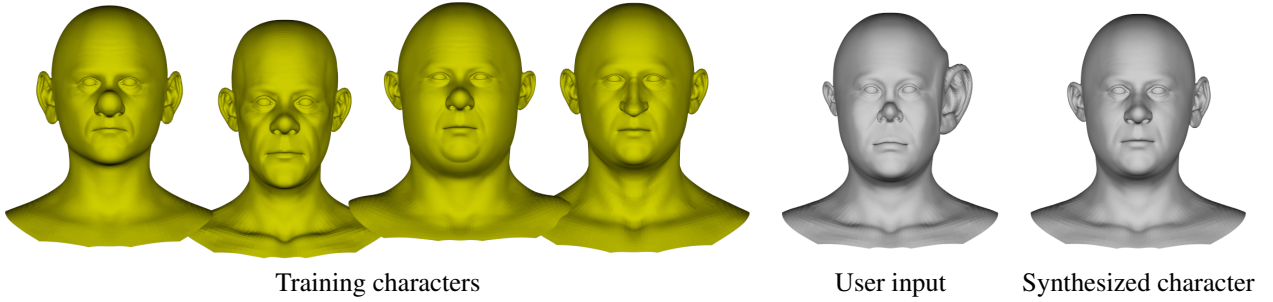


A Guided Synthesizer for Blendshape Characters

Wan-Chun Ma*

J.P. Lewis†

Weta Digital



We have designed an interactive editing system that will help users to create new characters based on a limited number of training characters. The user provides a desired preliminary shape of the new character. The output of our system will be a convex linear combination of the existed characters in a local sense while maintaining the proximity to the user desired shape as much as possible. This feature guarantees the creation process is guided by the information of artists' previous works. The system first finds the best set of training characters and blending weights that represents the user desired shape, then computes the result shape, which matches the obtained composition, using an edge-based integration.

Optimal Composition Assume both of the input shape s and character targets $T = \{t_i | 1 \leq i \leq n_t\}$ are corresponded and represents a similar expression and (e.g. neutral). Following [Ma et al. 2011], the optimal composition is looking for the best set of training characters $P = \{p_i | 1 \leq i \leq n_v\}$ and blending weights that minimizes geometrical reconstruction error in a local sense, while maintaining a global spatial-coherence with respect to the character targets used. We can formulate the problem into an optimization based on the above criteria:

$$\min_P \mathbf{E} = \mathbf{E}_g + \alpha \mathbf{E}_c. \quad (1)$$

\mathbf{E}_g is a function that measures the geometrical reconstruction error. It returns the sum of the Euclidean distance between $\mathbf{r}(s, i)$, the local geometrical measurement of the input shape around the i th vertex, and its approximation from the character targets $\hat{\mathbf{r}}(T, p_i, i)$ based on the convex linear interpolation of a character set represented by p_i :

$$\mathbf{E}_g(P) = \sum_{i \in V} \|\mathbf{r}(s, i) - \hat{\mathbf{r}}(T, p_i, i)\|^2, \quad (2)$$

$$\hat{\mathbf{r}}(T, p_i, i) = \sum_{j=1}^{n_k} w_j^i \mathbf{r}(t_{p_j^i}, i). \quad (3)$$

$\sum_{j=1}^{n_k} w_j^i = 1$, $w_j^i \geq 0$, and n_k is a pre-defined number of targets in a character set. Different than [Ma et al. 2011], we define \mathbf{r} to be the Laplacian operator because its translational invariant property, since the character targets are usually in different sizes and locations. Equation 3 can be solved by constrained least squares.

\mathbf{E}_c is a function that measures the cost of assigning a character set label p^i to vertex i with respect to its neighbor vertex j . \mathbf{E}_c returns

the sum of the difference between each adjacent pair p^i and p^j such that it encourages adjacent vertices to use similar targets as much as possible. The Hamming distance function \mathbf{h} is used to model \mathbf{E}_c :

$$\mathbf{E}_c(P) = \sum_{(i,j) \in E} \mathbf{h}(p^i, p^j). \quad (4)$$

Equation 1 can be regarded as finding the maximum a posteriori (MAP) estimator of a Markov random field (MRF). Exact inference is computationally intractable in general. Loopy belief propagation is applied to provide an approximate solution. α controls the balance between the geometrical error term and the coherence term. To reduce the complexity for speed-up, both of the input shape and the character targets are resampled into a lower resolution graph. The weights for each vertex in the original mesh can be interpolated from the nearby samples by radial basis functions or other scattered data interpolation techniques.

Character Synthesis Once blending weights at each vertex are recovered, instead of directly blending the vertex positions (which is invalid, as the same reason of using the Laplacian operator for optimal composition), we blend the edge vector of each character target according to the weights:

$$d^q = \sum_{k=1}^{n_t} w_k^i (v_k^i - v_k^j). \quad (5)$$

d^q denotes the q th edge vector that connects the i th and j th vertices. The vertex positions u of the new shape is the solution of the linear system $Mu = d$, where M is the oriented incidence matrix based on the undirected graph G , and d stores all the edge vectors. Note that M is not full rank, so we have to set at least one vertex a fixed value as a constraint. Since the edge-based integration favors preserving the edge lengths of the character targets, locally the surface will conform to the training character targets better. The same process of blending/integration can be repeated to construct other expressions in the character asset based on the same weights. The overall algorithm generates a greater variety of characters than would be possible using simple linear blending, while nevertheless reflecting the types of geometric variation evident in the training shapes.

References

MA, W.-C., FYFFE, G., AND DEBEVEC, P. 2011. Optimized local blendshape mapping for facial motion retargeting. In *ACM SIGGRAPH 2011 Talks*.

*e-mail: wma@wetafx.co.nz

†e-mail: noisebrain@wetafx.co.nz