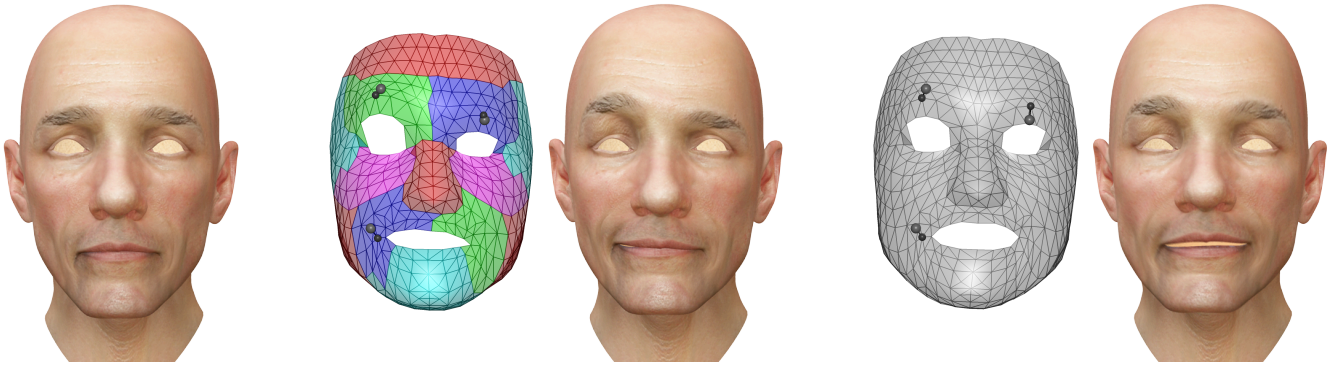
# Interactive Region-Based Linear 3D Face Models

J. Rafael Tena*
Disney Research Pittsburgh

Fernando De la Torre†
The Robotics Institute
Carnegie Mellon University

Iain Matthews‡
Disney Research Pittsburgh



(a) (b) (c) (d) (e)

**Figure 1:** *Face posing using interactive region-based (b) and holistic (d) face models. The models drive the human character shown in (a). User-given constraints (black markers) create a wink with a smirk, when issued to the region-based model (b and c). In contrast, the same constraints produce uncontrolled global deformations when the holistic model is used (d and e).*

## Abstract

Linear models, particularly those based on principal component analysis (PCA), have been used successfully on a broad range of human face-related applications. Although PCA models achieve high compression, they have not been widely used for animation in a production environment because their bases lack a semantic interpretation. Their parameters are not an intuitive set for animators to work with. In this paper we present a linear face modelling approach that generalises to unseen data better than the traditional holistic approach while also allowing *click-and-drag* interaction for animation. Our model is composed of a collection of PCA sub-models that are independently trained but share boundaries. Boundary consistency and user-given constraints are enforced in a soft least mean squares sense to give flexibility to the model while maintaining coherence. Our results show that the region-based model generalises better than its holistic counterpart when describing previously unseen motion capture data from multiple subjects. The decomposition of the face into several regions, which we determine automatically from training data, gives the user localised manipulation control. This feature allows to use the model for face posing and animation in an intuitive style.

**CR Categories:** I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation;

**Keywords:** Face modelling, animation, linear model, piece-wise model, interactive model

*e-mail:rafael.tena@disneyresearch.com
†e-mail:ftorre@cs.cmu.edu
‡e-mail:iainm@disneyresearch.com

**Links:** DL PDF

## 1 Introduction

Linear models, particularly those based on principal component analysis (PCA), have been used successfully on a broad range of human face-related applications, examples include Active Appearance Models [Cootes et al. 1998; Matthews and Baker 2004] and 3D Morphable Models [Blanz and Vetter 1999]. In the production of computerised facial animation, a common practice is to use blendshape animation models (or rigs). These models aim to represent a given facial configuration as a linear combination of a predetermined subset of facial poses that define the valid space of facial expressions [Bergeron and Lachapelle 1985; Pighin et al. 1998]. PCA and blendshape models differ from each other only in the nature of their basis vectors. The bases are orthogonal and lack a semantic meaning in PCA, versus non-orthogonal with an artist defined and interpretable meaning for blendshape models.

Although PCA models achieve high compression, they are not generally used for animation because their bases lack semantic interpretation. Their parameters are not an intuitive set for animators to work with. This is typically not the case for blendshape models. However, until recently there were few published methods to manipulate blendshape models other than directly specifying the blend weights [Lewis and Anjyo 2010; Joshi et al. 2003]. The work of

Lewis and Anjyo [2010] describes a mathematical framework for computing the blend weights of a blendshape model from a sparse set of user-given constraints, including the extreme case of a single constraint. Their solution can be computed at interactive rates and allows the animator to work by clicking and dragging parts of the mesh directly.

In this paper, we present a linear piecewise modelling method that generalises better to unseen data than the traditional holistic approach. The segmentation of the model into multiple parts allows the generation of parts combinations beyond those in the training data. Our model also allows click-and-drag interaction for animation, in the spirit of Lewis and Anjyo's [2010] direct-manipulation blendshapes work. We refer to our solution as *region-based linear models*. These models consist of a collection of linear models that relate to each other through shared boundaries. Rather than following the convention of weighted blending for managing boundary discontinuities, our formulation restricts the model solutions to have semi-consistent boundaries while simultaneously enforcing user-given constraints in a soft least squares sense (Section 3). We demonstrate our approach on dense facial motion capture data using a PCA region-based face model, and we show that our model is also flexible enough to generalise to multiple people (Section 5). The segmentation of the face into multiple sub-models allows us to interactively modify the model at a local level, which is typically not possible with a holistic PCA model (see Figure 1). In the context of face-posing for keyframe animation (Section 6), this means the region-based model is locally intuitive and globally consistent. Our mathematical formulation is applicable to linear models in general, regardless of whether they are based on PCA or other linear basis.

## 2 Related Work

Linear PCA models have been used successfully for a wide range of tasks in the fields of Computer Vision and Computer Graphics. In some cases, linear models have been replaced by non-linear models that provide better performance for some applications at the expense of increased computational complexity [Lawrence 2007]. However, linear models continue to be of common use because of their simplicity and inexpensive computational nature. One of the earlier works that introduced PCA models into Computer Vision is that of Turk and Pentland [1991] with their *eigenface* approach to face recognition. Their work was followed by Cootes *et al.* [1998] with active shape and active appearance models, which have been used for face recognition [Edwards et al. 1998] and tracking [Matthews and Baker 2004], among many other applications. Blanz and Vetter [1999] applied the concept of active appearance models to the realm of 3D and Computer Graphics with their work in face morphable models. Allen *et al.* [2003] built linear PCA shape models of human bodies in 3D. More recently, Vlasic *et al.* [2005] described a higher order generalisation of the linear model called multilinear which they used for modelling identity, expression, and speech independently. The work cited here comprises only a sample of the successful application of linear models.

