Eurographics/ ACM SIGGRAPH Symposium on Computer Animation (2011)
A. Bargteil and M. van de Panne (Editors)

Content Retargeting Using Parameter-Parallel Facial Layers

Natasha Kholgade[1], Iain Matthews[2,1], and Yaser Sheikh[1]

[1]Carnegie Mellon University, Pittsburgh, PA, United States, [2]Disney Research, Pittsburgh, PA, United States

Figure 1: *We retarget an actor's expressions to characters with dissimilar facial structure. Using a layered composition model of expressions, we first deconstruct the content of the actor's facial expression into emotion, speech, and blink layers. We transfer parameters of each layer to parallel layers for the character. Finally, we construct the character's facial expression by composing the content of the emotion, speech, and blink layers. By definition, the retargeting parameter values are identical for all characters, but each character may have its own unique parametrization.*

Abstract

Facial motion retargeting approaches often transfer expressions by establishing correspondences between shared units of motion, such as action units, or spatial correspondences of landmarks between the source actor and target character faces. When the actor and character are structurally dissimilar, shared units of motion or spatial landmarks may not exist, and subtle styles of performance may differ. We present a method to deconstruct the content of an actor's facial expression into three parameter-parallel layers using a composition function, transfer the content to equivalent parameter-parallel layers for the character, and reconstruct the character's expression using the same composition function. Our algorithm uses the same parameter-parallel layered model of facial expression for both the actor and character, separating the content of facial expressions into emotion, speech, and eye-blink layers. Facial motion in each layer is embedded in simplicial bases, each of which encodes semantically significant configurations of the face. We show the transfer of facial motion capture and video-based tracking of the eyes and mouth of an actor to a number of faces with dissimilar facial structure and expressive disposition.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

1. Introduction

Computer graphics characters have been used to augment or embody the protagonist's performance in several recent theatrical releases including *Tron*, *Avatar*, and *The Curious Case of Benjamin Button*. In each of these movies, facial motion retargeting — the process of transferring performance from a source face to a target face — was used to allow actors to realistically control the timing and con-

SCA 2011, Vancouver, British Columbia, Canada, August 5 – 7, 2011.
© 2011 ACM 978-1-4503-0923-3/11/0008 $10.00

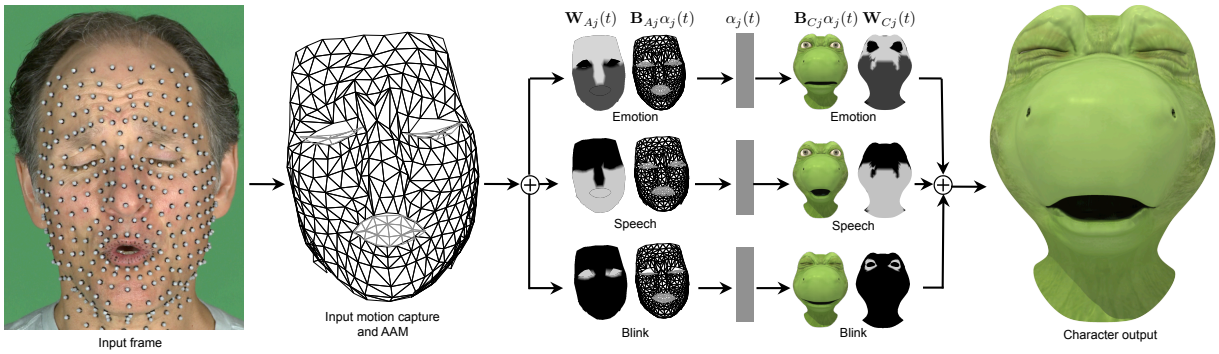

Figure 2: *Parameter-parallel layered model: we use motion capture data and AAM tracks from the actor as input to the system. The actor performance is deconstructed into emotion, speech, and blink layer outputs, and weighted actor masks. Using basis coefficients and weights of each layer, we reconstruct the layer outputs and weighted influence masks for the character in parallel, and compose them in vertex space to yield the character output.*

tent of the characters' expressions with their own. With a few exceptions, current approaches to facial retargeting deconstruct and transfer facial motion over shared elementary units of motion such as muscle activations or action units [PW96, CLK01, PL06] or transfer facial deformations based on explicit spatial correspondences between the faces [LWP10, WLGP09, MJC*08, NJ04]. Retargeting facial expressions by transferring local motion fields is difficult when the source and target faces are dissimilar in their facial structure (e.g., retargeting to a character with an oddly shaped nose or two mouths) because the source face may not have the necessary elementary units of motion or explicit spatial correspondences needed to drive the motion of the target. Instead, compelling target animations that preserve the intent of the source expressions can be created by recognizing and retargeting the *content* of the source performance rather than transferring the *motion*. By retargeting content, we let animators define the target face's expressions (how do crocodiles *smile*? [Gle98]), and give creative control of timing and content to actors.

We define the content of facial motion in terms of three underlying processes: emotion, speech, and eye-blinks. These processes describe the degree of various emotional expressions in the actor or character performance, the types of visemes (stylized mouth patterns corresponding to speech production such as 'ah,' 'oo,' and 'wa'), and the blink patterns present in the performance. Each process forms a layer in a compositional form, which also incorporates rigid head motion. We express the content of each layer using a simplicial basis, where the simplex vertices correspond to semantic extremes for each process. The influence of each layer in producing the final character expression is modulated over time using weighted masks. This allows us to seamlessly produce the facial expression in cases where more than one layer influences a region of the face. For instance, blinking occurs involuntarily to irrigate the eyes, but it can also occur due to emotions such as grief or submission. The weights allow us to emphasize or de-emphasize involuntary blinking as opposed to eye-closure due to emotion, at the time instances when they occur, in the final composition.

We use the same composition function for facial expressions of both the actor and the character. As input, we obtain active appearance model (AAM) [MB04] points for the eyes and lips from actor video, and we use motion capture markers for the rest of the actor's face. As shown in Figure 2, we extract model parameters from the actor by deconstructing the actor's expression into the three layers and actor masks of the composition function; we then transfer the same parameters to parallel character layers; finally, we reconstruct the 3D character output using the same composition function. The simplices of the layers form parameter-parallel retargeting spaces induced on top of the input actor and output character spaces. The measurement and representation of the actor and parameterization of the character is independent of the induced simplices and retargeting method. The motivation behind using such a parameter-parallel approach is that it does not require explicit spatial correspondence between the actor and character spaces. Instead, correspondences are provided in terms of content that carries semantic significance to artists and actors. Artists design characters that often possess facial features which do not correspond to an actor's face, and therefore providing explicit spatial correspondences may not be intuitive. However, artists are able to expertly pose semantically significant facial configurations such as emotions and speech patterns on characters. The parameter-parallel approach leverages the talent of animators to pose believable emotions and speech on dissimilar characters.

