Face Rigging through Curvenet Parametrization

Jeremie Talbot
jtalbot@pixar.com
Pixar Animation Studios
Emeryville, CA, USA

William Sheffler
sheffler@pixar.com
Pixar Animation Studios
Emeryville, CA, USA

Fernando de Goes
fernando@pixar.com
Pixar Animation Studios
Emeryville, CA, USA

Figure 1: Poses of characters from *Hoppers* (2026) whose face rigs were constructed using curvenet parametrization. On the left side of each animal, we exhibit in purple the face curvenet overlaid on top of the face surface.

Abstract

We present a new technique for generating parametrization weights that eases the construction of face rigs. At its core, our approach makes use of curvenets as a sparse yet flexible geometric proxy for authoring parametrization contours aligned to user-selected face profiles. After outlining curvenet weights, we present an optimization routine that interpolates these sparsely authored weights into weight maps throughout the mesh connectivity of the face surface, which are then used to drive the face articulation. We demonstrate our curvenet parametrization using results of face rigs built for various characters from Disney and Pixar's feature *Hoppers* (2026).

ACM Reference Format:
Jeremie Talbot, William Sheffler, and Fernando de Goes. 2026. Face Rigging through Curvenet Parametrization. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Talks (SIGGRAPH Talks '26), July 19–23, 2026, Los Angeles, CA, USA.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3799818.3812074

1 Introduction

Since *The Incredibles* (2004), the character team at Pixar has approached face rigging as a two-step process that first parametrizes facial regions onto NURBS patches, and then weights the deformation within each region based on the 2D coordinates defined by these patches. For instance, the mouth-lip parametrization is authored with a u-parameter to respond to left-right controls, versus a v-parameter to separate the upper-lower lip geometry while specifying the jaw influence. Despite the extensive production usage over the years, constructing face parametrizations has persisted as a time-consuming task. In particular, the shape and resolution disparities between NURBS patches and face surfaces make naive parametrization solutions such as closest-point projections error-prone. Moreover, the rectangular grid layout of NURBS patches are often considered too strict to incorporate geometric features into the face parametrization. As a result, artists have overridden the face parametrization by manually weighting the coordinates for each mesh point.

To overcome these rigging issues, we developed a new approach for generating face parametrizations that replaces NURBS patches with curvenets. Unlike NURBS patches, curvenets are collections of parametric curves that can be drawn over the surface mesh in a free-form manner with no layout restriction. By adopting curvenets as the parametrization proxy, artists can conform the parametrization coordinates onto the face anatomy more precisely, while still weighting a small number of points. Besides the typical uv parametrization, we also leverage curvenets to design additional coordinates, such as radial maps from lips outward, making the correspondence between facial cues and parametrization weights more direct and, thereby, easing the setup of face rigs. Once the curvenet is created and weighted, we compute the face parametrization through a surface-based solver that interpolates the curvenet weights to the mesh points in a vectorized fashion. We describe next the implementation of our curvenet parametrization technique and showcase face articulations produced with our tool in Pixar's feature *Hoppers* (2026).

2 Curvenet Parametrization

Introduced by de Goes et al. [2022], curvenets are Presto-specific primitives providing a curve-based rigging representation that separates the articulation controllers from the deforming surface. Artists pose curvenets by simply moving their points with any preexisting rigging techniques such as skinning and sculpting, while the surface deformations are generated afterwards through a custom deformer, named the *Profile Mover*, that interpolates the deformation of the rigged curvenet over the surface mesh. In past shows, we have built face rigs using a hybrid approach that poses curvenets

