On Modelling Nonlinear Shape-and-Texture Appearance Manifolds

C. Mario Christoudias Trevor Darrell

Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139
cmch,trevor@csail.mit.edu

Abstract

Statistical shape-and-texture appearance models employ image metamorphosis to form a rich, compact representation of object appearance. They achieve their efficiency by decomposing appearance into simpler shape-and-texture representations. In general, the shape and texture of an object can vary nonlinearly and in this case the conventional shape-and-texture mappings using Principle Component Analysis (PCA) may poorly approximate the true space. In this paper we propose two nonlinear techniques for modelling shape-and-texture appearance manifolds. Our first method uses a mixture of Gaussians in image space to separate the different parts of the shape and texture spaces. A linear shape-and-texture model is defined at each component to form the overall model. Our second approach employs a nearest-neighbor method to find a local set of shapes and images that can be morphed to explain a new input. We test each approach using a speaking-mouth video sequence and compare both approaches to a conventional Active Appearance Model (AAM).

1. Introduction

Statistical shape-and-texture appearance models [9, 2] use image metamorphosis to define rich, compact models of appearance. They are useful in a variety of applications including object recognition, tracking and segmentation [5, 13, 14]. Traditionally these methods use linear models (i.e., PCA) to represent the shape and texture of an object. There are many objects, however, that can exhibit nonlinear shape and texture variation, for which the conventional shape and texture mappings using PCA poorly approximate the true space. This is especially true of biological objects that can deform quite drastically, such as a hand or mouth, or whose texture can drastically vary across different examples (e.g., cats, dogs).

In this paper we investigate nonlinear techniques for modelling shape-and-texture appearance manifolds that exhibit a varying *topology* (i.e., a manifold that can have multiple parts or holes) or dimensionality. We test our methods using a speaking-mouth video sequence obtained from the AVTIMIT database [8]. In our experiments, the different parts of the space arise from varying mouth configurations (e.g., closed vs. open mouth). Different regions can take a different dimensionality in shape and texture. For example, an open mouth can have features associated with the teeth that are absent in a closed mouth.

We have implemented and compared two nonlinear techniques. The first technique uses a Gaussian mixture model to learn the nonlinear mouth appearance manifold. As demonstrated in the experiments, this method outperforms a simple linear model since it more tightly models the local variation of the nonlinear mouth appearance manifold. In general, it is difficult to know a priori the correct number of components to use in the mixture model. Also, a complex appearance manifold may require arbitrarily many mixture components, making such a model inefficient.

To overcome these limitations of a mixture-model approach we have implemented an example based shape and texture appearance model that computes a small neighborhood of example images and shapes that can be combined to explain a new input. In particular, we compute a morph between a neighborhood of examples on the manifold found using nearest neighbor, using a convex (or bounded) combination of the neighborhood's shape and texture to match the input image. Unlike the mixture model method, this approach makes no assumptions about the global structure of the manifold. It also lends itself more naturally to shape features having multiple dimensionality across examples in the database.

In our experiments we evaluate the performance of each of the above algorithms using a mouth sequence from a single speaker. We build each model using a small set of frames taken from the sequence and then fit each model to frames outside of the training set. For comparison, we also

build an AAM and show examples for which the conventional AAM fails and our methods succeed.

2. Related Work

Linear models of shape and texture have been widely applied to the modelling, tracking and recognition of objects [5, 9, 13]. Provided a set of example images, linear shape and texture appearance models decompose each image into a shape and texture representation and then model the variation of the data in these spaces using Principle Component Analysis (PCA). The shape of an object describes the object's geometry and is typically defined by a set of feature points that outline the object contours. The texture is the "shape free" representation of the object and is obtained by warping each image to a reference coordinate frame that is usually defined by the average shape computed from the training images.

The Active Appearance Model (AAM) [2] and Multidimensional Morphable Model (MMM) [9] are probably the most well known linear shape and texture appearance models. By decomposing appearance into separate shape and texture spaces they achieve a compact, expressive model of appearance, more powerful than pure intensity models defined with PCA (e.g. Eigenfaces [17]). As we show in this paper, these models are unable to faithfully represent the appearance of complex objects with nonlinear appearance manifolds, such as mouths, whose manifolds have parts and holes.