We illustrate results of retargeting to characters that have structurally different facial configurations from the actor. Our characters include a human with a familiar facial configuration, a tortoise with non-human-like facial parts, a globe

with facial parts whose numbers and arrangements are unusual, and a cassette player, with non-facial components. We show results for vignettes covering a broad spectrum of emotions, and demonstrate that the model effectively modulates the time-varying contributions of emotions, blinks, and speech.

2. Related work

There are two broad approaches for performing retargeting of face and body motion: mesh deformation and parameter transfer. In mesh deformation techniques, deformations of the geometry of a source mesh with respect to a base mesh are computed and the deformation is spatially mapped to deform a target base mesh through initial vertex correspondences. Sumner and Popovic [SP04] map deformation transformations from a source mesh to a target mesh while ensuring that adjacent triangles in the target mesh continue sharing vertices when deformed. Weise et al. [WLGP09] smoothly deform a generic model to conform to an actor's face, and track it offline over time with optical flow, border, and mesh constraints. They use a PCA model to improve online tracking performance, and apply deformation transfer to retarget the actor's face to a similar character face for live puppetry. Techniques for using 3D scans of the actor include polynomial displacement maps (PDMs) [MJC*08] and mesh refinement by hierarchical subdivision [NJ04]. Bickel et al. [BLB*08] perform large-scale deformation of high-resolution meshes with a few manual correspondences, and represent finer details through a pose space that is based on a strain-vector representation.

In parameter-transfer techniques, the span of facial motions is expressed in terms of parameters of a model or animation rig. These parameters are then transferred from the source space to the target space. A popular approach for doing parameter transfer is to use predetermined actions units or muscles [PW96, EFH02, PL06, Osi07]. Another approach is to use a linear basis. The vectors of the basis may be obtained automatically or selected manually from actor performances. In these approaches, basis weights may be unconstrained or may include constraints such as non-negativity bounds, sum-to-one, or combinations of the previous three [CB02, BLCD02, JTDP03, CLK01]. Joshi et al. [JTDP03] perform scattered data interpolation using RBF kernels to interpolate dense spatial data for expressions given sparse tracked data and dense neutral frame data. To develop the linear bases for the source, Chuang et al. [CB02] extract data points with minimum and maximum projections on the principal eigenvectors, Bregler et al. [BLCD02] manually pick key shapes, and enlarge their basis set with non-linear interpolation, and Choe et al. [CLK01] use skin deformations in response to an underlying muscle model whose weights are muscle actuation parameters. Other bases include motion captured action unit data [CBK*06], and non-rigid 3D models of the face [ZWT09]. Linear bases for the character typically consist of 3D character meshes, but retargeting is more general, and can be done to 2D cartoon data [BLCD02], 3D scans [CBK*06], and from facial geometry to texture information [JTDP03, ZLT*06]. Baran et al. [BVGP09] express source and target meshes in patch-based linear rotation invariant coordinates to retarget to targets whose motions are visually different but semantically similar. Buck et al. [BFJ*00] deconstruct the face into piecewise linear models of motions of the eye, mouth, and face to retarget the motion of an actor from video to facial expressions in sketches.

To separate facial expressions and speech, [DCFN06] use PCA coefficients of mouth points to represent phonemes. They express frames between the two or three phoneme samples of a diphone or triphone as weights of the PCA coefficients, and learn polynomial fitting functions from multiple training instances. They subtract neutral facial motion from expressive motion for the same sentence to get a phoneme-independent space, and reduce its dimensionality by PCA. During synthesis, they use a set of 13 key-shapes to generate diphone or triphone coarticulations by a greedy algorithm, and a patch-based sampling method to synthesize expressive motion from the phoneme-independent eigenspace. Vlasic et al. [VBPP05] use bilinear and trilinear tensor models to separate identity, expressions, and speech visemes. They use subspace decomposition to obtain the tensor whose each mode represents one feature, and determine weights that multiply the tensor to generate the data point. Chuang and Bregler [CB05] use a bilinear model to express the interaction of expression (style) variables with speech (content) variables. These models are multiplicative and like most multilinear models require a number of cross-model observations to ensure tractability (although Vlasic et al. [VBPP05] propose strategies to mitigate this problem). Cao et al. [CFP03] extract independent components from data, separate them into an emotion component and a speech + emotional component. They reconstruct speech independent of expressiveness from the mouth region for speech motion editing. Pyun et al. [PKC*03] compute weights of radial and linear basis functions using source key-models for a source expression, and map the weights to target key-models. They too have an importance mask to weight the contribution of speech and emotion, but they compute their weights in a separate pre-processing step, while mask weights and coefficients in our system are compute simultaneously within a single optimization framework.

Our work lies in the realm of parameter transfer techniques, but whereas other methods encode and transfer motion, we recover and transfer the content of facial performance from the actor to the character. We use a layered model to capture subtleties of emotions, speech, and eye-blinks, and we estimate time varying weights to modulate the parameters. The weights define the contributions of emotion, speech, and blink layers to the final facial output at different points on the face. They enable us to represent a wide range of facial motions with a concise basis of semantically

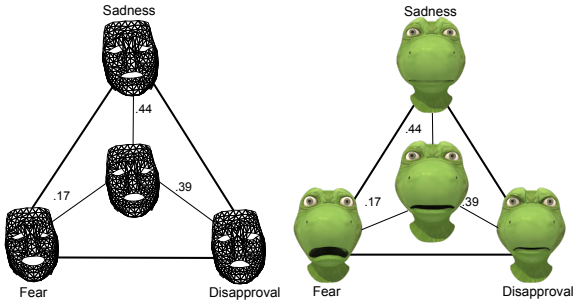

Figure 3: *Simplicial basis for the emotion layer. The layer output within the simplex is represented by barycentric coordinates of 29 simplex extremes (for instance, extremes of the sadness, fear, and disapproval emotions). The actor simplex has hand-picked emotion expressions, and the character simplex has emotional poses designed by the artist.*

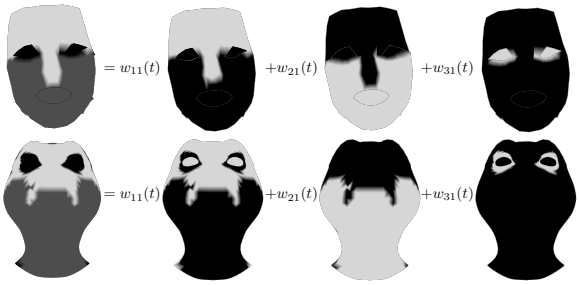

Figure 4: *Masks for emotion layer. The emotion mask is a weighted sum of three manually specified masks for the forehead, mouth, and eye regions. The masks mark out regions that move similarly in the actor and the character.*

meaningful facial poses (emotions defined by Plutchik, commonly used visemes, and blink patterns). For each layer in the model, we induce a parameter-parallel retargeting space in terms of semantically significant simplices of emotion, speech, and eye-blinks. The layered model differs from prior multilinear models that assume cross-talk between emotion and speech components, i.e., they interact multiplicatively. We additively composite outputs of emotion, speech, and blink layers using weighted masks, and there is no cross-talk between emotion and speech. We describe the contribution of emotion and speech through their weighted influence on the final output.