Previous work in face modelling has also decomposed faces into regions to improve expressiveness. An early example is the work by Black and Yacoob [1995] that explored the use of local parameterised models for recovering and recognising non-rigid motion in human faces. DeCarlo and Metaxas [2000] manually built a parametric deformable 3D face model for tracking faces on video. The shape of the model is controlled by parameterised deformations that are applied to a particular set of face parts, ranging from a single part to the entire face. In their work on 3D morphable models, Blanz and Vetter [1999] divided the face into four regions to augment the expressiveness of their PCA model. When fitting the model to images, each region was optimised independently and the results were then blended following a Gaussian pyramid approach. Joshi *et al.* [2003] demonstrated an automatic, physically-motivated approach to segment a blendshape face model. Similar to our approach, the segmented regions have overlapping boundaries. To fit their blendshape model to motion capture data, each region was optimised independently to find the corresponding blending weights. In contrast, our approach solves simultaneously for all regions while explicitly enforcing soft boundary consistency. Zhang *et al.* [2006] presented a system for synthesising facial expressions applicable to 2D images and 3D face models. They empirically divided the face into several regions to allow the synthesis of asymmetric expressions. To avoid image discontinuity at region boundaries, they did a fade-in-fade-out blending using a manually defined weight map. Zhang and colleagues [2004] created a linear 3D face model for animation that is automatically segmented into regions at runtime depending on the user edits. The segmented regions are independently modelled and then blended into a single expression. Overall, their model behaves holistically unless the user specifies constraints to anchor desired locations. Because the number of regions is constantly changing, their model does not have a defined parameter space and interaction is limited to explicitly controlling points on the mesh. In contrast, our model establishes a mapping between parameter space and face configuration, thus allowing parametric manipulation.

There is also related work on creating face models that allow localised modifications without explicitly dividing the face into regions. Noh and colleagues [2000] used radial basis functions to produce localised real-time deformations by controlling an arbitrary sparse set of control points. Feng and collaborators [2008] described an animation interface that learns optimal control points from a set of surface deformation examples. The control points are mapped through canonical correlation analysis to a sparse set of abstract bones whose deformation parameters specify the deformation of every vertex on the surface. The user can then create new deformations by modifying the position of the optimal control points. In contrast, our approach does not limit user interaction to a set of predefined points. Lau *et al.* [2009] presented a system for interactive modeling of 3D facial expressions using facial priors. They formulated the problem of face posing in a maximum *a posteriori* framework that combines user inputs with priors embedded in a large set of facial expression data. Their approach optimises a non-linear cost function with terms for different types of user-provided constraints and facial priors from expression data. This method allows the user to modify any region of the face. However, the local behaviour is limited by the data available in the database of facial expressions. Meyer and Anderson [2007] described a scheme that uses examples to compute a statistical subspace and a corresponding set of characteristic key points. For a given operation, the required calculations are only computed at the key points and the result is used to provide a subspace based estimate of the entire calculation. They demonstrated their method on the problem of calculating the facial articulation of an animated character. The subspace is first obtained using PCA, and the resulting basis vectors are then rotated to obtain a basis that is more semantically meaningful.

More recently, Lewis and Anjyo [2010] presented a mathematical formulation for manipulating blendshape models based on user-provided constraints. Their approach is the closest to that presented here and provides a least squares solution to the ill-posed problem of computing the model's blending weights from a sparse set of constraints, but it can only produce facial configurations that are linear combinations of the blendshapes in the model. Conversely, by segmenting the face into multiple regions, our approach increases the range of possible configurations. For instance, our model can produce data consistent asymmetric expressions even if the training set contains only symmetric ones.

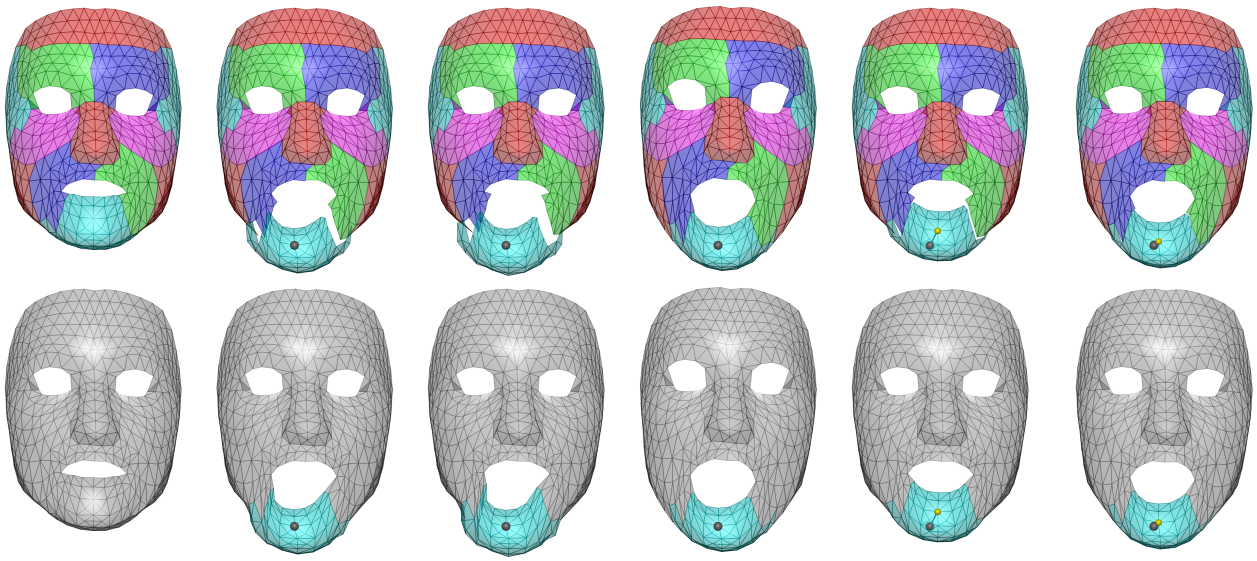



**Figure 2:** *Interaction with a region-based model with various combinations of $\beta$ (boundary constraint) and $\gamma$ (deformation constraint). The top row shows results before boundary blending and the bottom row after blending (see section 3.1). The user-given constraint is showed in black, and the constrained vertex in yellow. Low values of $\beta$ de-couple the sub-models, resulting in potentially large boundary discrepancies. With higher values of $\beta$ enforcing boundary consistency, a low value of $\gamma$ allows unrestrained deformation updates and produces nearly holistic behaviour; while a high $\gamma$ makes the model resist change. A medium value, such as $\gamma = 0.025$ provides local control with good boundary consistency.*

## 3 Region-Based Linear Face Modelling

Building a modular subspace in order to increase a model's expressiveness is not new in the literature [Nishino et al. 2005; Pentland et al. 1994]. It has also been demonstrated in the context of active appearance models and their morphable model 3D counterparts that segmenting a model increases its expressiveness and therefore its ability to generalise [Blanz and Vetter 1999; Peyras et al. 2007]. In previous work, the face has been segmented into regions using automatic methods [Joshi et al. 2003] or by manual selection [Zhang et al. 2006]. Each region is then treated as an independent model, and a blending scheme is implemented to deal with discontinuities at the inter-region boundaries [Buck et al. 2000]. In contrast, our approach solves for all the sub-models simultaneously while explicitly enforcing boundary consistency in a soft least squares sense. The use of soft constraints allows discrepancies at the inter-model boundaries, keeping the model flexible, and the simultaneous solve requires all the sub-models form a coherent unit.