Many nonlinear models have been defined separately for shape and appearance [12, 3, 10]. Romdhani et. al. [12] use Kernel PCA to define a nonlinear shape model for representing shape across object pose. Cootes et. al. [3] show how a Gaussian mixture model can be used to construct a nonlinear active shape model that restricts its search to valid shapes on the object shape manifold, thus avoiding erroneous matches. A nearest neighbor algorithm is explored by Grauman et. al. [7]. In her work she defines an active shape model across body poses. Several authors have developed example based models of object appearance, including the metric mixtures approach of Toyama and Blake [16], however, these methods do not exploit shape and texture decomposition. Similarly, Murase and Nayar [10] present a manifold learning algorithm that maps out the space of images of an object imaged across different poses. To the author's knowledge this is the first work that explores nonlinear techniques for modelling shape and texture appearance manifolds. The only exception is the view-based AAM [4].

The view-based AAM defines a piecewise linear representation of the shape and texture appearance manifold in a very similar fashion to the Gaussian mixture model described in Section 3.1. The key differences between the Gaussian mixture model and the method described in [4] is that our method automatically learns the different regions of the manifold from the data and is not restricted to learning mixture components that vary across pose alone.

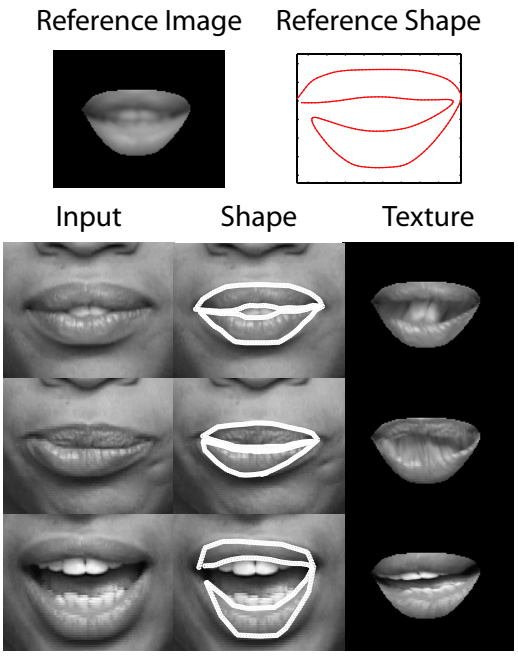

Figure 1. Linear models compute a texture space by warping each example to a single reference frame. Note the stretched region present in the closed mouth textures and that the inside of the mouth is lost in the texture of the open mouth.

3. Nonlinear Appearance Manifolds

The images of a complex object such as a mouth generally belong to a nonlinear appearance manifold with parts or holes as illustrated by Figure 1. This figure illustrates the shape and texture of example mouth images taken from the AVTIMIT database [8]. The average image and shape are displayed along with example textures and shapes of select prototype images.

Consider modelling the mouth appearance using a linear model such as an AAM. Figure 1 demonstrates the difficulty with modelling the mouth using a linear method. In particular, notice the stretched region in the texture of a closed mouth and that the inside of the mouth is lost in the texture of an open mouth. These artifacts cripple the computed model; in general, linear methods have difficulty modelling the full range of mouth appearance. Such artifacts are a result of the varying topology of the appearance manifold of this object—some features (or surfaces) are visible in certain images but not in others (e.g., teeth). Intuitively, this is seen by the fact that there exist sets of mouth configurations for which the same parts of the mouth are visible in each set.

In addition to varying topology, the shape-and-texture

Figure 2. First five nearest neighbors computed with our algorithm on a database of 100 mouth images.

spaces of nonrigid object classes have varying dimensionality across examples. Once again, consider the mouth images of Figure 1. The presence of teeth in the open mouth introduces new shape features that are absent from the image of the closed mouth. Allowing for varying shape dimensionality results in a more expressive and accurate model of appearance.

Below we present two nonlinear models for modelling shape-and-texture appearance manifolds. The first method takes into account the varying topology in image space by fitting a mixture model to the PCA coefficients of the image data. As the number of components necessary is difficult to know a priori (or estimate via cross-validation) and the number of components may increase substantially with the complexity of the appearance manifold, we also develop an alternative nearest-neighbor model. Unlike the first method, the nearest-neighbor model makes no assumptions of the global structure of the appearance manifold. Instead, it looks at local neighborhoods on the manifold that are assumed to belong to the same region of the topology. While in principle it is possible to extend both of the nonlinear approaches below to include varying shape dimensionality, the nearest-neighbor model lends itself more naturally to this task.

3.1. Mixture Manifold

In this section we develop a Gaussian mixture model for representing a shape-and-texture appearance manifold. To begin, let $\mathbf{x}_i$ and $\mathbf{s}_i$, $i = 1, ...n$, be a set of prototype images and their corresponding shapes. As in [2], we define shape to be

$$\mathbf{s} = \langle x_1, x_2, .., x_k, y_1, y_2, ..., y_k \rangle^T, \tag{1}$$

where $\langle x_i, y_i \rangle$ is a two dimensional image feature point.