3. Layered Model

The input from the actor is P_1 3D motion capture points from the face, and P_2 active appearance model points (AAM) [MB04] tracked from the eyes and the lips. These points are arranged in a vector $\mathbf{x}_A(t) \in \mathbb{R}^{D_A}$, where $D_A = 3P_1 + 2P_2$. We represent the facial expression of the actor in terms of head motion, and three layers, namely, emotion, speech, and eye blinks. We indicate emotion, speech, and eye-blinks by subscripts 1, 2, and 3 respectively throughout the text. At any time instant t, these layers form the content of the actor's performance in the following additive compositional form,

$$\mathbf{x}_A(t) = \tilde{\mathbf{R}}(t)\left(\mathbf{x}_{\mu A} + \sum_{j=1}^{3} \mathbf{W}_{Aj}(t)\mathbf{B}_{Aj}\alpha_j(t)\right) + \tilde{\mathbf{t}}(t). \quad (1)$$

$\tilde{\mathbf{R}}(t)$ and $\tilde{\mathbf{t}}(t)$ represent the rotation and translation parameters of head motion, and $\mathbf{x}_{\mu A}$ is the actor mean. $\mathbf{B}_{Aj}, j \in \{1,2,3\}$ are simplices bases corresponding to emotion, speech, and blink processes respectively. $\alpha_j(t)$ are simplex coefficients, and $\mathbf{W}_{Aj}(t)$ are matrices with weights modulating the influence of layers over time.

To ensure parameter-parallel transfer of content, we have an identical compositional form for the character,

$$\mathbf{x}_C(t) = \tilde{\mathbf{R}}_{3D}(t)\left(\mathbf{x}_{\mu C} + \sum_{j=1}^{3} \mathbf{W}_{Cj}(t)\mathbf{B}_{Cj}\alpha_j(t)\right) + \tilde{\mathbf{t}}_{3D}(t). \quad (2)$$

The character mesh is in 3D, and $\mathbf{x}_C \in \mathbb{R}^{D_C}$ where $D_C = 3P$, P is the number of character vertices; $\mathbf{x}_{\mu C}$ is the character mean, $\tilde{\mathbf{R}}_{3D}(t)$ and $\tilde{\mathbf{t}}_{3D}(t)$ are the 3D components of rotation and translation, $\mathbf{B}_{Cj}$ are simplicial bases for the character, $\mathbf{W}_{Cj}(t)$ are matrices with weights of influence and $\alpha_j(t)$ are simplex coefficients. Note that Equations 1 and 2 are driven by the same $\alpha_j(t)$ which are components of the three layer processes. Sections 3.1 and 3.2 describe the bases, coefficients, and weights in depth. Figure 2 depicts the interaction of the weights of influence with the layer outputs for the actor and in parallel for the tortoise character Oliver.

3.1. Simplicial Basis and Coefficients

We represent the data for each layer using emotion, speech, and blink simplicial bases for the actor ($\mathbf{B}_{A1} \in \mathbb{R}^{D_A \times K_1}$, $\mathbf{B}_{A2} \in \mathbb{R}^{D_A \times K_2}$, and $\mathbf{B}_{A3} \in \mathbb{R}^{D_A \times K_3}$) and the character ($\mathbf{B}_{C1} \in \mathbb{R}^{D_C \times K_1}$, $\mathbf{B}_{C2} \in \mathbb{R}^{D_C \times K_2}$, and $\mathbf{B}_{C3} \in \mathbb{R}^{D_C \times K_3}$). The numbers of extremes in the emotion, speech, and blink simplices are K_1, K_2, and K_3 respectively. The components of each basis form a simplex[†]. We define that the data for each layer is contained within the simplex as shown in Figure 3, and we express the data in terms of the simplex extremes using nonnegative barycentric coordinates that sum to one. Barycentric coordinates $\alpha_1(t)$, $\alpha_2(t)$, and $\alpha_3(t)$ of the emotion, speech, and blink simplices form the set of simplex coefficients that are common to the actor and the character.

[†] A simplex in $\mathbb{R}^D$ is the simplest possible polytope in that space, and has $D+1$ vertices. A triangle is a simplex in $\mathbb{R}^2$, a tetrahedron in $\mathbb{R}^3$, etc. If a simplex is centered at one of its vertices, the remaining vertices are linearly independent.

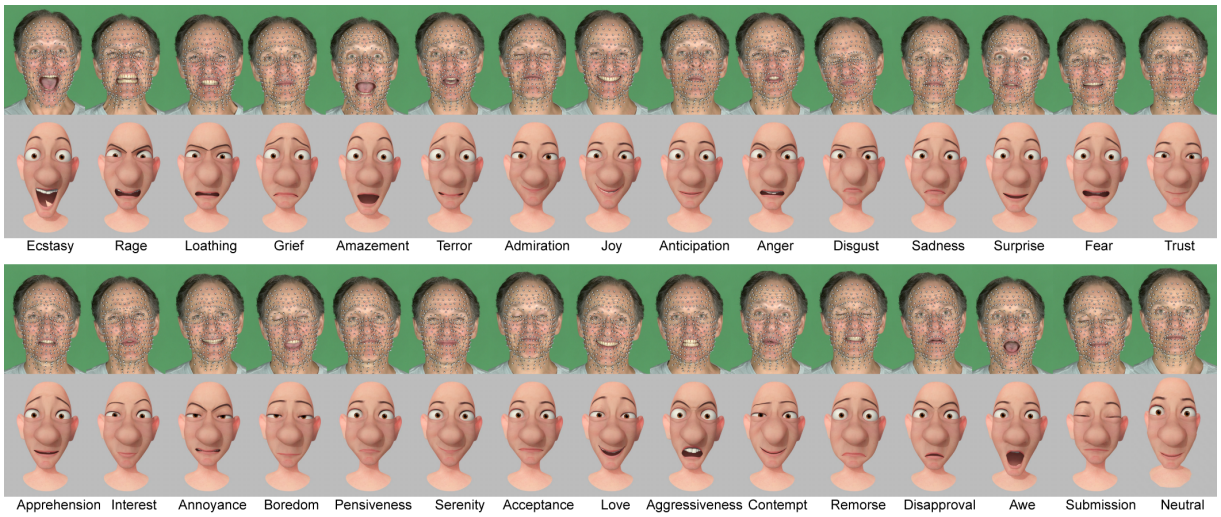

Figure 5: *Plutchik [Plu80] inspired 29 emotions with the neutral face for the actor and the character Cappellini.*