### 3.1 Region-Based Model Building

A linear model may be defined by the following equation:

$$\mathbf{v} = \mathbf{B}\mathbf{c}. \tag{1}$$

In the specific case of 3D faces, $\mathbf{v} \in \Re^{3N\times 1}$ is a vector containing the $(x, y, z)$ spatial coordinates of the $N$ *vertices* in a mesh that represents the face; $\mathbf{B} \in \Re^{3N\times P}$ contains the $P$ linear bases of the model, and $\mathbf{c} \in \Re^{P\times 1}$ the corresponding parameters. The model parameters, $\mathbf{c}$, that best describe the input data, $\mathbf{v}$, in a least squares sense can be found by minimising:

$$E(\mathbf{c}) = \|\mathbf{v} - \mathbf{B}\mathbf{c}\|_2^2. \tag{2}$$

Equation 2 is a convex function of $\mathbf{c}$ and its minimum occurs at the unique extremum where its derivative with respect to $\mathbf{c}$ vanishes:

$$\mathbf{B}^T\mathbf{B}\mathbf{c} - \mathbf{B}^T\mathbf{v} = \mathbf{0}, \tag{3}$$

yielding the closed form solution,

$$\mathbf{c} = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\mathbf{v}, \tag{4}$$

which reduces to $\mathbf{c} = \mathbf{B}^T\mathbf{v}$ if $\mathbf{B}$ is an orthonormal basis.

In a region-based face model, different sections $\mathbf{v}^i \subset \mathbf{v}$ are independently modelled. Each region contains a subset $N^i$ of the vertices that compose the full face. The only requirement for defining the regions is that each shares at least one vertex with at least one other region. The shared vertices will be referred to as *boundary* vertices. For a region-based model with $M$ regions, Equation 2 is replaced by:

$$E(\boldsymbol{\zeta}) = \sum_{i=1}^{M} \|\mathbf{v}^i - \mathbf{B}^i\mathbf{c}^i\|_2^2, \tag{5}$$

$$\boldsymbol{\zeta} = [\mathbf{c}^1; \mathbf{c}^2; \ldots; \mathbf{c}^M]$$

where $\mathbf{B}^i \in \Re^{3N^i\times P^i}$ contains the $P^i$ bases of the $i^{th}$ sub-model, which models the $i^{th}$ region $\mathbf{v}^i \in \Re^{3N^i\times 1}$ with model coefficients $\mathbf{c}^i \in \Re^{P^i\times 1}$. However, boundary vertices should remain consistent in all the regions that share them; therefore, Equation 5 is reformulated to enforce equality at the boundaries:

$$E(\boldsymbol{\zeta}) = \sum_{i=1}^{M} \|\mathbf{v}^i - \mathbf{B}^i\mathbf{c}^i\|_2^2 + \beta \sum_{i=1}^{M}\sum_{j=i}^{M} \|\mathbf{B}_j^i\mathbf{c}^i - \mathbf{B}_i^j\mathbf{c}^j\|_2^2, \tag{6}$$

where $\mathbf{B}_j^i \in \Re^{3Q\times P^i}$ contains the elements of bases $\mathbf{B}^i$ that model the $Q$ boundary vertices shared by the $i^{th}$ and $j^{th}$ regions; if no

vertices are shared then $\mathbf{B}_j^i \in \mathbf{0}$. Constant $\beta$ weights the contribution of the boundary constraints to that of the reconstruction error. The minimum of Equation 6 occurs where the gradient vanishes. That is $[\partial E/\partial \mathbf{c}^1; \ldots; \partial E/\partial \mathbf{c}^M] = \mathbf{0}$. Each partial derivative with respect to $\mathbf{c}^i$ of Equation 6 yields:

$$\mathbf{B}^{i\,T}\mathbf{B}^i\mathbf{c}^i - \mathbf{v}^{i\,T}\mathbf{B}^i + \beta \sum_{j=1}^{M} \left( \mathbf{B}_j^{i\,T}\mathbf{B}_j^i\mathbf{c}^i - \mathbf{B}_j^{i\,T}\mathbf{B}_i^j\mathbf{c}^j \right) = \mathbf{0}. \quad (7)$$

The coefficients of the gradient can be arranged in matrix form to define a system of linear equations on $\boldsymbol{\zeta}$, that can be directly solved by Gauss elimination, conjugate gradient, or equivalent methods. Because boundary consistency is enforced in a soft least squares sense, there are always differences at the sub-model boundaries. These discrepancies are eliminated by taking the mean value of the boundary vertices shared by the sub-models (see Figure 2). This approach suffices because the boundary differences have been minimised by the boundary constraints. However, gradient domain methods could be used to distribute the boundary differences across all the vertices of the sub-models [Sorkine et al. 2004].

### 3.2 Interacting with the Model

We extend our region-based framework to allow user interaction. Our approach follows that of Lewis and Anjyo [2010] to provide click and drag interaction, in which the user is allowed to manipulate vertices on the mesh of the face model and constrain them to a desired location. More generally, the user may decide to constrain one vertex, several, all or none; the approach is not limited by the number of user-given constraints. Consider the case of a single model for which the user has provided constraints for all vertices. Equation 2 exemplifies this case, and Equation 4 provides the solution. If the number of user-given constraints falls beneath the number of the model's bases (which determines the degrees of freedom) the system becomes underconstrained. Lewis and Anjyo [2010] address this problem by also minimising the Euclidean distance in parameter space from the previous state to the solution,

$$E(\mathbf{c}) = \sum_{k=1}^{K} \|\mathbf{v}_k - \mathbf{B}_k\mathbf{c}\|_2^2 + \gamma\|\mathbf{c}_0 - \mathbf{c}\|_2^2, \quad (8)$$

where $\mathbf{v}_k \in \Re^{3\times 1}$ is the $k^{th}$ user-given constraint, $\mathbf{B}_k \in \Re^{3\times P}$ are the corresponding bases, $\mathbf{c}_0 \in \Re^{P\times 1}$ are the model's parameters before the vertex constraints were given, and $\gamma$ is a constant that weights the regularisation term. For our region-based model, we extend Equation 8 to multiple sub-models and we incorporate boundary constraints,

$$E(\boldsymbol{\zeta}) = \sum_{i=1}^{M}\sum_{k=1}^{K} \|\mathbf{v}_k^i - \mathbf{B}_k^i\mathbf{c}^i\|_2^2 + \beta \sum_{i=1}^{M}\sum_{j=i}^{M} \|\mathbf{B}_j^i\mathbf{c}^i - \mathbf{B}_i^j\mathbf{c}^j\|_2^2 + \gamma \sum_{i=1}^{M} \|\mathbf{c}_0^i - \mathbf{c}^i\|_2^2, \quad (9)$$

where $\mathbf{v}_k^i \in \Re^{3\times 1}$ is the $k^{th}$ user given constraint for the $i^{th}$ model, $\mathbf{B}_k^i \in \Re^{3\times P^i}$ are the corresponding bases, and $\mathbf{c}_0^i \in \Re^{P^i\times 1}$ are the initial model parameters of the $i^{th}$ model. The second and third terms of Equation 9 control the coupling between sub-models. Notice that as $\beta$ (boundary constraint) and $\gamma$ (deformation constraint) approach zero, the region-based model becomes a collection of independent models described by Equation 2. Figure 2 illustrates the behaviour of a region-based model with a single user-given constraint for different values of $\beta$ and $\gamma$ before and after boundary blending (see Section 3.1). Low values of $\beta$ de-couple the sub-models, which may result in large boundary discrepancies and distortion after blending. With higher values of $\beta$ to enforce boundary consistency, variations in $\gamma$ modulate coupling and deformation. A low value of $\gamma$ produces holistic-like behaviour because all sub-models deform freely to accommodate the boundary constraints. Conversely, a high $\gamma$ makes the model resist change to maintain boundary consistency and current sub-model parameters. Intermediate values such as $\gamma = 0.025$ provide local control by balancing boundary inconsistencies and changes in sub-model parameters.