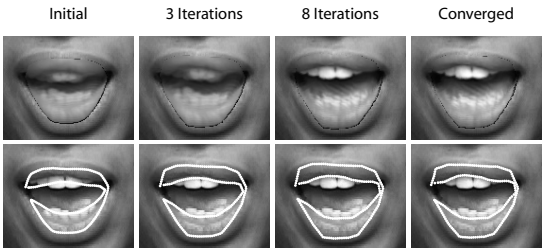

Figure 3. A convincing reconstruction of the shape and texture of an input mouth image is computed in a few iterations using the gradient descent algorithm of the nearest neighbor model.

Assuming that the different regions of the object appearance manifold are well approximated as linear, the underlying structure of the manifold can be explained using a Gaussian mixture model. More specifically, we wish to learn the underlying probability distribution $p(\mathbf{x})$ of the object appearance manifold. Using a Gaussian mixture, this distribution is found as

$$p(\mathbf{x}) = \sum_j p(\mathbf{x}|j)p(j), \tag{2}$$

where $j = 1, ..., m$ represents the jth mixture component. Given the prototype images, $\mathbf{x}_i$, the Gaussian mixture is learned using Expectation Maximization (EM). See [1] for details. To make the computation of (2) computationally tractable, we use PCA to reduce the dimensionality of the images and then approximate $p(\mathbf{x})$ by computing a mixture model over the PCA coefficients. Namely, we approximate $p(\mathbf{x})$ as

$$p(\mathbf{x}) \approx \sum_j p(\mathbf{b}|j)p(j), \tag{3}$$

where $\mathbf{b}$ are the PCA coefficients of the input image found as

$$\mathbf{b} = \mathbf{P}^+(\mathbf{x} - \mu), \tag{4}$$

the columns of $\mathbf{P}$ are the $d < n$ principle axis computed with PCA and μ is the mean image. More generally, we could use PPCA [15], which would allow for different dimensional sub-spaces.

Each component of the mixture model defines a region of image space for which the same parts of the mouth are visible. Consequently, each component is associated with its own shape-and-texture space. Assuming that the shape and texture varies linearly in each component, we can model the local shape-and-texture variation using a linear deformable model. In particular, at each component, we compute an AAM [2] using the examples that lie under the support of the component's Gaussian. Since each Gaussian has infinite support, we segment the manifold according to the mixture model by truncating each Gaussian using a threshold. We consider an example to be under the support of a Gaussian if it is less than three standard deviations away from the mean.

Figure 4. Multidimensional shape representation used by the nearest-neighbor model. Each example image is labelled with varying feature sets according to what parts of the mouth are visible. Three examples are shown: (left) with only lip features, (middle) with lip and top teeth features, (right) with lip, top and bottom teeth features.

The Gaussian mixture model defines a piecewise linear model of shape and texture, each region of the topology modelled using a separate AAM. To analyze a new example image, we independently fit each local AAM to the example and retain the fit with the lowest error. Note the model provides a nonlinear mapping of shape and texture that observes the varying topology of the manifold. In particular, the components of the Gaussian mixture model map out the different regions and holes of the nonlinear appearance manifold. Using this model, an input image is mapped to a set of local shape-and-texture coefficients associated with the region in image space that best explains the new input. Given $p(\mathbf{x})$, we are less likely to map an image to a point off of the manifold (e.g. a non-mouth), since $p(\mathbf{x})$ equips the model with detailed knowledge of the manifold structure.

The above model provides a concise representation of the shape and texture of a nonrigid object class whose appearance manifold has a varying topology. This model requires knowledge of m, however, which may be difficult to estimate, and may be arbitrarily large for complex manifolds. Part of the reason for this, is that it assumes that each region of the topology is locally linear. In general, each part of the manifold can have arbitrary shape and thus we expect this model to perform poorly when this occurs. In the next section, we present an alternative nearest-neighbor algorithm that relaxes the above assumptions.

3.2. Nearest-Neighbor Model

In this section, we present our nearest-neighbor shape-and-texture appearance model. Instead of modelling each region explicitly, the nearest-neighbor model provides an implicit representation of the object appearance manifold. Specifically, this model focuses on local neighborhoods of the manifold defined by k examples. In this region it is assumed that the same parts of the nonrigid object are visible. Given the local neighborhood, the shape and texture of a new input is found by taking bounded combinations of the shape and texture of the k nearest-neighbor examples. Therefore, given a new image, we wish to find a local neighborhood observing the above properties, whose shape and texture best explain the input.