For the j-th simplex,

$$\sum_{k=1}^{K_j} \alpha_{jk}(t) = 1, \alpha_{jk}(t) \geq 0 \; \forall j,k.$$

The products $\mathbf{B}_{Aj}\alpha_j(t)$ and $\mathbf{B}_{Cj}\alpha_j(t)$ (shown in Figure 2) are outputs of the j-th layer for the actor and the character. The layer outputs for the actor and character emotion simplices ($\mathbf{B}_{A1}$ and $\mathbf{B}_{C1}$) are shown in Figure 3. The non-negativity and summation-to-one constraints of each simplex provide a bound on the L_1-norm of the simplex coefficients. These constraints induce sparsity as opposed to the more common purely nonnegative constraints. Bound constraints of nonnegativity alone can turn on too many coefficients to reconstruct the input with low error, and combine several emotions that do not plausibly occur together. The sparse constraints maintain perceptual plausibility by mixing a small subset of emotions and the sum-to-one bound keeps the motions within the span of the extremes.

The emotion simplex consists of $K_1 = 29$ extremes of emotion from a set of 32 emotions characterized by Plutchik [Plu80]. There are 8 primary emotions (joy, anger, sadness, surprise, disgust, fear, trust, and anticipation), each of which has three degrees. An additional 8 emotions lie at the junctures of the primary emotions. Figure 5 shows these emotions for the actor and the character Cappellini. The speech simplex is composed of $K_2 = 12$ commonly used viseme extremes, and the blink simplex consists of extremes of closed eyes, open eyes and partially open eyes, $K_3 = 3$. We handpick the frames for the actor simplices from actor performance, and the poses for the character simplices are created by an artist. Under the parameter-parallel approach, we can directly transfer simplex coefficients from the actor layers to the character layers.

3.2. Weights and Masks of Influence

The matrices $\mathbf{W}_{A1}(t)$, $\mathbf{W}_{A2}(t)$, and $\mathbf{W}_{A3}(t)$ are $D_A \times D_A$ diagonal matrices[‡] that specify the influence of the emotion, speech, and blink layers $\mathbf{B}_{Aj}\alpha_j(t)$ to each vertex of the actor's face. The i-th row of each $\mathbf{W}_{Aj}(t)$ weights the importance of the i-th element of each layer output $\mathbf{B}_{Aj}\alpha_j(t)$ in generating the i-th element $x_i(t)$ of $\mathbf{x}_A(t)$. For instance if $x_i(t)$ is a forehead point, at time instant t, its motion is dominated by the emotion component $\mathbf{B}_{A1}\alpha_1(t)$, and the diagonal of $\mathbf{W}_{A1}(t)$ has a high value at the i-th location; diagonals of $\mathbf{W}_{A2}(t)$ and $\mathbf{W}_{A3}(t)$ have low values. Mouth points get their contributions from both emotion and speech leading to high values at their locations in diagonals of $\mathbf{W}_{A1}(t)$ and $\mathbf{W}_{A2}(t)$. To conserve energy, we constrain the diagonals of the $\mathbf{W}_{Aj}(t)$'s to be nonnegative and sum to 1 across j.

$$\sum_{j=1}^{3} \mathbf{W}_{Aj}(t) = \mathbf{I}, \mathbf{W}_{Aj}(t) \geq 0.$$

The weight matrices $\mathbf{W}_{C1}(t)$ to $\mathbf{W}_{C3}(t)$ similarly account for the influence of the three layers to the character output, and

$$\sum_{j=1}^{3} \mathbf{W}_{Cj}(t) = \mathbf{I}, \mathbf{W}_{Cj}(t) \geq 0.$$

We cannot transfer the weights from the actor to the character directly because $D_A \neq D_C$, and $\mathbf{W}_{Aj}(t) \neq \mathbf{W}_{Cj}(t)$. To address this issue, we introduce a structure on the diagonals of the weight matrices using masks. The diagonal of the j-th actor weight matrix $\mathbf{W}_{Aj}(t)$ can take on one of m values, $w_{j1}(t)$, $w_{j2}(t)$, ... $w_{jm}(t)$ ($m = 3$), and m masks mark

[‡] We make $\mathbf{W}_{Aj}(t)$ diagonal to apply the i-th diagonal entry of $\mathbf{W}_{Aj}(t)$ to all elements of the i-th row of $\mathbf{B}_{Aj}\alpha_j(t)$.

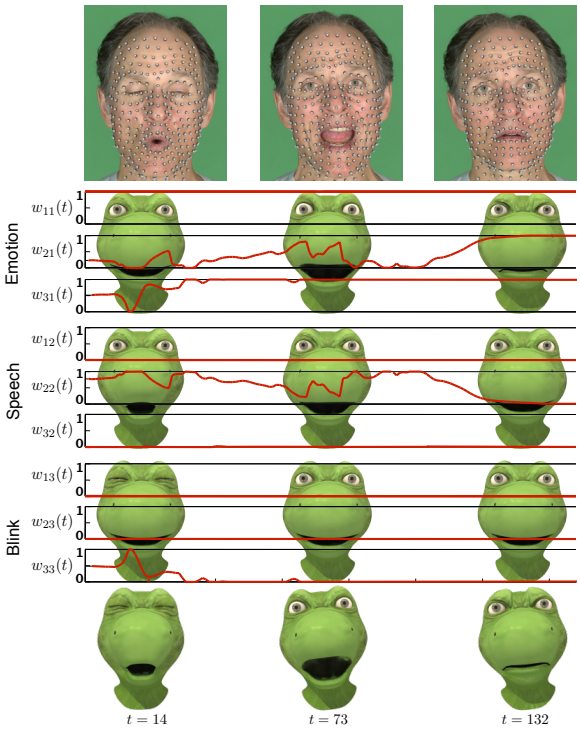

Figure 6: *Timeline of variation of mask weights. The rows show actor input, weighted outputs of the emotion, speech, and blink layers, and character output for frames 14, 73, and 132.*

