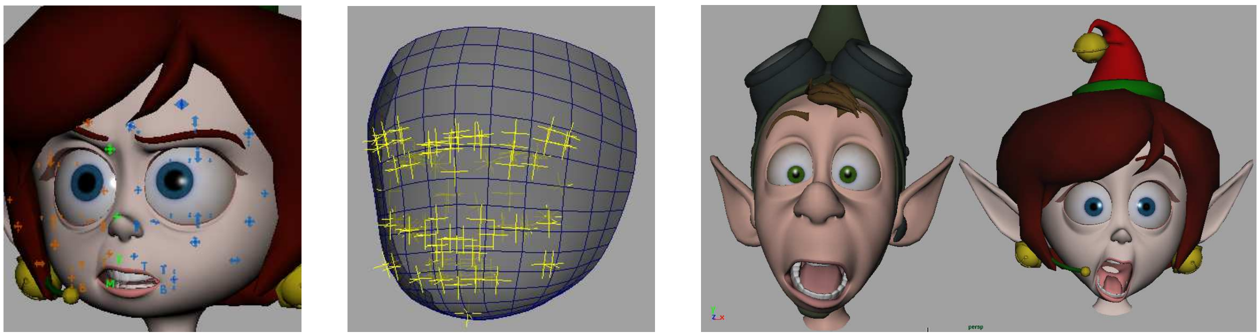
A Hybrid Approach to Facial Rigging

David Komorowski* Walt Disney Animation Studios
Vinod Melapudi† Sidhe Interaction
Darren Mortillaro‡ Sony Pictures
Gene S. Lee§ Walt Disney Animation Studios



(a) *Controls on Face* (b) *Locators on Skull* (c) *Animation Retargeting from One Rig to Another*

Figure 1: *Facial Rigging using Pose Space Deformation*

1 Introduction

In production environments, facial rigging is commonly done by either geometric deformations or blendshapes. Geometric deformations are driven by simulated muscle actions, which are loosely based upon the dynamics of facial tissue [Magnenat-Thalmann et al. 1988]. Blendshapes interpolate a large number of sculpted shapes [Bergeron and Lachapelle 1985]. The former approach is intuitive, yet slow and less precise. The latter is fast, yet memory intensive and sensitive to model changes. Conventional implementations of both approaches are difficult to generalize in order to build rigs quickly and retarget animation efficiently.

This paper introduces a general approach to facial rigging that is a hybrid of the two approaches. It uses *pose space deformation* (PSD) [Lewis et al. 2000] to seamlessly combine both geometric deformations and blendshapes. PSD is a technique for scattered data interpolation that is useful in manipulating both numerics and shapes [Lee and Hanner 2009]. In the context of this work, PSD indirectly influences the application of geometric deformations and, if required, the interpolation of blendshapes. The generality of our approach is achieved by establishing a common, fixed PSD system for all character rigs. It is with this approach that the characters of "Bolt" and "Prep and Landing" were quickly rigged and animated.

2 Hybrid Approach

The basic architecture of our hybrid approach appears in Figure 2. It consists of components with two types of data flowing between them, solid-red for geometry and dotted-blue for numerics. The deformers and blendshapes regulate the appearance of the face. The deformers begin with a neutral face and progressively apply geometric deformations. The blendshapes refine these deformations globally or in select areas of the face. The deformers and blendshapes are regulated by PSD and an action table, a specialized data structure for facial animation. PSD animates

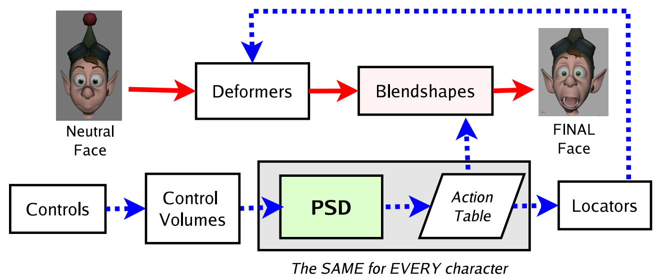


Figure 2: *Architecture of Hybrid Approach*

the values of the action table according to the positioning of controls, which are simply UI widgets.

Both PSD and the action table are initially set the same for every character, regardless of scale or topology. Together, they establish a general approach to facial animation that uses a fixed set of values normalized within the range -1 to 1. These values are mapped onto specific characters by positioning and scaling control volumes and locators. The control volumes define a reference frame for interpreting the positioning of controls, while the locators define a means for the deformers to interpret the output from the action table.

2.1 Controls and Control Volumes

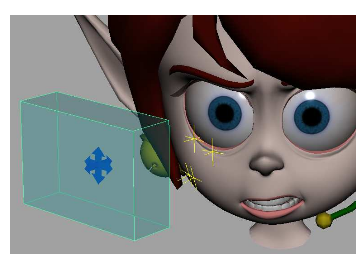


Figure 3: *Control inside normally invisible Control Volume*

Controls are widgets riding on top of the face, as shown in Figure 1(a). Users manipulate them to alter facial expressions. As shown in Figure 3, each control is linked with a control volume, which is an immovable, invisible region as-

*dave.komorwoski@disneyanimation.com
†m_vinod7@hotmail.com
‡darren@darrenmortillaro.com
§gene.s.lee@disneyanimation.com

sociated with a select area of the face. Each volume is parametrically defined and clamped in every dimension from -1 to 1. The position of a control is referenced by its parametric location inside the volume, not by its location in world space. The control volumes map the positions of the controls to a fixed range. Hence, it is these parametric values that are provided to PSD and ultimately stored as animation keys.