SIGGRAPH Talks '26, Los Angeles, CA, USA

ACM ISBN 979-8-4007-2541-8/2026/07
https://doi.org/10.1145/3799818.3812074

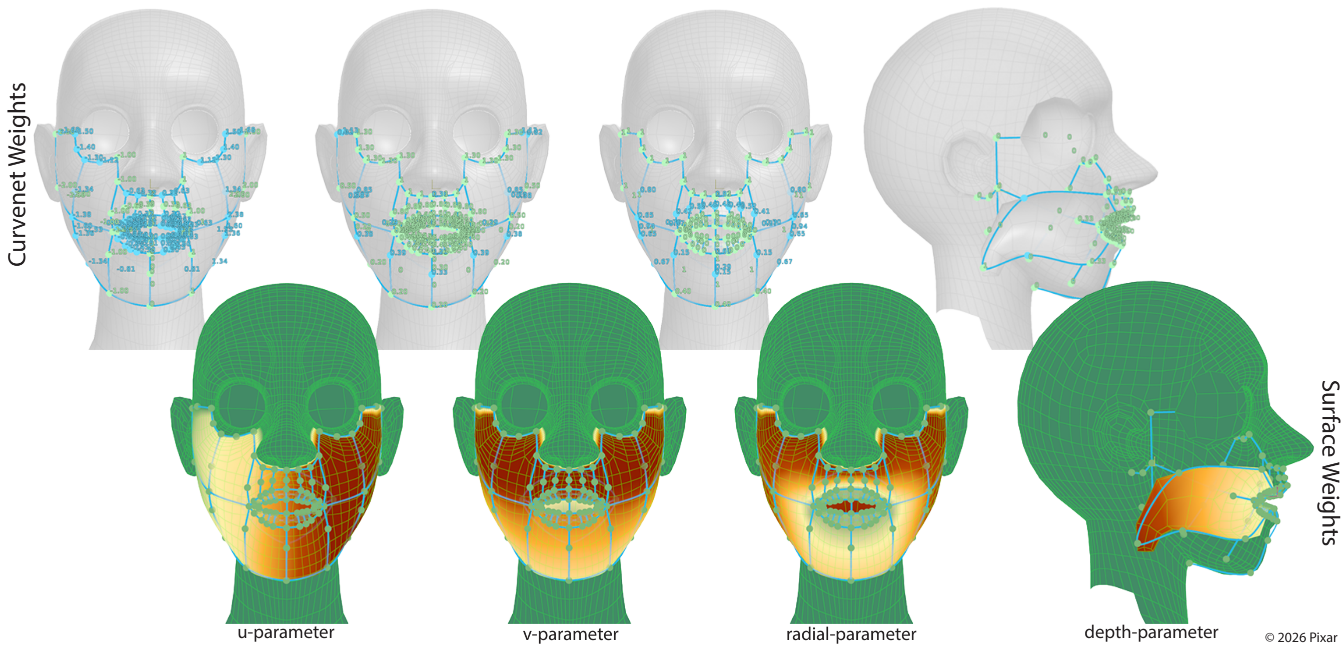

Figure 2: The top row displays the four coordinates defining our mouth-lip parametrization authored on the curvenet, with values in cyan corresponding to *auto-smoothed* points, while values in green explicitly set by the user. The bottom row shows the face parametrization generated by interpolating the curvenet weights over the surface mesh color-coded from white to red.

based on NURBS patches coordinates, before propagating the curvenet deformation to the face surface via the Profile Mover, e.g., see the Panda characters from *Turning Red* (2022) or the aliens from *Elio* (2025), to cite a few. While adopting curvenets reduced the number of rigged points, the limitations inherited by computing patch-based parametrizations remained and artists continued to manually weight face coordinates point-by-point. In this work, we draw inspiration from the surface-aware deformation interpolation devised for the Profile Mover, and repurpose this computation to propagate the parametrization coordinates from the curvenet throughout the surface mesh.