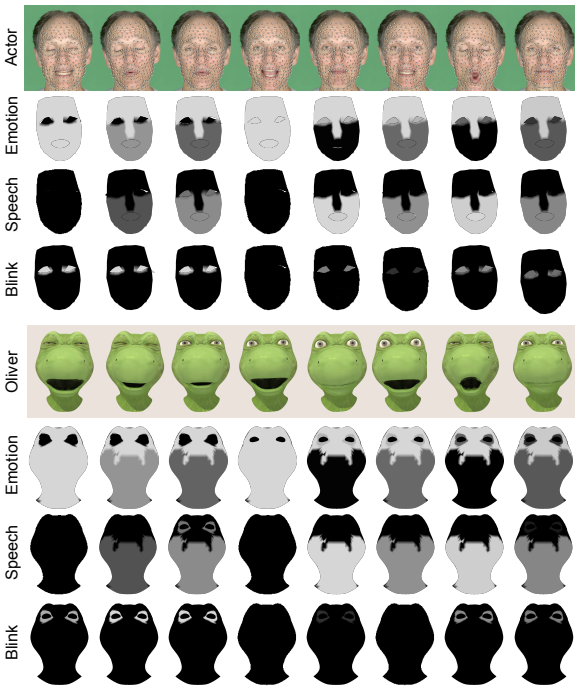

Figure 7: *Time-varying masks weighted masks for the actor and the character Oliver for a 'guilty' sequence.*

out points on the actor that will take on each value. The l-th mask $\mathbf{M}_{Al}$ is a $D_A \times D_A$ matrix with ones on the diagonal for points at which $\mathbf{W}_{Aj}(t)$ has the value $w_{jl}(t)$. Similarly, for the character, the diagonal of the l-th mask $\mathbf{M}_{Cl} \in \mathbb{R}^{D_C \times D_C}$ has ones where $\mathbf{W}_{Cl}(t)$ takes the value $w_{jl}(t)$.

We manually specify the l-th mask for the actor and character $\mathbf{M}_{Al}$ and $\mathbf{M}_{Cl}$, such that points marked out by the l-th mask have shared influences from a particular layer, and they move similarly in the actor and the character. We have an upper face mask, a lower face mask, and an eyes mask for the actor, Oliver, and Cappellini. For Monstergea, a four-mouthed globe, the masks correspond to the body (devoid of the eyes and mouth), the mouths, and the eyes. For Radiohead, they correspond to the radio chassis, the cassette compartment, and the speakers. Using the mask, we can write the j-th weight matrix for the actor and the character as:

$$\mathbf{W}_{Aj}(t) = \sum_{l=1}^{m} w_{jl}(t)\mathbf{M}_{Al}, \; \mathbf{W}_{Cj}(t) = \sum_{i=1}^{m} w_{jl}(t)\mathbf{M}_{Cl}, \quad (3)$$

$$\sum_{j=1}^{3} w_{jl}(t) = 1, w_{jl}(t) \geq 0 \; \forall j,l.$$

The weight matrices can now be viewed as weighted masks.

Figure 4 shows these weighted masks and their relationship with the manually specified masks for the actor and for Oliver. The figure is a diagrammatic representation of Equation 3. Figure 2 shows the action of the weighted masks on the layer outputs $\mathbf{B}_{Aj}\alpha_j(t)$ for the actor and $\mathbf{B}_{Cj}\alpha_j(t)$ for Oliver.

We can substitute Equation 3 in Equations 1 and 2 to get the following parameter-parallel forms:

$$\mathbf{x}_A(t) = \tilde{\mathbf{R}}(t)\left(\mathbf{x}_{\mu A} + \sum_{j=1}^{3}\sum_{l=1}^{m} w_{jl}(t)\mathbf{M}_{Al}\mathbf{B}_{Aj}\alpha_j(t)\right) + \tilde{\mathbf{t}}(t), \quad (4)$$

$$\mathbf{x}_C(t) = \tilde{\mathbf{R}}_{3D}(t)\left(\mathbf{x}_{\mu C} + \sum_{j=1}^{3}\sum_{l=1}^{m} w_{jl}(t)\mathbf{M}_{Cl}\mathbf{B}_{Cj}\alpha_j(t)\right) + \tilde{\mathbf{t}}_{3D}(t). \quad (5)$$

The weights $w_{jl}(t)$ and the coefficients $\alpha_j(t)$ at each time step are now common for the actor and the character, and they can be directly retargeted to the character. We discuss the extraction of parameters $\tilde{\mathbf{R}}(t)$, $\tilde{\mathbf{t}}(t)$, $w_{jl}(t)$, and $\alpha_j(t)$ from the actor's data in Section 4, and their retargeting to the character in Section 5. For notational convenience, we also use the matrices $\mathbf{W}(t)$ and $\alpha(t)$ represented as:

$$\mathbf{W}(t) = \begin{bmatrix} w_{11}(t) & w_{21}(t) & w_{31}(t) \\ w_{12}(t) & w_{22}(t) & w_{32}(t) \\ \vdots & \vdots & \vdots \\ w_{1m}(t) & w_{2m}(t) & w_{3m}(t) \end{bmatrix}, \; \alpha(t) = \begin{bmatrix} \alpha_1(t) \\ \alpha_2(t) \\ \alpha_3(t) \end{bmatrix}$$

In $\mathbf{W}(t)$, $w_{jl}(t)$ represents the weight by which the j-th layer (emotion, speech, or eye-blinks) influences points demarcated by the l-th mask. Figure 6 provides an example of the weights for an angry sequence applied to the tortoise Oliver. Points in the forehead (masked out by $\mathbf{M}_{C1}$) have almost complete influence from emotion (i.e., $w_{11}(t) = 1$, $w_{12}(t) = 0$, and $w_{13}(t) = 0$) throughout the sequence. Mouth points (masked out by $\mathbf{M}_{C2}$) have mixed contributions from emotion and speech, and almost no influence from blinking. In particular, when the actor produces a pronounced 'wa', 'o', or 'mm' sound, the mouth weight for the speech layer, $w_{22}(t)$ becomes high. In a state of strong emotion and minimal speech, the mouth weight for the emotion layer, $w_{21}(t)$, spikes up. Eyelid points ($\mathbf{M}_{C3}$) show mixed influence from the emotion and blink layers. Figure 7 shows the time-varying weighted masks generated for a guilty sequence.