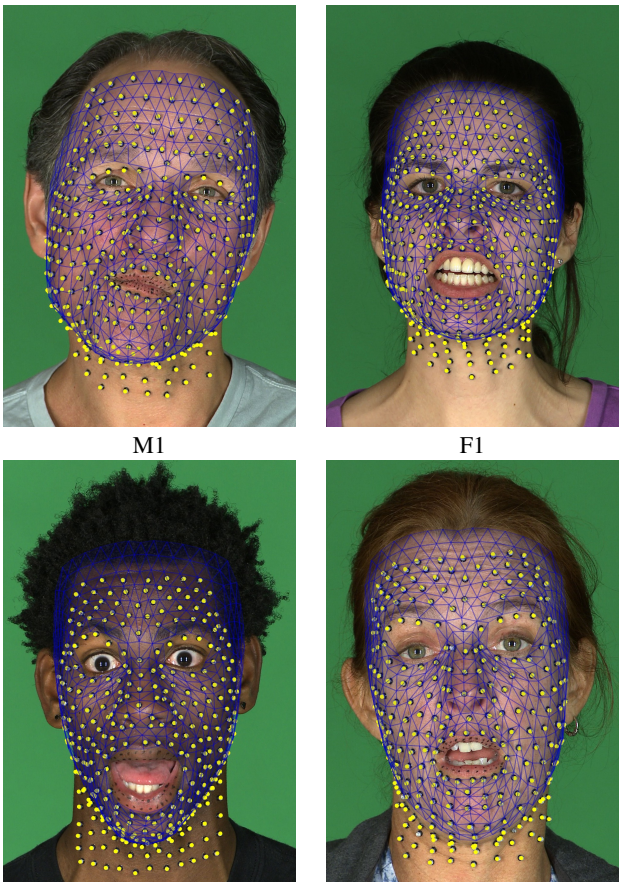



**Figure 3:** *The four actors used in our experiments. A generic face mesh (blue) was fitted to all the motion capture data (yellow) to establish dense correspondence.*

Lewis and Anjyo [2010] minimise Equation 8 subject to $\mathbf{c} \in [0, t = 1]$ as a quadratic program, in order to constrain the model parameters to the positive interval valid for artistically designed blendshape models. We do not assume such constraints exist, and instead minimise Equation 9 by finding the point at which the gradient vanishes. The partial derivative of Equation 9 with respect to $\mathbf{c}^i$ yields,

$$\sum_{k=1}^{K} \left( \mathbf{B}_k^{i\,T}\mathbf{B}_k^i\mathbf{c}^i - \mathbf{v}_k^{i\,T}\mathbf{B}_k^i \right) + \beta \sum_{j=1}^{M} \left( \mathbf{B}_j^{i\,T}\mathbf{B}_j^i\mathbf{c}^i - \mathbf{B}_j^{i\,T}\mathbf{B}_i^j\mathbf{c}^j \right) + \gamma\left( \mathbf{c}^i - \mathbf{c}_0^i \right) = \mathbf{0} \quad (10)$$

The coefficients of the gradient can be arranged in matrix form to define a system of linear equations on $\boldsymbol{\zeta}$ that can be directly solved by Gauss elimination, conjugate gradient, or equivalent methods.

## 4 Facial Motion Capture Data Modelling

We applied our region-based linear model formulation to the problem of modelling dense facial motion capture data. For this purpose, we collected a database that includes captures from four different individuals, three professional actors and one art student. The subjects covered both genders, two ethnicities, and a broad age spectrum. We will refer to them as M1, M2, F1 and F2 (see Figure 3). Each subject performed 18 sentences. Prior to the recording of each sentence, the actor was given a background story to provide the inspiration for an emotional performance. All actors performed the same sentences with the same emotional background. The emotions the actors performed were pride, rage and contempt. The data was split in training and test sets for each actor. The training sets contained four sentences from each emotion, while the test sets contained two per emotion. Actors wore 3 $mm$ reflective markers spaced approximately 1 $cm$ apart and were recorded using a commercial motion capture system at 120 frames per second (fps). For actor M1 we collected additional data consisting of a range of motion sequence, during which the actor performs random facial motion to achieve extreme expressions, and 86 combinations of facial action coding system (FACS) units as described by Ekman and Friesen [1978].

### 4.1 Motion Capture Data Registration

The creation of linear models that span the data of multiple people requires establishing dense point-to-point correspondences across the data set [Allen et al. 2003; Blanz and Vetter 1999]. This means that the position of each vertex may vary in different samples, but its context label should remain the same. During capture, our actors wore different numbers of reflective markers depending on the dimensions of their faces. Additionally, no two actors had an identical marker configuration. To establish dense correspondence, we fit a dense 3D generic mesh template with 8820 vertices to our entire motion capture database using the method of Tena *et al.* [2006]. We then uniformly subsample the fitted dense meshes down to 397 vertices, providing us with a data set of 3D meshes that are in full dense correspondence. Figure 3 shows our four subjects with the original markers and the subsampled fitted mesh. Finally, we remove rigid-body transformations from our data by aligning each motion capture frame to the subsampled generic mesh template using ordinary procrustes analysis [Dryden and Mardia 2002].

### 4.2 Face Region Segmentation

To create the sub-models for our region-based model framework, we are interested in grouping vertices that are highly correlated and connected to form compact regions. Regions containing highly correlated vertices will be better compressed by PCA. To find a data-driven segmentation of the face, we used the full data set from M1. This set includes the range of motion, emotional speech, and FACS sequences. The data is subsampled to 15 fps to reduce redundancy, after which the data set contains 4787 motion capture frames. Assuming data set $\mathbf{D} \in \Re^{F\times 3N}$ with $F$ frames and $N$ 3D vertices, we split $\mathbf{D}$ into three subsets $\mathbf{D}_{i=\{x,y,z\}} \in \Re^{F\times N}$; each containing the corresponding spatial coordinate of the vertices. To obtain a measurement of the correlation between vertices, the normalised correlation matrices $\mathbf{C}_{i=\{x,y,z\}} \in \Re^{N\times N}$ are computed from $\mathbf{D}_{i=\{x,y,z\}}$ and then averaged into correlation matrix $\bar{\mathbf{C}}$. Vertices in the same region should also be close to each other on the face surface. Accordingly, we also compute the inter-vertex distance on the mesh as described for the isomap algorithm [Tenenbaum et al. 2000] to form matrix $\mathbf{G} \in \Re^{N\times N}$. In order to combine