Our approach starts by drawing a curvenet over the neutral pose of a face shape using the modeling package available in Presto. In this system, artists can insert points on the surface mesh and click-and-drag curves connecting these points. In contrast to our previous curvenet use cases, we chose centripetal Catmull-Rom as our curve basis type so that every curve point is rooted on the surface exactly. Once the curvenet is constructed, we define parametrization outlines by authoring weights directly on the curvenet points. For instance, our new mouth-lip parametrization is now made of four weight maps, including the usual left-right and upper-bottom uv coordinates, in addition to a radial coordinate that assigns zero to the lips edges and one at the extent of the mouth, and a depth coordinate which is zero at the face surface and one at the back of the throat. To ease weight authoring, we can also mark curvenet points with a special *auto-smooth* tag, indicating that the weight at those selected points are implicitly assigned to a smooth interpolation of the weights from the neighboring points along the curvenet. By doing so, manual weighting is required only at a few curvenet points, and the weights for every other point are computed on-the-fly by solving a discrete Laplacian equation over the curvenet connectivity. Figure 2 illustrates our mouth-lip parametrization color-coding the curvenet setup with auto-smoothed points in cyan and hand-weighted points in green. Since curvenets and weights are both authored in Presto, we can edit the curvenet layout and update the parametrization weights interchangeably.

After setting the curvenet with the desired face coordinates, we extend the face parametrization over the surface mesh through a new weight-object primitive implemented in Presto. Similar to de Goes et al. [2022], this tool first rasterizes the curvenet into a dense polyline with sample density set to five times the ratio between the mean curve length of the curvenet and the mean edge length of the underlying surface mesh. In this process, we also assemble a matrix $\mathbf{S}$ encoding the subdivision stencil mapping control points of each centripetal Catmull-Rom curve to the sampled points, thus allowing the propagation of each weight map w from the curvenet points towards the sample points. Additionally, we compute another matrix $\mathbf{B}$ containing the barycentric coordinates for each curvenet sample projected onto the closest-point of the face surface. Equipped with these two matrices, we generate the corresponding parametrization weights x on the points of the surface mesh by minimizing the objective function $x^t\mathbf{L}x + \kappa\|\mathbf{B}x - \mathbf{S}w\|^2$, where $\mathbf{L}$ is the polygonal Laplacian matrix and κ is a stiffness scalar that balances surface smoothness versus reproducing the input weights at the projected samples, with default set to 100 times the mean length of mesh edges. The solution to this optimization amounts to a sparse linear system, which we factorize once and reuse it for multiple input weights w, thus producing all face coordinates x over the surface mesh in parallel.

3 Results

We deployed our curvenet parametrization in the face module of our shared human rig, which is inherited by every human asset in our shows. When rigging a new human character, the rigger needs to refit the points of predefined face curvenet onto the new face shape, and then the face weights are generated automatically. If necessary, new points and even curves can be easily inserted to the face curvenet while preserving the existing weights. To improve the curvenet fitting over the surface mesh, we typically sculpt the mouth surface to make the lips slightly separated during the generation of the face parametrization. For assets with the same mesh connectivity as our shared human, the refit can be performed more easily via a single warping tool. To further customize the face deformation, we also allocate rig splines for each component of the face rig (e.g., mouth, lips, jaw, brows), enabling fine-tuning of the surface weights passed to each face deformer. In our supplemental video, we include live Presto recordings exemplifying the authoring of face deformations by manipulating either the curvenet weights directly or the splines remapping the surface weights. It is worth reiterating that we made no modifications to the articulation controllers neither to the deformation stack present in our face rigs, instead we simply updated the generation of face weights. Our tool was also employed for rigging the face and brow articulation of various animal characters from *Hoppers* (Figure 1). Adopting curvenets was particularly impactful for handling the broad range of face shapes of these different animal groups. Finally, we provide clips of expression tests for selected characters in the supplemental video.

References

Fernando de Goes, William Sheffler, and Kurt Fleischer. 2022. Character articulation through profile curves. *ACM Trans. Graph.* 41, 4, Article 139 (July 2022), 14 pages. doi:10.1145/3528223.3530060