4. Extraction of Layered Model Parameters

We use actor motion capture and tracks from active appearance models (AAMs) [MB04] from an actor's performance as our input data. We use 283 motion capture markers to capture facial motion from the forehead, nose, cheeks, upper jaw and lower jaw, and we track eye-blinks (18 points) and lip motion (22 points) using AAMs. We separately align the P_{3D} 3D motion capture and P_2 2D AAM tracks of the actor input $\mathbf{x}_A$ to those of the mean face, $\mathbf{x}_{\mu A}$ using Procrustes analysis. In Equation 4,

$$\tilde{\mathbf{R}}(t) = \begin{bmatrix} \mathbf{R}_{3D}(t) \otimes \mathbf{I}_{P_{3D}} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{2D}(t) \otimes \mathbf{I}_{P_{2D}} \end{bmatrix},$$

$$\tilde{\mathbf{t}}(t) = \begin{bmatrix} \mathbf{t}_{3D}(t) \otimes \mathbf{I}_{P_{3D}} \\ \mathbf{t}_{2D}(t) \otimes \mathbf{I}_{P_{2D}} \end{bmatrix},$$

where $\mathbf{R}_{3D}(t)$ and $\mathbf{R}_{2D}(t)$ are 3D and 2D rotation matrices obtained from aligning motion capture and AAM tracks respectively, and $\mathbf{t}_{3D}(t)$ and $\mathbf{t}_{2D}(t)$ are corresponding translations. $\tilde{\mathbf{R}}(t)$ and $\tilde{\mathbf{t}}(t)$ are parameters of head motion.

After extracting the head motion parameters $\tilde{\mathbf{R}}(t)$ and $\tilde{\mathbf{t}}(t)$, we remove their effect, and the mean shape, from the actor data:

$$\hat{\mathbf{x}}_A(t) = \tilde{\mathbf{R}}(t)^{-1} \left(\mathbf{x}_A(t) - \tilde{\mathbf{t}}(t) \right) - \mathbf{x}_{\mu A}.$$

We now need to extract the parameters $\mathbf{W}(t)$ and $\alpha(t)$ from $\hat{\mathbf{x}}_A$. This involves performing the following optimization:

$$\begin{aligned} &\left(\mathbf{W}(t)^*, \alpha(t)^*\right) = \\ &\arg \min_{\mathbf{W}(t), \alpha(t)} \left\| \hat{\mathbf{x}}_A(t) - \sum_{j=1}^{3} \sum_{l=1}^{m} w_{jl}(t) \mathbf{M}_{Al} \mathbf{B}_{Aj} \alpha_j(t) \right\|^2 \quad (6) \\ &\text{s.t.} \sum_{j=1}^{3} w_{jl}(t) = 1,\ w_{jl}(t) \geq 0\ \forall j, l, \\ &\sum_{k=1}^{K_j} \alpha_{jk}(t) = 1,\ \alpha_{jk}(t) \geq 0\ \forall j, k. \end{aligned}$$

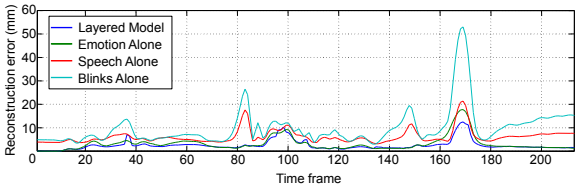

Figure 8: *Comparison of root mean square reconstruction error for the layered model versus using just the emotion, speech, and blink simplex bases. The layered model accurately represents facial expression.*

This optimization is bilinear in $\mathbf{W}(t)$ and $\alpha(t)$. For each frame, we obtain a local minimum using the interior-point followed by sequential quadratic programming algorithms for constrained minima. We use the parameters at time $t-1$ to initialize the optimization for the frame at time t. The coefficients at the first frame are initialized randomly. At each stage, convergence is obtained when the change in function tolerance falls to below 10^{-6}. Figure 8 shows the root mean square reconstruction errors for a single sequence of the actor using the layered model as compared with using just the emotion, speech, and blink simplices to do reconstruction. The layered model has a lower reconstruction error for majority of the sequence, using emotion alone does slightly worse, followed by speech, and finally blinks have a very high reconstruction error.

We introduce sensitivity to lip closing (/p/, /b/, or /m/ sounds) as lip closing is perceptually important for convincing animation. We bias the weights and coefficients for the mouth mask toward the /m/ viseme ($w_{21}(t) = 0$, $w_{22}(t) = 1$, $w_{23}(t) = 0$, and $\alpha_2(t) = \alpha_{mm}$) using factor $\gamma = e^{-\frac{d^2}{2\sigma^2}}$, where d is the distance between the upper and lower lips of the actor AAM. We also bias the weights and coefficients in frame t towards those in frame $t-1$ to maintain temporal smoothness. The minimization is augmented to:

$$\begin{aligned} &\left(\mathbf{W}(t)^*, \alpha(t)^*\right) = \\ &\arg \min_{\mathbf{W}(t), \alpha(t)} \left\| \hat{\mathbf{x}}_A(t) - \sum_{j=1}^{3} \sum_{l=1}^{m} w_{jl}(t) \mathbf{M}_{Al} \mathbf{B}_{Aj} \alpha_j(t) \right\|^2 \quad (7) \\ &+ \frac{\gamma}{1-\gamma} \left(\left\| \begin{bmatrix} w_{21}(t) \\ w_{22}(t) \\ w_{23}(t) \end{bmatrix} - \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right\|^2 + \|\alpha_2(t) - \alpha_{mm}(t)\|^2 \right) \\ &+ \frac{\lambda}{1-\gamma} \left(\|\mathbf{W}(t) - \mathbf{W}(t-1)\|_F^2 + \|\alpha(t) - \alpha(t-1)\|^2 \right) \\ &\text{s.t.},\ \sum_{j=1}^{3} w_{jl}(t) = 1,\ w_{jl}(t) \geq 0,\ \sum_{k=1}^{K_j} \alpha_{jk}(t) = 1,\ \alpha_{jk}(t) \geq 0. \end{aligned}$$

5. Retargeting of Parameters to the Character

The parameters calculated in Section 4 are applied to our characters through Equation 5. To simplify the retargeting,