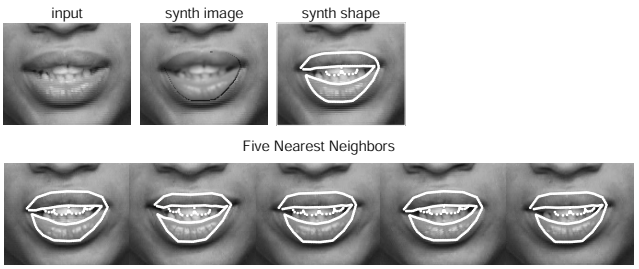

Figure 5. Shape intersection algorithm used by the nearest-neighbor model. To compute the shape of the input, the shape of the nearest neighbors is intersected and the shape features common to all examples is used.

We use nearest-neighbor search to find a set of examples on the manifold whose appearance most closely approximate that of the input. Given a novel input, $\mathbf{x}_s$, we compare it to each image, $\mathbf{x}_i$, of the prototype set to compute its k nearest neighbors. Although we use an exhaustive search there exist fast methods for computing approximate nearest neighbors [6] that we leave for future work. In our algorithm, we compute proximity using Euclidean distance in pixel space. We compute the distance,

$$d(\mathbf{x}_s, \mathbf{x}) = \|\mathbf{x}_s - \mathbf{x}\|^2, \tag{5}$$

between $\mathbf{x}_s$ and each prototype image and retain the k examples having smallest distance. Figure 2 displays the results of this nearest-neighbor algorithm on a database of 100 images of a single subject's mouth taken from the AVTIMIT database. The nearest neighbors of a novel input appear to form a local neighborhood in image space.

The shape and texture of an input image are computed by taking a convex combination of the shape and texture of its k nearest neighbors. Let $\mathbf{x}_j$ and $\mathbf{s}_j$, $j = 1, ..., k$ be the k nearest neighbors of the input and their associated shapes. The texture of each example is computed as

$$\mathbf{t}_j = \mathbf{x}_j \circ W(\mathbf{s}_j, \mathbf{s}_{ref}), \quad j = 1, ..., k, \tag{6}$$

where $\circ$ denotes the warping function, $W()$ is a function that computes the piecewise affine correspondence between two images given their shape [2], and $\mathbf{s}_{\text{ref}}$ is the reference shape of the local neighborhood defined to be the mean of the example shapes,

$$\mathbf{s}_{\text{ref}} = \frac{1}{k} \sum_j \mathbf{s}_j. \tag{7}$$

Given the k nearest neighbors of the input, we search over bounded combinations of their shape and texture that best match the input by minimizing the following error objective function,

$$E(\mathbf{x}_s, \mathbf{b}, \mathbf{c}) = \|\mathbf{x}_s \circ W(\mathbf{s}_m(\mathbf{c}), \mathbf{s}_{\text{ref}}) - \mathbf{t}_m(\mathbf{b})\|^2, \tag{8}$$

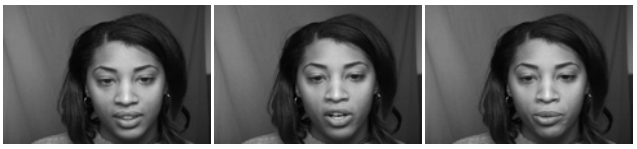

Figure 6. Video sequence taken from the AVTIMIT database [8] used to train and test our models. Select frames from this sequence are shown.

Variables	Perturbations
x, y	$\pm 5\%$ and $\pm 10\%$ of the height and width of the reference shape
θ	± 5, ± 15 degrees
scale	$\pm 5\%$, $\pm 15\%$
c_{1-k}	± 0.25, ± 0.5 standard deviations

Table 1. Perturbation scheme used to train the local linear models of the Gaussian mixture model and used by the AAM. [14]

where

$$\mathbf{t}_m(\mathbf{b}) = \sum_j b_j \mathbf{t}_j,$$
$$\mathbf{s}_m(\mathbf{c}) = \sum_j c_j \mathbf{s}_j,$$

and b_j, c_j take values on the closed interval $[\alpha, \beta]$. Note that α and β restrict the search to a bounded region of the manifold containing the k nearest-neighbor examples. If $\alpha = 0$ and $\beta = 1$ then the search is restricted to the convex hull of the example shape-and-texture vectors. This restriction results in a compact representation of the manifold and assures that we match an input to a point on the manifold. In our experiments, we bound the mixture weights to lie on the closed interval [-1.5,1.5].