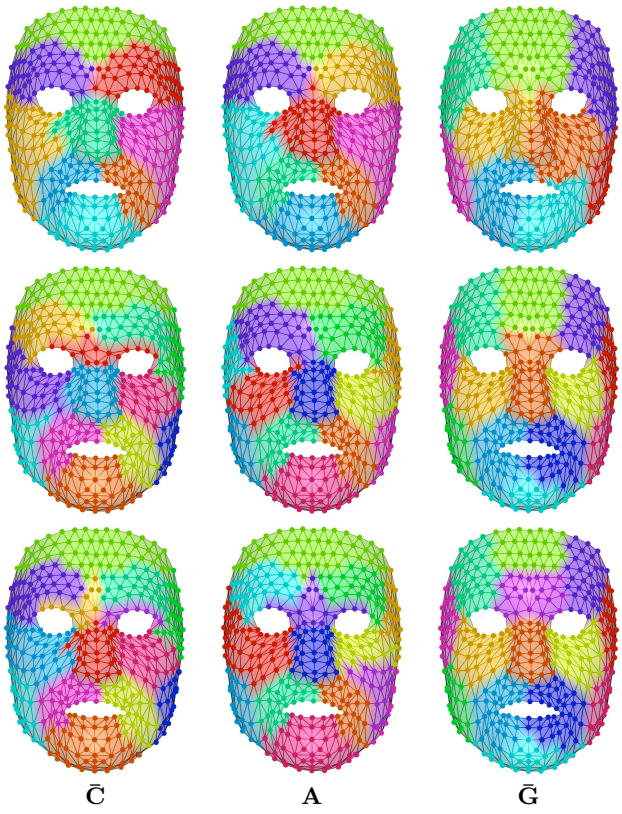



**Figure 4:** *Clustering results on motion capture data from M1. Each row shows the results for 9, 13, and 14 clusters. Notice that for 14 clusters,* $\mathbf{A}$ *produces the non-compact purple cluster in the middle of the forehead. Accordingly segmentation is stopped at 13 clusters.*

$\bar{\mathbf{C}}$ and $\mathbf{G}$, we normalise the latter to the $[0, 1]$ interval,

$$\bar{\mathbf{G}} = e^{-\mathbf{G}/r}, \tag{11}$$

where $r$ controls how rapidly $\bar{\mathbf{G}}$ decreases as mesh distance increases. Both matrices are added in a weighted manner into the affinity matrix,

$$\mathbf{A} = \lambda\bar{\mathbf{C}} + (1-\lambda)\bar{\mathbf{G}}, \tag{12}$$

where $\lambda$ controls the relative importance of the inter-vertex correlation and distance on the mesh. The values of $\lambda$ and $r$ can be adjusted to obtain different regions. For our experiments, $r = 10$ (at this value $\bar{\mathbf{G}}$ approaches zero for mesh distances greater than 50 mm) and $\lambda = 0.7$ to emphasize correlation over mesh distance. Additionally, prior to calculating $\mathbf{A}$, we thresholded $\bar{\mathbf{C}}$ at 0.7 to obtain regions with high correlation values. Finally, we perform spectral clustering on the affinity matrix $\mathbf{A}$, as described by Ng *et al.* [2001], which relies on the k-means algorithm. We start with $k = 2$ (two clusters) and continue increasing $k$ until one of the created clusters is non-compact. At this point we stop increasing $k$, and we keep the clusters obtained with $k - 1$. A cluster is defined as non-compact if some of its vertices do not form triangles on the template mesh. In our experiments, we obtained 13 compact clusters. To account for variability due to seed selection, the k-means algorithm was performed 20 times for each value of $k$ using random seeds. The clustering result with the smallest within-cluster sums of point-to-centroid distances was kept. Figure 4 shows the clustering results on M1's motion capture data. For comparison, we also show the results obtained using $\bar{\mathbf{G}}$ and $\bar{\mathbf{C}}$ as affinity matrices on their own. Notice that the clustering results for $\mathbf{A}$ and $\bar{\mathbf{C}}$

are very similar, as intended by our choice of $\lambda$. However, for 13 clusters, $\bar{\mathbf{C}}$ produces a non-compact cluster (red) while $\mathbf{A}$ does not. Matrix $\bar{\mathbf{G}}$ produces only compact clusters, which explains the difference in results obtained with $\bar{\mathbf{C}}$ and $\mathbf{A}$. Notice that the clusters obtained with matrices $\bar{\mathbf{C}}$ and $\mathbf{A}$ are nearly symmetric, an intuitive but not enforced result. After segmentation, we manually define the vertices that are shared by the different regions.

## 5 Results

To test how our region-based model generalises to unseen data and across multiple identities, we built region-based and holistic PCA models from our emotional speech motion capture data. For the region-based model, the face was split into the 13 regions obtained by spectral clustering as previously described. PCA was applied to each region independently to produce its own subspace. The experiment began by training the holistic and region-based models only with the training set from individual M1, reconstructing each of the four available test sets (M1, F1, M2 and F2), and measuring the reconstruction error as a function of the number of principal components added to the model. We then retrained the region-based and holistic PCA models using the training sets of M1 and F1, and tested again on each of the four test sets. The process was repeated adding M2's training set, and finally that of F2. The reconstruction error reported is the root mean squared error per vertex,

$$E_{\text{RMS}} = \sqrt{\frac{1}{F}\sum_{i=1}^{F}\frac{1}{N}\|\mathbf{x}^i - \hat{\mathbf{x}}^i\|^2}, \qquad (13)$$

where $F$ is the number of frames in the test set, $N$ is the number of vertices, $\mathbf{x}^i$ is the $i^{th}$ frame of the test set, and $\hat{\mathbf{x}}^i$ is its reconstructed counterpart. Because the experiment was aimed at reconstructing data from multiple identities, the PCA models were trained and tested with neutral subtracted data. This was achieved by subtracting the corresponding neutral pose from the training and test set of each individual. In practice, this means that the models learn the space of facial deformation limiting the inclusion of variation due to identity. Accordingly, to fully reconstruct a face, it is necessary to add the correct neutral pose to recover the subject's identity. From an application perspective, this implies that a new neutral pose is required to apply the the PCA models to a new individual. Figure 5 shows plots of the reconstruction error for the different iterations of our experimental set up. The horizontal axes shows the number of principal components retained for each sub-model and for the holistic model. Figure 5 (top row) shows the extreme case in which the models are trained with the data from M1 only. There are two significant facts to be noticed. First, the reconstruction error achieved for the test set of M1 with the region based model is lower than that attained by the holistic model. This shows the region-based model generalises better to unseen data from the same individual. Second, there is less variance on the error obtained across individuals, showing that the region-based model generalises better to multiple identities. Figure 5 (second to fourth rows), show the error progression as more data is added to the training set. Figure 6 shows reconstruction examples for the region-based and holistic models with the corresponding ground truth.