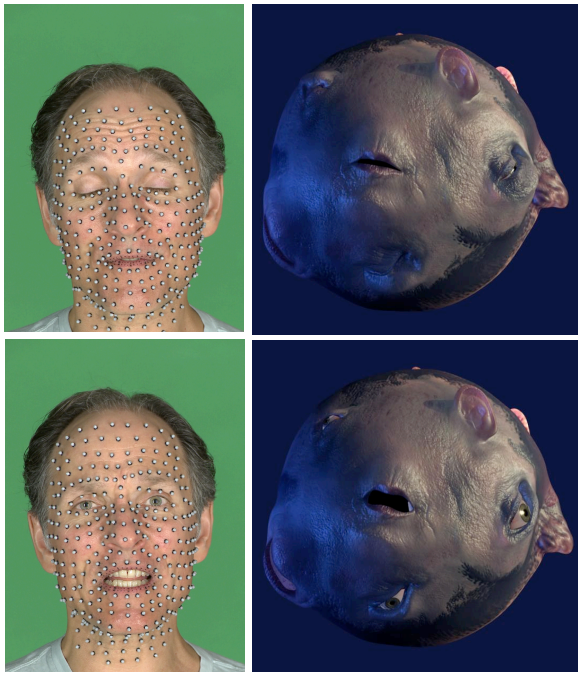

Figure 9: *Retargeting to a character with an irregular number of human-like features arbitrarily arranged over a sphere. Spatial retargeting would require a difficult to define and complex one-to-many mapping.*

we only apply the rotation and translation obtained from motion capture to the 3D character mesh. In Equation 5, $\tilde{\mathbf{R}}_{3D}(t)$ and $\tilde{\mathbf{t}}_{3D}(t)$ are given as:

$$\begin{aligned} \tilde{\mathbf{R}}_{3D}(t) &= \mathbf{R}_{3D}(t) \otimes \mathbf{I}_P, \\ \tilde{\mathbf{t}}_{3D}(t) &= \mathbf{t}_{3D}(t) \otimes \mathbf{1}_P. \end{aligned}$$

The resulting character mesh is rendered in Maya by projecting it onto a set of blendshapes created by the artist for the character.

6. Results

To illustrate our content-based retargeting approach, Figure 9 shows two frames from an emotional vignette for an imaginary character Monstergea. This character has an unusual number human like facial features arranged around the surface of a sphere. The character has five eyes, four mouths, two noses and two ears which can all be posed independently. Retargeting to this model using a spatial approach would be hard to define.

We retarget emotional sentences to four characters with dissimilar facial structures. Figure 10 shows frames from the actor retargeted to the three characters, for a 'surprised' sequence. The actor initially expresses surprise by saying 'Wow', follows up with 'I had no idea you were into that!'. The first three frames show the visemes involved in saying 'Wow'; these are followed by frames for 'I', the 'o' of 'no', the 'i' and 'e' of 'idea', the 'a' of 'that', and an ending smile. We capture the knotted eyebrows of the interested expression on the actor's face as he utters the word 'idea'. In case of Radiohead, mouth movement of the actor is mapped to the motion of the cassette compartment. Radiohead's entire chassis curves upwards when happy (last frame in Figure 10), and expands when excited (fourth frame in Figure 10). Animations of these and other sequences are shown in the video accompanying this submission. We generate 59 sentences by setting $\sigma = .02$ and $\lambda = .2$ for all characters except for Cappellini, where $\sigma = .005$ and $\lambda = .05$.

7. Discussion

We present a parameter-parallel approach to retarget the content of an actor's expression to a variety of characters with dissimilar facial structures. We transfer coefficients of simplicial bases for emotion, speech, and blink layers, and time-varying weights of influence of each layer to various regions of the face. Under the parameter-parallel approach, the resulting animations capture the expressiveness of the actor's performance in the distinctive style assigned by the artist to each character.

The layered model we describe is not a unique decomposition of facial expression. The requirement is to span the space of facial expression in a semantically meaningful way that an artist can define for retargeting. Our goal is to produce animations onto which viewers can plausibly project the content of an actor's performance. The simplex provides a sparse set of coefficients, that capture the most meaningful simplex vertices towards the emotion, speech, and blink content of a particular facial expression. By combining a narrow set of simplex vertices, the simplex helps to generate perceptually plausible emotional content in our animations.

The layered model provides expression transfer through a basis that is interpretable by both the actor and the character artist. Transferring expressions via a FACS basis can be somewhat challenging as it requires the actor to perform locally isolated motions to define the mapping. It is also unclear how mesh deformation techniques will work for characters with facial morphologies that deviate considerably from the human face. In future work, we plan to look into adapting FACS and deformation transfer for retargeting to characters like Monstergea and Radiohead so as to compare our performance against these existing techniques.

Perceptually, it is challenging to evaluate the animation quality of non-anthropomorphic characters like Radiohead and Monstergea, because we are tuned to facial expressions of more human characters. Through our algorithm, we attempt to hit the emotions provided by the artist, while maintaining convincing speech patterns.

There are three principal limitations to our approach.

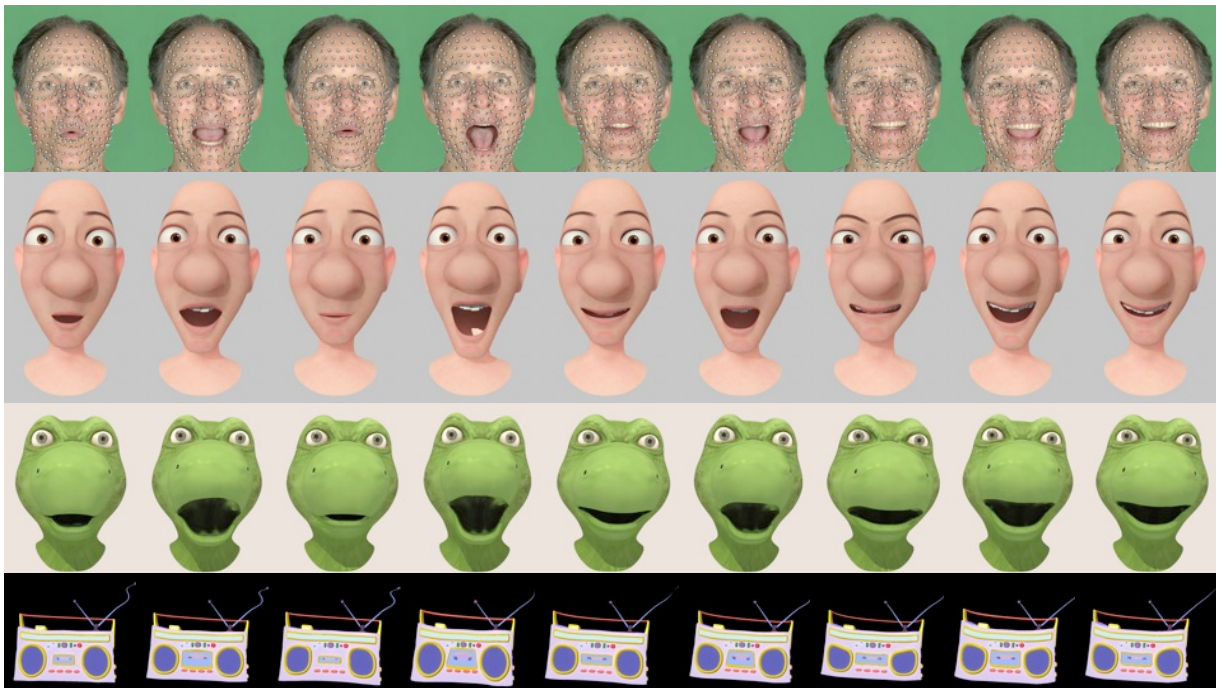

Figure 10: *Frames from the actor for a 'surprised' sequence retargeted to Cappellini, Oliver, and Radiohead.*

First, our algorithm operates offline and in batch mode. As our processes are first order Markovian (i.e., they consider only the previous frame), we expect that the model will be amenable to online design. Second, we use Procrustes alignment with respect to the L_2-norm for rigid bodies to compute rotation and translation for head motion; excessive non-rigid motion can dominate the alignment algorithm and provide an incorrect rigid estimate. Finally, the simplex structure best captures the motion on its boundary and within its interior. A limitation of the simplex is that motions outside the simplex extremes are truncated to projections onto the simplex boundary. We assume that the actor provides natural performances as data, and extremes of emotion, speech, and blinks as the basis.