We minimize the objective function (8) using gradient descent. Figure 3 displays an example match using the above algorithm. The algorithm is able to generate a convincing reconstruction of the mouth from the shape and texture of the nearest-neighbor examples.

It is straightforward to extend the nearest-neighbor model to handle multiple shape dimensionality. With this representation a shape vector, $\mathbf{s}^M$, is defined as

$$\mathbf{s}^M = \langle x_1, x_2, ..., x_M, y_1, y_2, ..., y_M \rangle . \qquad (9)$$

In the above representation, each shape has dimensionality $2M$. This multidimensional shape representation is illustrated by Figure 4. In the nearest-neighbor model we associate each prototype image with a shape vector that has dimensionality according to what is visible in the image. When computing nearest neighbors, we intersect the shapes of the neighborhood examples and use the shape features common to all examples to match the novel input. This process is illustrated by Figure 5. The use of multiple shape dimensionality results in a more expressive and accurate appearance model.

4. Experiments

To evaluate each algorithm, we used a mouth sequence of a single person taken from the AVTIMIT database [8]. This sequence contained a total of 2300, 720×480 grayscale images; select frames from this sequence are displayed in Figure 6. We randomly selected 100 frames and manually labelled each frame with mouth features (see Figure 1). Using the labelled features, we cropped each image about the center of the mouth using a 111×139 window to form our training set. Using this training set we constructed the Gaussian mixture deformable model discussed in Section 3.1 and an Active Appearance Model [2]. The same 100 frames, along with the multidimensional shape feature vectors displayed in Figure 4, were used by the nearest-neighbor model discussed in Section 3.2.

We built the Gaussian mixture model using a three dimensional subspace of the image data computed with PCA, retaining 56 % of the total model variance, and with $m = 5$ mixture components. We found these parameters to work well in our experiments. Using a three dimensional subspace also allowed us to visualize our models. To compute the Gaussian mixture, we used the NetLab library [11]. The local AAMs constructed in the Gaussian mixture model and the single AAM models were constructed using the parameters listed in Table 1. In each local AAM, as well as the single AAM, 95 % of the model variance was retained by the combined shape-and-texture space. In our experiments we evaluated the nearest-neighbor algorithm for varying values of k. The value used is made explicit in each experiment. We also allowed each interpolation weight to take values between -1.5 and 1.5, allowing it more freedom to extrapolate from the data. These values were found empirically from the training data.

In our experiments, we assume that the location of the mouth is coarsely initialized by an external mouth detector. Both the Gaussian mixture model and the AAM optimize for location during model search and therefore require only approximate initialization of the mouth location. We refine the mouth location estimate in the nearest-neighbor model by finding the nearest neighbor using the input location and then computing a normalized cross correlation between the nearest neighbor and same-sized patches in the input image centered about locations in an 11×11 search window about the initial center. We reset the center of the mouth to the location having the highest correlation score and repeat this process until convergence or the maximum number of iterations is reached. In our experiments, we found this algorithm typically converged in a few iterations.

In the following section, we perform both a qualitative and quantitative comparison of each of the non-

linear algorithms and compare them against the baseline AAM approach, each model constructed as specified above. We perform this comparison over 540 mouth images outside of the training set taken from the mouth sequence of Figure 6. The nearest-neighbor model's dependency on k is also evaluated using this test set. Finally, we show an example for which the multidimensional shape representation improves model accuracy. A description of additional results and experiments can be found at *<http://groups.csail.mit.edu/vision/vip/nlam.htm>*.

5. Results

Three images taken from the 540-image test set along with the synthesized texture and shape generated by each model are displayed in Figure 7. The RMS fit error is also provided above each fit. In this experiment, the nearest-neighbor model has $k = 10$. The first test image is modelled well using all three models. Comparing the RMS error of each fit, however, both the Gaussian mixture and nearest-neighbor models outperform the AAM. The synthesized texture of the AAM is also quite blurred. The next two examples reveal scenarios for which the single AAM model fails and the nonlinear methods succeed. In particular, the AAM has difficulty modelling any images whose geometry is very different from the model reference image. This is seen in the case of the open mouth image, where the inside of the mouth is poorly represented in the texture space of the linear model (see Figure 1).

A quantitative comparison of each model is provided by Figure 8. In the figure, a Root-Mean-Square error box plot is shown for each approach computed over the 540-image test set. Both the Gaussian mixture model and the nearest-neighbor model do the same or significantly better than the single AAM throughout the test sequence. The error box plot shows that with $k = 10$ the nearest-neighbor algorithm outperforms each approach on a whole (different values of k are considered next). The noteworthy performance of the nearest-neighbor model is expected since it makes the fewest assumptions about the underlying structure of the appearance manifold.