## 6 Application to Face Posing and Animation

In section 3.2, we presented a mathematical framework to interact with the region-based model by specifying positional vertex constraints. Our formulation allows the user to constrain one, multiple, all, or none of the vertices of the model's face mesh. The model's equation finds the best solution that satisfies, in a soft least mean squares sense, the user-provided constraints and the model's

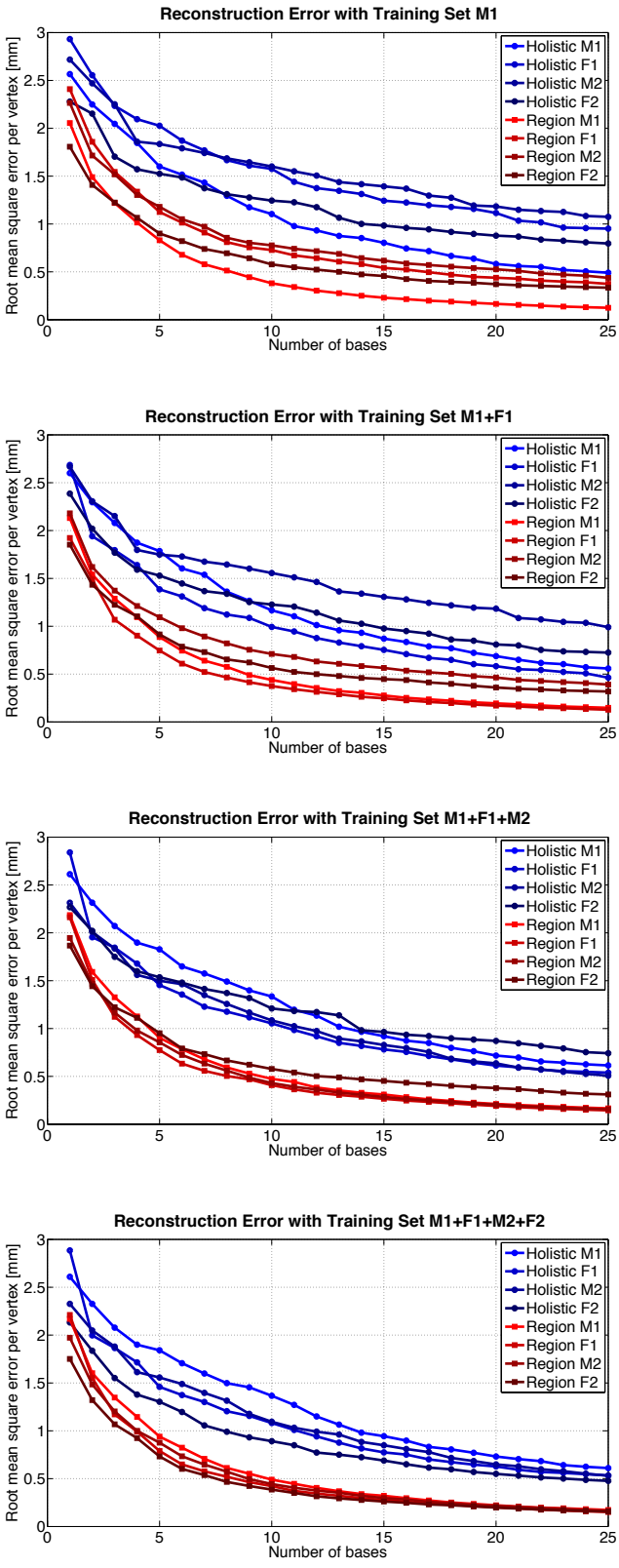



**Figure 5:** *Reconstruction error with holistic and region-based models. As more identities are added to the training set, the reconstruction error on the corresponding test set decreases. Regardless of the training set, the reconstruction error for the region-based model is lower and with less variance across individuals.*

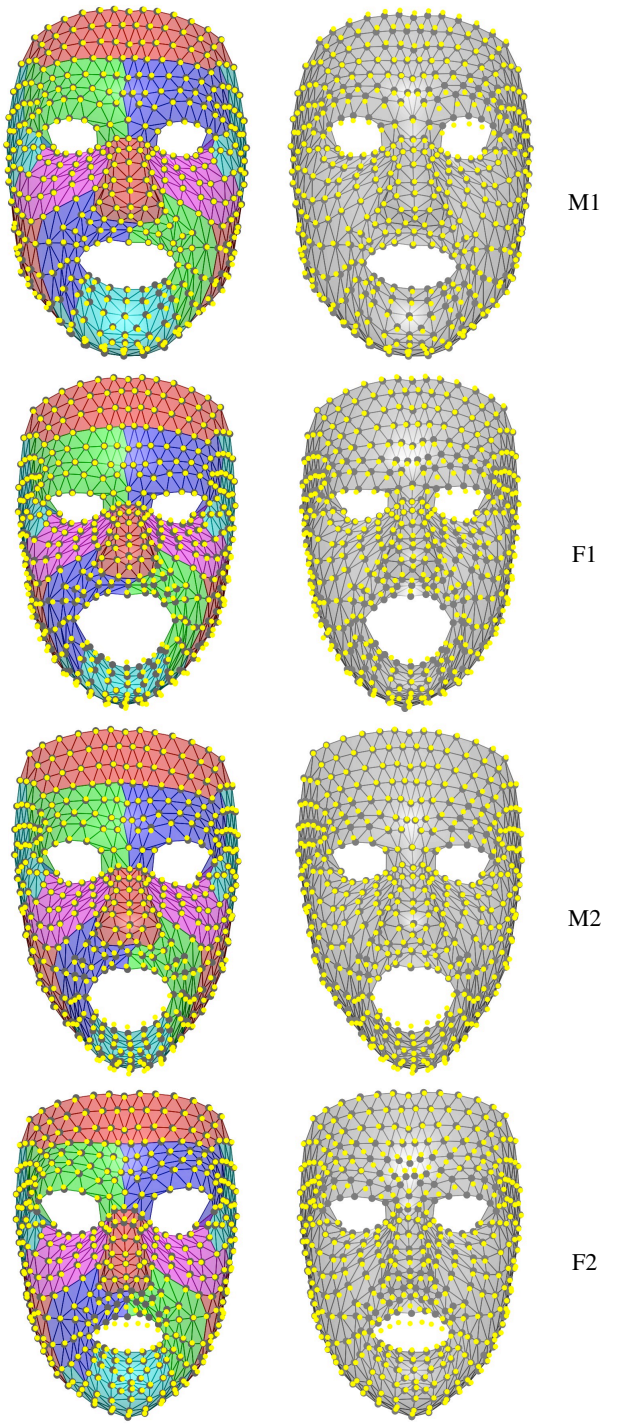



**Figure 6:** *Reconstruction error examples with region-based and holistic models. The ground truth is shown in yellow markers and the corresponding vertices on the reconstructed mesh are highlighted in dark grey. The examples shown were obtained when only the data from M1 was used for training. Note that for the region-based model more ground truth markers lie within the grey highlighted vertices of the reconstructed mesh.*