2.2 PSD

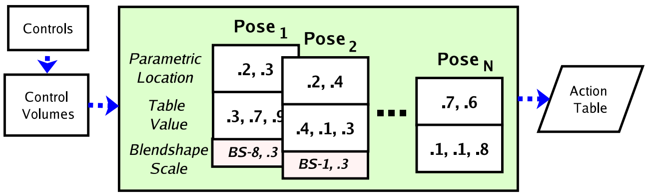


Figure 4: *PSD with "N" poses*

PSD operates by storing multiple poses and interpolating between the poses according to the positioning of controls. As shown in Figure 4, each pose of PSD identifies a pairing of a parametric location of a control to a set of values for the action table. Many of these values indirectly influence the use of deformers while others apply directly to the scaling of blendshapes. The blendshapes introduce refinements to the face that are either unattainable by the deformers alone or are unsatisfactorily configured by the action table. As the controls move, the action table is updated, and both deformers and blendshapes modify the face.

The interpolation of every pose is performed collectively. This results in the simultaneous interpolation of values for both deformers and blendshapes. Each modifies the face independently and the interpolation of one is ultimately layered upon the interpolation of the other.

2.3 Action Table and Locators

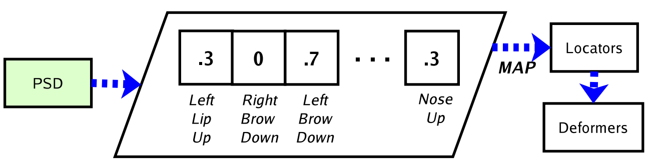


Figure 5: *Action Table for Expression in Figure 1(a)*

Loosely based on FACS [Ekman and Frisen 1978], the action table is a breakdown of the face into elementary facial actions. As shown in Figure 5, each action is assigned a single value between zero and one, with one specifying maximum influence. A combination of these values elicits a particular facial expression.

Except for those affecting blendshapes, the values of the action table influence the positioning of *locators*, invisible handles riding on top of a parameterized skull, as shown in Figure 1(b). The skull, which approximates the shape of the face, is also normally hidden. Maps that translate table values into parametric locations determine the positioning of the locators on the skull. Ultimately, the positioning of the locators determines how the deformers modify the face. Another set of maps determines how the deformers interpret the locators. The action table is the same for every rig, while the maps for each are subtle variants of a commonly defined map.

3 Benefits

Direct benefits of our hybrid approach are several. First, it supports the creation of a generic rig-building system for facial animation. Riggers start by building a generic skull along with a general action table, an appropriate set of control volumes, and an efficient map between the action table and the locators. Then, they morph the generic skull to match the skulls of individual characters. They apply similar morphs to the control volumes. These steps do not involve making any changes to either PSD or the action table. Finally, riggers tweak the maps to customize the results.

Second, our approach allows for the rapid retargeting of animation from one rig to another. As shown in Figure 1(c), a shocked stare in one character is interpreted equally as a shocked stare in the next. The retargeting of animation occurs easily because every rig shares a common, fixed PSD and because the positioning of controls is defined parametrically. The positions of the controls on one rig transfer easily onto the controls of another rig if the corresponding control volumes are sized and placed appropriately.

Third, our approach enables users to carefully tweak the interpolation of both deformers and blendshapes. PSD supports various means to interpolate shapes and numerics. Some interpolations oscillate a great deal while others interpolate smoothly and clamp at boundaries. By tweaking the interpolation, users can subtly change the way that deformations occur and how blendshapes apply.

4 Production Results

The animated productions, "Bolt" and "Prep and Landing," applied our hybrid approach with much success. In "Bolt," it took three months for animators and riggers to carefully configure a rig-building system using our approach. Afterward, it took only days to build one fully functional rig after another. There was only moderate tweaking for primary characters and practically none at all for secondary characters. Most changes involved only the re-sculpting of blendshapes to match the features of individual characters.

In "Prep and Landing," the riggers reused the rig-building system of "Bolt," easily adapting it with only minor modifications. Most of the changes were limited to the maps applied by the deformers to interpret the positions of the locators. The biggest gain to the production was the continual retargeting of animation across many characters.

References

Bergeron, P., and Lachapelle, P. 1985. Controlling facial expressions and body movements in the computer-generated animated short 'Tony De Peltrie'. In *SIGGRAPH '85: ACM SIGGRAPH 1985 Tutorial Notes*, ACM, New York.

Ekman, P., and Frisen, W. 1978. *Manual for the Facial Action Coding System*. Consulting Pyschologists Press, Palo Alto, CA.

Lee, G. S., and Hanner, F. 2009. Practical experiences with pose space deformation. In *SIGGRAPH ASIA '09: ACM SIGGRAPH ASIA 2009 Sketches*, ACM, New York, NY, USA, 1–1.

Lewis, J. P., Cordner, M., and Fong, N. 2000. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 165–172.

Magnenat-Thalmann, N., Primeau, E., and Thalmann, D. 1988. Abstract muscle action procedures for human face animation. *Visual Computer 3*, 5, 290–7. MIRALab, HEC/IRO, Montreal Univ., Que., Canada.