The poor performance of the single AAM on the mouth sequence is a direct result of the simplicity of the model. This model assumes a single texture space over the mouth appearance manifold. Since the appearance manifold has varying topology, a global texture space is ill-defined; the appearance variation of the mouth is not well represented using a single reference coordinate frame. This point was demonstrated by Figure 1 in Section 3. Also, the single AAM has no knowledge of the local structure of the manifold and can therefore converge to non-mouth images. Each of these properties contribute to the AAM's poor performance in modelling the appearance of the mouth. The nonlinear techniques of Section 3 provide shape-and-texture

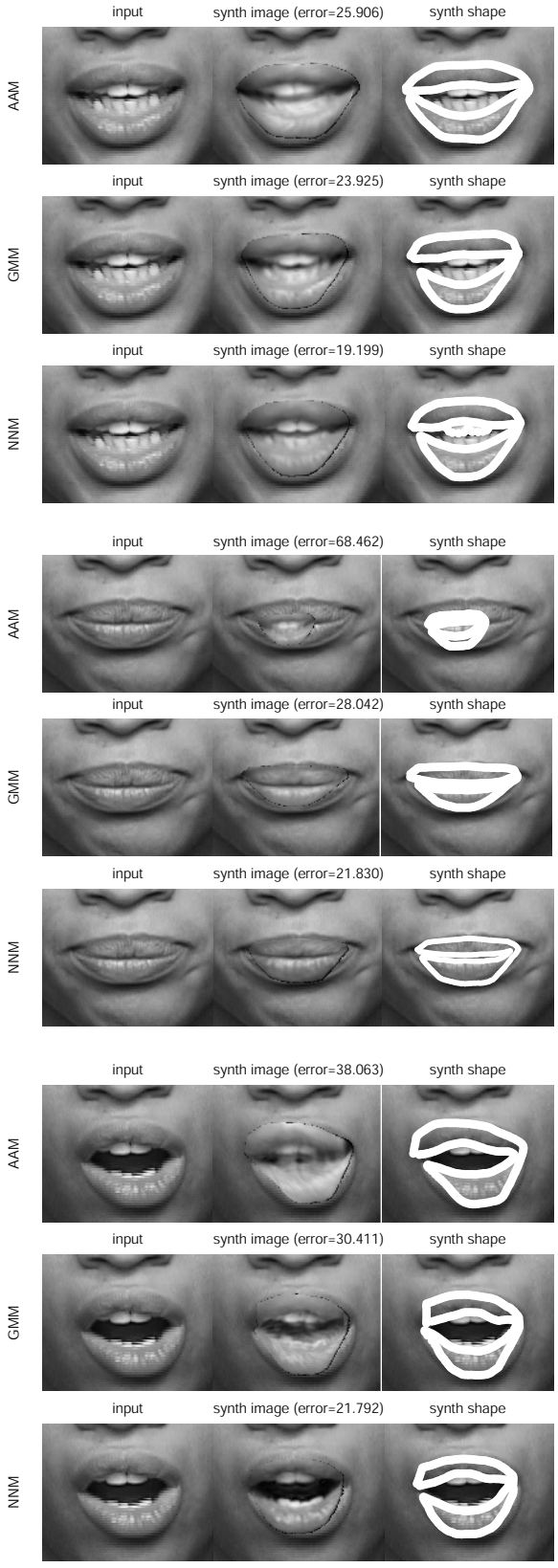

Figure 7. Qualitative comparison between each method and a baseline linear model. The input, synthesized shape and texture, computed with each model, is shown for each example. The AAM has difficulty modelling the full range of mouth appearance. The last two examples illustrate scenarios where the AAM fails and our methods succeed. These examples contain regions of the mouth that are absent from either the reference image or input mouth image and thus the AAM cannot faithfully represent their appearance.

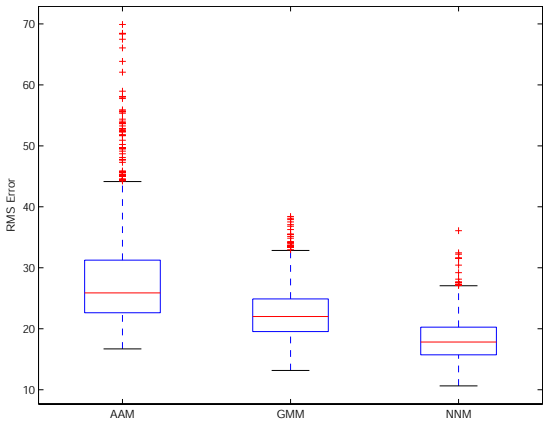

Figure 8. Quantitative comparison between each method and a baseline linear model. A box plot of the RMS error of each model evaluated over 540 test mouth images is shown. In the plot, the horizontal lines of each box represent the top quartile, median and bottom quartile values, the whiskers give the extent of the rest of the data and the red crosses label data outliers. Of the three methods, the AAM displays the worst performance and the nearest-neighbor model performs the best.