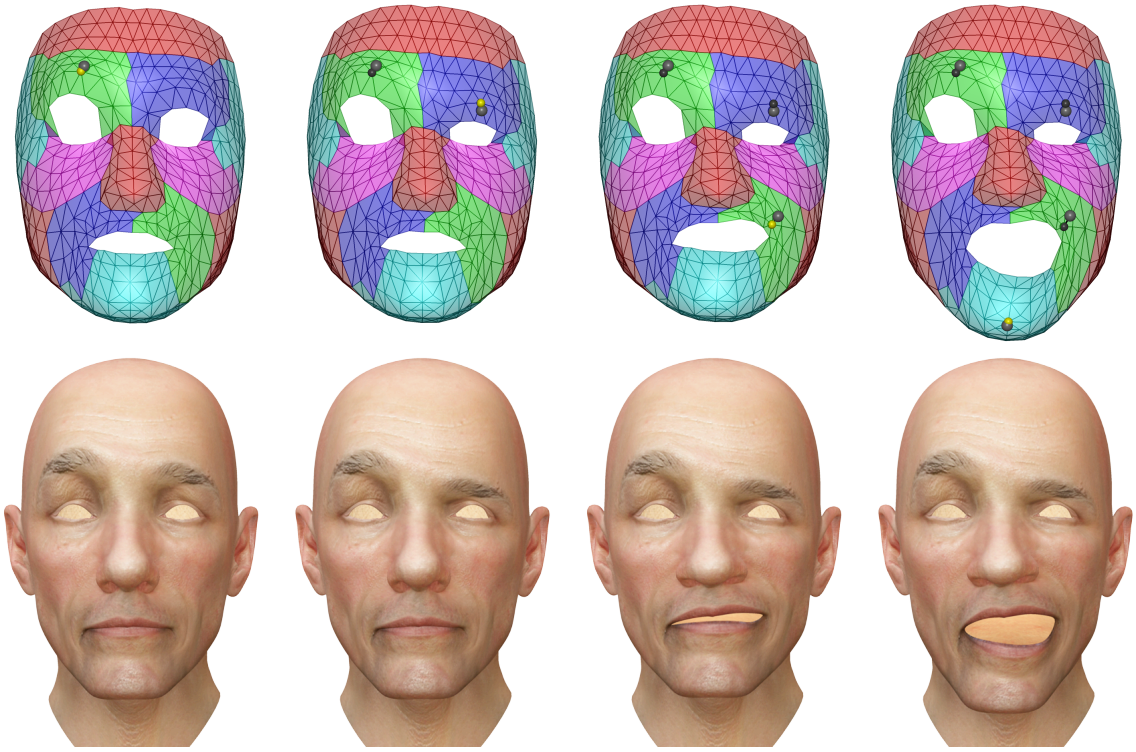
boundary and parameter space constraints (last two terms in Equation 10). The user can produce different model behaviours by adjusting the intrinsic parameters $\beta$ (boundary constraint) and $\gamma$ (deformation constraint). A high value of $\beta$ combined with low $\gamma$ produces nearly holistic behaviour by enforcing boundary consistency and freeing changes in the local parameter space. Relaxing $\beta$ while increasing $\gamma$ allows the user to *mold* the face model without the need of user-given positional constraints because boundary consistency is compromised in order to maintain the current configuration of the sub-models. Intermediate values of $\beta$ and $\gamma$ allow the user to configure the face by explicitly constraining the model's vertices.

Our approach only requires solving a linear system of equations and can be implemented at interactive rates without the need of specialised hardware. We have implemented a real-time MATLAB based prototype system for interacting with region-based models to perform face posing and animation. The system allows the user to specify constraints by clicking on a face vertex and then dragging to the desired location. Once the vertex is released, consecutive constraints may be added in the same manner to sculpt the desired pose. Figures 1 and 7 in this paper were produced using the prototype system with a region-based PCA model consisting of the 13 regions obtained by spectral clustering and trained with the complete motion capture set from subject M1. Parameters $\beta$ and $\gamma$ were set to $0.5$ and $0.025$ respectively. A holistic PCA model was trained with the same data for comparison. To interact with the holistic model, we followed Lewis and Anjyo's [2010] approach and minimised Equation 8 with $\gamma$ set to $0.025$. We did not constrain the solution to the $[0, 1]$ interval (see Section 3.2) because our model is PCA based and does not have that restriction. A sculpted textured model of subject M1 was bound to the region-based and holistic models to demonstrate them on a computer generated character. Figure 7 shows consecutive pose edits specified by user-given constraints. The top shows the region-based model driving the human character, while the bottom shows the results for the holistic model. Edits performed on the region-based model produce local deformation while maintaining global consistency. Conversely, the same edits produce global deformations on the holistic model.

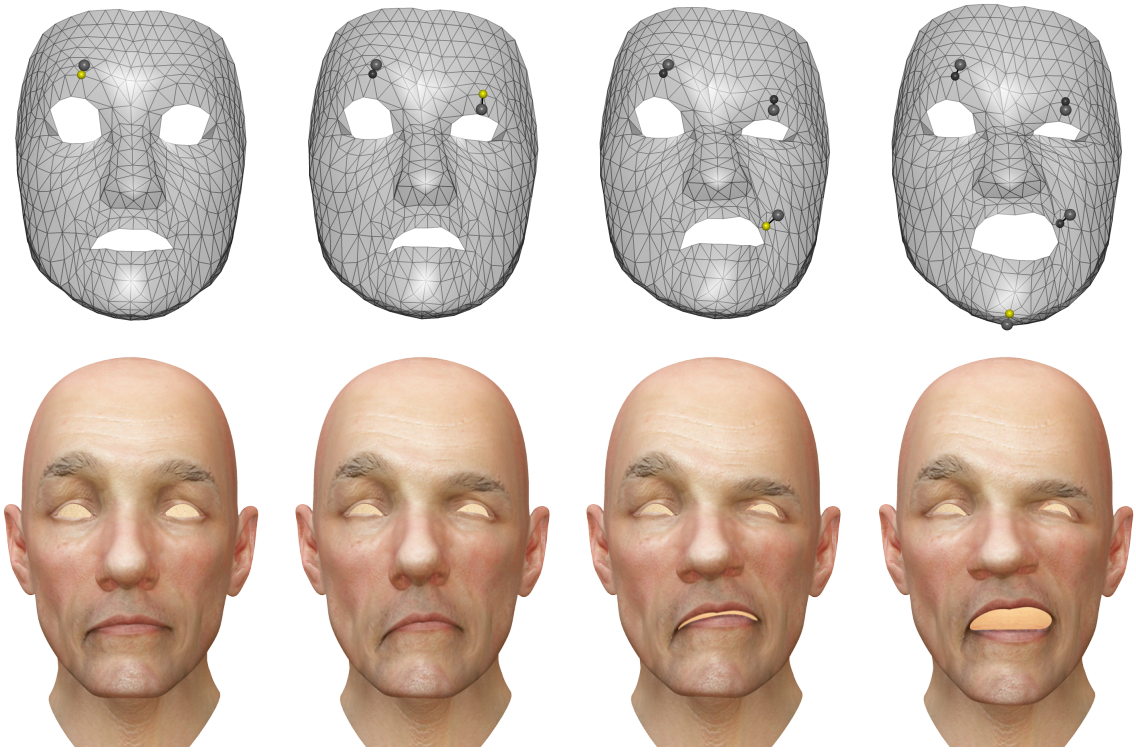
## 7 Discussion

We have presented the mathematical formulation for a linear piecewise modelling approach in which a collection of independently trained models with shared boundaries are coupled and solved simultaneously. This region-based approach increases flexibility for modelling local deformations while keeping the model coherent. The formulation is applicable to models based on a set of linear bases. The region-based formulation was applied to the problem of face modelling and was compared to a holistic approach. Our model also accommodates user interaction by specifying positional constraints, which gives local control for face posing and animation. The coupling and flexibility of the model can be controlled by modifying its two intrinsic parameters $\beta$ (boundary constraint) and $\gamma$ (deformation constraint).

