_{k=0}^{n_p} w_k r^k_{ij}$. Consequently, many of the capabilities of classical blendshapes can be easily transfer to our method. Finally, we have shown that it is possible to incorporate collision detection and handling to allow the blendshape model to interact with other objects.

References

Choi, K.-J., and Ko, H.-S. 2002. Stable but responsive cloth. In *Proc. of SIGGRAPH 2002*, 604–611.

SIGGRAPH Asia 2011, Hong Kong, China, December 12 – 15, 2011.
ISBN 978-1-4503-0807-6/11/0012