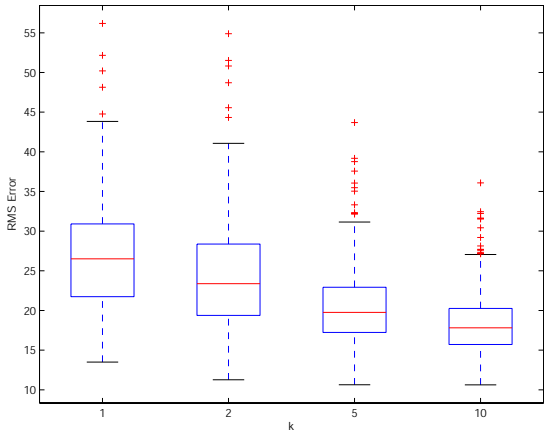

Figure 9. Quantitative comparison of the nearest-neighbor method for different k. The model performs better for increasing values of k. As the number of examples increases the model is provided with more degrees of freedom and can therefore match the input more closely.

mappings that take into account the varying topology of the mouth appearance manifold and therefore are able to faithfully represent the full range of mouth appearance variation.

Next, we evaluate the performance of the nearest-neighbor algorithm for different k values. Figure 9 displays an RMS error box plot for the nearest-neighbor model evaluated over the 540 test frames with $k = 1, 2, 5, 10$. The figure illustrates that the model performs better for increasing values of k. This verifies our intuition that morphing between examples does better than simply taking the nearest neighbor. As the number of examples increases the model is provided with more degrees of freedom and can therefore match the input image more closely. Of course taking too large a value of k complicates the search and can lead to poor performance.

Finally, we consider how the use of a multidimensional shape representation can improve model-fitting accuracy. For this experiment, we use a simplified version of the multidimensional shape representation displayed in Figure 4 that contains only mouth contours and features of the top middle teeth. This multidimensional shape representation divides the mouth data set into two equivalence classes, each containing images with and without teeth. We compare performing nearest-neighbor over the entire mouth training set verses within each class separately. The results of this experiment for an example mouth image are displayed in Figure 10 with $k = 5$. The figure shows the synthesized mouth image using the single-class and dual-class nearest-neighbor methods. The results for the single-class and each class of the dual-class are shown along with the computed nearest neighbors. Taking the shape and texture with the smallest fit error as the result of the dual-class model, Figure 10 shows that the dual-class model outperforms the single class nearest-neighbor method.

The difference in performance can be explained by observing the nearest neighbors computed with each model. The nearest neighbors of the single-class model are less like the input than those found with the dual-class model for the class containing mouth images with teeth. This is especially true of the last two nearest neighbors computed with the single class model. This simple experiment demonstrates how a multidimensional shape representation can be used to guide the model matching process to increase fitting accuracy.

6. Conclusions and Future Work

We have presented two nonlinear techniques for modelling the shape-and-texture appearance manifolds of complex objects whose appearance manifold has a varying topology consisting of parts and holes. We showed how a piecewise linear model of a shape-and-texture appearance manifold can be defined using a Gaussian mixture model. We also provided a nearest-neighbor model that generalizes well to complex manifolds, offers a compact representation of the manifold and allows for varying feature sets. In particular, with this technique a new input is analyzed by morphing a local set of examples that belong to a convex or bounded region of the manifold.

We evaluated the performance of each algorithm using the AVTIMIT database, where we built a shape-and-texture appearance model of the mouth. We compared each approach to a baseline linear model and showed examples