In its current formulation, the main limitation of our model is that it remains constrained to the space learned from its training set. This limitation may also be a benefit for inexperienced animators because it restricts the facial configurations to plausible expressions defined by the model. Nevertheless, experienced animators may find that the model does not allow the exaggeration required to produce emotionally appealing performances. This is especially true for a model trained from real facial motion capture data. Currently we have implemented a simple user interface for demonstrating our technique. However, it needs further development before rigorous user studies can be conducted.



Region-based model



Holistic model

**Figure 7:** *Sequential posing using region-based and holistic models driving a human character. The top shows the results on the region-based model while the bottom shows the results on the holistic model. The grey markers are the user-given constraints and the smaller black markers the constrained vertices. The currently active vertex is highlighted in yellow. Notice that only the region-based model provides predictable local control. Parameters $\beta$ (boundary constraint) and $\gamma$ (deformation constraint) were set to $0.5$ and $0.025$ respectively.*

## Acknowledgements

We would like to thank Moshe Mahler for sculpting and texturing the 3D human character in our figures and providing additional artistic support. We would also like to thank our actors.

## References

ALLEN, B., CURLESS, B., AND POPOVIĆ, Z. 2003. The space of human body shapes: Reconstruction and parameterization from range scans. *ACM Transactions on Graphics 22*, 3 (July), 587–594.

BERGERON, P., AND LACHAPELLE, P. 1985. Controlling facial expressions and body movements in the computer-generated animated short "Tonly De Peltrie". In *Computer Graphics, Advanced Computer Animation seminar notes*.

BLACK, M., AND YACOOB, Y. 1995. Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion. In *Proceedings of the Fifth International Conference on Computer Vision*, 374 –381.

BLANZ, V., AND VETTER, T. 1999. A morphable model for the synthesis of 3D faces. In *Proceedings of SIGGRAPH*, 187–194.

BUCK, I., FINKELSTEIN, A., JACOBS, C., KLEIN, A., SALESIN, D. H., SEIMS, J., SZELISKI, R., AND TOYAMA, K. 2000. Performance-driven hand-drawn animation. In *Proceedings of the 1st International Symposium on Non-photorealistic Animation and Rendering*, 101–108.

COOTES, T. F., EDWARDS, G. J., AND TAYLOR, C. J. 1998. Active appearance models. In *Proceedings of the 5th European Conference on Computer Vision*, 484–498.

DECARLO, D., AND METAXAS, D. 2000. Optical flow constraints on deformable models with applications to face tracking. *International Journal of Computer Vision 38*, 2 (July), 99–127.

DRYDEN, I. L., AND MARDIA, K. V. 2002. *Statistical Shape Analysis*. John Wiley & Sons.

EDWARDS, G. J., COOTES, T. F., AND TAYLOR, C. J. 1998. Face recognition using active appearance models. In *Proceedings of the 5th European Conference on Computer Vision*, 581–595.

EKMAN, P., AND FRIESEN, W. V. 1978. *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press, Palo Alto, CA.

FENG, W.-W., KIM, B.-U., AND YU, Y. 2008. Real-time data driven deformation using kernel canonical correlation analysis. *ACM Transactions on Graphics 27* (August), 91:1–91:9.

JOSHI, P., TIEN, W. C., DESBRUN, M., AND PIGHIN, F. 2003. Learning controls for blend shape based realistic facial animation. In *Proceedings of ACM SIGGRAPH/Eurographics symposium on Computer Animation*, 187–192.

LAU, M., CHAI, J., XU, Y.-Q., AND SHUM, H.-Y. 2009. Face poser: Interactive modeling of 3D facial expressions using facial priors. *ACM Transactions on Graphics 29*, 1 (Dec.), 3:1–3:17.

LAWRENCE, N. D. 2007. Learning for larger datasets with the gaussian process latent variable model. In *International Workshop on Artificial Intelligence and Statistics*.

LEWIS, J. P., AND ANJYO, K. 2010. Direct manipulation blendshapes. *Computer Graphics and Applications, IEEE 30*, 4 (July), 42 –50.

MATTHEWS, I., AND BAKER, S. 2004. Active appearance models revisited. *International Journal of Computer Vision 60* (November), 135–164.

MEYER, M., AND ANDERSON, J. 2007. Key point subspace acceleration and soft caching. *ACM Transactions on Graphics 26*, 3 (July), 74:1–74:8.

NG, A. Y., JORDAN, M. I., AND WEISS, Y. 2001. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, MIT Press, 849–856.

NISHINO, K., NAYAR, S. K., AND JEBARA, T. 2005. Clustered blockwise PCA for representing visual data. *IEEE Transactions on Pattern Analysis Machine Intelligence 27* (October), 1675–1679.

NOH, J.-Y., FIDALEO, D., AND NEUMANN, U. 2000. Animated deformations with radial basis functions. In *Proceedings of the ACM symposium on Virtual reality software and technology*, 166–174.

PENTLAND, A., MOGHADDAM, B., AND STARNER, T. 1994. View-based and modular eigenspaces for face recognition. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 84 –91.

PEYRAS, J., BARTOLI, A., MERCIER, H., AND DALLE, P. 2007. Segmented AAMs improve person-independent face fitting. In *British Machine Vision Conference*.

PIGHIN, F., HECKER, J., LISCHINSKI, D., SZELISKI, R., AND SALESIN, D. H. 1998. Synthesizing realistic facial expressions from photographs. In *Proceedings of SIGGRAPH*, 75–84.

SORKINE, O., COHEN-OR, D., LIPMAN, Y., ALEXA, M., RÖSSL, C., AND SEIDEL, H.-P. 2004. Laplacian surface editing. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, 175–184.

TENA, J. R., HAMOUZ, M., HILTON, A., AND ILLINGWORTH, J. 2006. A validated method for dense non-rigid 3D face registration. In *Proceedings of the IEEE International Conference on Video and Signal Based Surveillance*.

TENENBAUM, J. B., DE SILVA, V., AND LANGFORD, J. C. 2000. A global geometric framework for nonlinear dimensionality reduction. *Science 290*, 5500, 2319–2323.

TURK, M., AND PENTLAND, A. 1991. Eigenfaces for recognition. *Journal of Cognitive Neuroscience 3* (January), 71–86.

VLASIC, D., BRAND, M., PFISTER, H., AND POPOVIĆ, J. 2005. Face transfer with multilinear models. *ACM Transactions on Graphics 24*, 3 (Aug.), 426–433.

ZHANG, L., SNAVELY, N., CURLESS, B., AND SEITZ, S. M. 2004. Spacetime faces: High resolution capture for modeling and animation. *ACM Transactions on Graphics 23*, 3 (Aug.), 548–558.

ZHANG, Q., LIU, Z., GUO, B., TERZOPOULOS, D., AND SHUM, H.-Y. 2006. Geometry-driven photorealistic facial expression synthesis. *IEEE Transactions on Visualization and Computer Graphics 12* (January), 48–60.