As future work, we are interested in examining how components such as rotation and dynamics of performance can be retargeted via a parameter-parallel layered approach. For instance, the natural performance of an actor may need to be nonlinearly sped up for a squirrel, or slowed down for a globular character like Jabba the Hutt. Similarly, we currently transfer rotation and translation parameters of head motion directly from the actor to the character. Future work may address cases where a particular head orientation for the actor may not directly map to that of the character.

Acknowledgments

This work was supported by NSF grant IIS-0916272. We would like to acknowledge the work of Moshe Mahler, Spencer Diaz, and Valeria Reznitskaya in modeling, rigging, and animating our characters. We thank Eakta Jain and Srinivasa Narasimhan for their valuable input.

References

[BFJ*00] BUCK I., FINKELSTEIN A., JACOBS C., KLEIN A., SALESIN D. H., SEIMS J., SZELISKI R., , TOYAMA K.: Performance-driven hand-drawn animation. *First International Symposium on Non Photorealistic Animation and Rendering* (2000), 101–108. 3

[BLB*08] BICKEL B., LANG M., BOTSCH M., OTADUY M. A., GROSS M.: Pose-space animation and transfer of facial details. *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2008). 3

[BLCD02] BREGLER C., LOEB L., CHUANG E., DESHPANDE H.: Turning to the masters: Motion capturing cartoons. *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques* (2002). 3

[BVGP09] BARAN I., VLASIC D., GRINSPUN E., POPOVIC J.: Semantic deformation transfer. *ACM Transactions on Graphics 28*, 3 (2009). 3

[CB02] CHUANG E., BREGLER C.: Performance driven facial animation using blendshapes interpolation. *Technical Report CS-TR-2002-02, Stanford University* (2002). 3

[CB05] CHUANG E., BREGLER C.: Mood swings: Expressive

speech animation. *ACM Transactions on Graphics 24*, 2 (2005), 331–347. 3

[CBK*06] CURIO C., BREIDT M., KLEINER M., VUONG Q. C., GIESE M. A., BULTHOFF H. H.: Semantic 3d motion retargeting for facial animation. *Proceedings of the 3rd symposium on Applied perception in graphics and visualization* (2006). 3

[CFP03] CAO Y., FALOUTSOS P., PIGHIN F.: Unsupervised learning for speech motion editing. In *Proceedings of the ACM SIGGRAPH/Eurographics symposium on Computer animation* (Aire-la-Ville, Switzerland, Switzerland, 2003), Eurographics Association, pp. 225–231. 3

[CLK01] CHOE B., LEE H., KO H.-S.: Performance-driven muscle-based facial animation. *Journal of Visualization and Computer Animation 12*, 2 (2001), 67–79. 2, 3

[DCFN06] DENG Z., CHIANG P.-Y., FOX P., NEUMANN U.: Animating blendshape faces by cross-mapping motion capture data. *Proceedings of the Symposium on Interactive 3D graphics and games* (2006). 3

[EFH02] EKMAN P., FRIESEN W. V., HAGER J. C.: *Facial Action Coding System: The Manual*. 2002. 3

[Gle98] GLEICHER M.: Retargetting motion to new characters. *Proceedings of ACM SIGGRAPH* (1998). 2

[JTDP03] JOSHI P., TIEN W. C., DESBRUN M., PIGHIN F.: Learning controls for blend shape based realistic facial animation. *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2003). 3

[LWP10] LI H., WEISE T., PAULY M.: Example-based facial rigging. *ACM Transactions on Graphics 29*, 4 (2010). 2

[MB04] MATTHEWS I., BAKER S.: Active appearance models revisited. *International Journal of Computer Vision 60*, 2 (2004), 134–164. 2, 4, 7

[MJC*08] MA W.-C., JONES A., CHIANG J.-Y., HAWKINS T., FREDERIKSEN S., PEERS P., VUKOVIC M., OUHYOUNG M., DEBEVEC P.: Facial performance synthesis using deformation-driven polynomial displacement maps. *ACM Transactions on Graphics 27*, 5 (2008). 2, 3

[NJ04] NA K., JUNG M.: Hierarchical retargeting of fine facial motions. *Eurographics 23*, 3 (2004). 2, 3

[Osi07] OSIPA J.: *Stop Staring*. Sybex, 2007. 3

[PKC*03] PYUN H., KIM Y., CHAE W., KANG H. W., SHIN S. Y.: An example-based approach for facial expression cloning. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer animation* (2003), pp. 167–176. 3

[PL06] PIGHIN F., LEWIS J. P. (Eds.):. *Performance-Driven Facial Animation* (2006), SIGGRAPH Course Notes. 2, 3

[Plu80] PLUTCHIK R.: Emotion, a psychoevolutionary synthesis. *Harper & Row* (1980). 5

[PW96] PARKE F. I., WATERS K.: *Computer Facial Animation*. A K Peters, 1996. 2, 3

[SP04] SUMNER R. W., POPOVIC J.: Deformation transfer for triangle meshes. *ACM Transactions on Graphics 23*, 3 (2004). 3

[VBPP05] VLASIC D., BRAND M., PFISTER H., POPOVIC J.: Face transfer with multilinear models. *ACM Transactions on Graphics 24*, 3 (2005), 426–433. 3

[WLGP09] WEISE T., LI H., GOOL L. V., PAULY M.: Face/off. *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2009). 2, 3

[ZLT*06] ZHANG Q., LIU Z., TERZOPOULOS D., GUO B., SHUM H.: Geometry-driven photorealistic facial expression synthesis. *IEEE Transactions on Visualization and Computer Graphics 12*, 1 (2006), 48–60. 3

[ZWT09] ZHANG W., WANG Q., TANG X.: Performance driven face animation via non-rigid 3d tracking. *Proceedings of the 17th ACM International Conference on Multimedia* (2009). 3