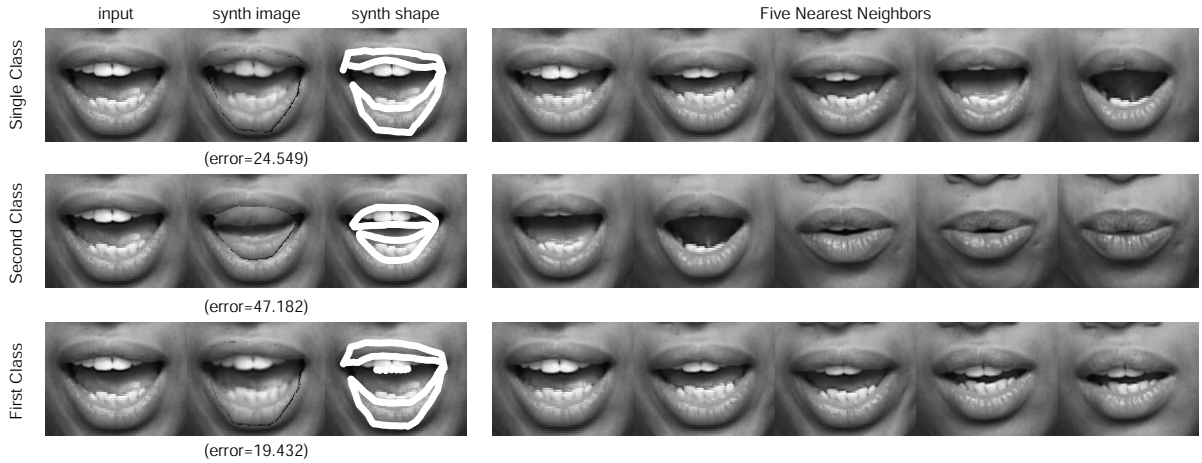

Figure 10. Qualitative comparison between a single and multi-dimensional shape representation. The first appearance model (top row) contains only lip features while the second contains both lip and teeth features for the top middle teeth. The images of the second model are separated into two equivalence classes with and without teeth. Taking the shape and texture with the smallest fit error as the result of the dual-class model, the dual-class model outperforms the single class nearest-neighbor method. The difference in performance is explained by the more accurate nearest neighbors found by the dual-class model.

where the conventional method fails and our methods succeed. We demonstrated that linear models poorly represent the appearance of complex objects such as mouths and that our methods are able to define a convincing shape-and-texture mouth appearance model by taking into account the varying topology of the mouth appearance manifold. Interesting avenues of future work include the construction of a person-independent mouth deformable model, the use of Locality Sensitive Hashing [6] as an alternative, more efficient method for computing nearest neighbors and the consideration of different distance metrics that are less sensitive to lighting, location, orientation and scale.

References

[1] Christopher M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[2] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *Lecture Notes in Computer Science*, 1407:484–98, 1998.

[3] T. F. Cootes and C. J. Taylor. A mixture model for representing shape variation. In *Image and Vision Computing*, volume 17, pages 567–574, 1999.

[4] T. F. Cootes, G. V. Wheeler, K. N. Walker, and C. J. Taylor. View-based active appearance models. *Image and Vision Computing*, 20:657–664, 2002.

[5] G. Edwards, C. Taylor, and T. Cootes. Interpreting face images using active appearance models. In *3rd International Conference on Automatic Face and Gesture Recognition*, pages 300–305, 1998.

[6] A. Gionis, P. Indyk, and R. Motwani. Similiarity search in high dimensions via hashing. In *25th International Conference on Very Large Data Bases*, pages 518–529, 1999.

[7] K. Grauman and T. Darrell. Fast contour matching using approximate earth mover's distance. In *CVPR*, June 2004.

[8] T. J. Hazen, K. Saenko C. H. La, and J. Glass. A segment-bsaed audio-visual speech recognizer: Data collection, development, and initial experiments. In *Proc. ICMI*, 2005.

[9] Michael J. Jones and Tomaso Poggio. Multidimensional morphable models. In *ICCV*, pages 683–688, 1998.

[10] H. Murase and S. Nayar. Visual learning and recognition of 3-d objects from appearance. *IJCV*, 14(1):5–24, 1995.

[11] Ian T. Nabney. *NETLAB Algorithms for Pattern Recognition*. Springer, 2004.

[12] S. Romdhani, S. Gong, and A. Psarrou. A multi-view nonlinear active shape model using kernel pca. In *British Machine Vision Conference*, pages 483–492, 1999.

[13] Stan Sclaroff and John Isidoro. Active blobs. In *ICCV*, Mumbai,India, 1998.

[14] M. B. Stegmann. Analysis and segmentation of face images using point annotations and linear subspace techniques. Technical report, DTU, 2002.

[15] M. Tipping and C. Bishop. Probabilistic principal component analysis. Technical report, NCRG/97/010, 1997.

[16] K. Toyama and A. Blake. Probabilistic tracking with exemplars in a metric space. *International Journal of Computer Vision*, 48(1):9–19, 2002.

[17] M. Turk and A. Pentland. Eigen faces for recognition. *Journal of Cognitive Neuroscience*, 3(1), 1991.