Parameterisation of 3D Speech Lip Movements

James D. Edge, Adrian Hilton and Philip Jackson

Centre for Vision, Speech and Signal Processing,
University of Surrey, Guildford, UK.

j.edge@surrey.ac.uk, a.hilton@surrey.ac.uk, p.jackson@surrey.ac.uk

Abstract

In this paper we describe a parameterisation of lip movements which maintains the dynamic structure inherent in the task of producing speech sounds. A stereo capture system is used to reconstruct 3D models of a speaker producing sentences from the TIMIT corpus. This data is mapped into a space which maintains the relationships between samples and their temporal derivatives. By incorporating dynamic information within the parameterisation of lip movements we can model the cyclical structure, as well as the causal nature of speech movements as described by an underlying visual speech manifold. It is believed that such a structure will be appropriate to various areas of speech modeling, in particular the synthesis of speech lip movements.

Index Terms: speech synthesis

1. Introduction

Synthetic talking heads are becoming increasingly popular across a wide range of applications: from entertainment (e.g. Computer Games/TV/Films) through to natural user interfaces and speech therapy. Increasingly the techniques used in synthesis are data-driven, a trend related to the increasing number of techniques which facilitate the capture of articulatory movements (e.g. motion-capture, stereo-photogrammetry etc.) In this paper we present a method for parameterising lip movements captured using a dynamic stereo-capture system, and discuss the properties of the underlying visual speech manifold and how these can aid the application of speech synthesis.

When trying to understand the process of speech production it is useful to project captured data, whichever form that may take (e.g. 3D markers, 2D contours, lip aperture/protrusion etc.), into a lower dimensional space which provides greater fidelity in the underlying structure of this task. This projection fulfils two main roles: firstly, to identify several latent parameters which better describe the underlying processes; and secondly to remove the components of the original signal which can be regarded as insignificant or as noise.

The most common method for parametrising speech data (and motion data in general), has been *Principal Component Analysis* (PCA.) This method forms an orthogonal basis by calculating the eigenvectors (principal components) of the covariance matrix. The eigenvalues corresponding to these basis eigenvectors hold the variance that each vector accounts for, and thus a reduced set of vectors can be retained holding the majority of the variance in the original data. PCA has been used as an underlying parameterisation for many applications, principally tracking/image synthesis [1, 2] and data-driven animation [3]. Similar techniques applied to the symmetric distance matrix are termed *Multi-Dimensional Scaling* (MDS), or *Principal Coordinate Analysis* (PCO, [4].)

Another linear method, particularly popular in source separation, is *Independent Component Analysis* (ICA.) This technique applies the concept of statistical independence to recovering an underlying parameterisation. This is a stronger assumption than that of PCA, but does not produce an orthogonal basis. ICA has been applied to recovering a basis for speech movements, and for identifying and separating speech-task and emotion-task parameters. In [5] ICA is employed to separate emotional parameters from speech control parameters, which are later recombined in a synthesis framework. The success of such techniques is arguable. Problematically, ICA decompositions tend to produce non-smooth trajectories which are generally inappropriate for tasks such as synthesis.

More recently a host of non-linear methods have come to the fore, e.g. [6, 7]. These methods mostly unwrap the data according to an approximation of geodesic distances across an underlying manifold structure. The main problem with PCA is that where non-linear relationships are evident in the underlying data, e.g. when projecting onto a pair of basis vectors a curve is plotted [1], this curve may be better described by a single non-linear vector recovered using a method such as Isomap. Unfortunately, many non-linear methods are not generative (i.e. you can project into the latent space, but not back to the data domain), and those which are often cannot deal with very large datasets. Furthermore, it is only worthwhile performing a non-linear parameterisation if the recovered space is more descriptive of the underlying data than simpler linear methods.

In [8] *Gaussian-Process Latent Variable Models* (GPLVM), a non-linear and generative dimensionality reduction technique is used to form a low dimensional manifold for motion data. Trajectories in the manifold can then be fitted to real data, such as video. This technique demonstrates one of the main problems with non-linear methods, as in some cases the optimisation of the projective space leads to discontinuities in motion trajectories. *Gaussian Process Dynamic Models* (GPDM) [9] have been designed to remove discontinuities in the generated projection from GPLVM. However, even in cases where dynamics or back-constraints have been used to maintain the contiguity of the projected trajectory, we may have cases where identical frames are projected to different locations in the nonlinear space.

In this paper we address the problem of finding a representation of speech lip movements captured using 3D stereo capture technology. It is often considered that the only role of this stage is to take high-dimensional data and map it into a low-dimensional space, i.e. the simplification of the task space is the end in itself. However, depending upon the end application there may be other qualities which define how good an embed-

[1] Recovered nonlinearities in PCA are often referred to as the ”horseshoe problem”.

ded space is. For the purposes of our work (i.e. synthesis) we can define several factors which are important:

- *minimal* - as with all dimensionality reduction techniques we aim for a greatly reduced set of variables.
- *interpretable* - the variables should have a logical interpretation when relating back to the physical process of speech production.
- *smooth* - projected speech trajectories should be smooth, i.e. there is some geometric continuity to movement through the task space. Smooth trajectories are more easily synthesised than those with random fluctuations.
- *generative* - at least for the task of synthesis, it is important that trajectories within our simpler task space can be projected back to the original data domain.

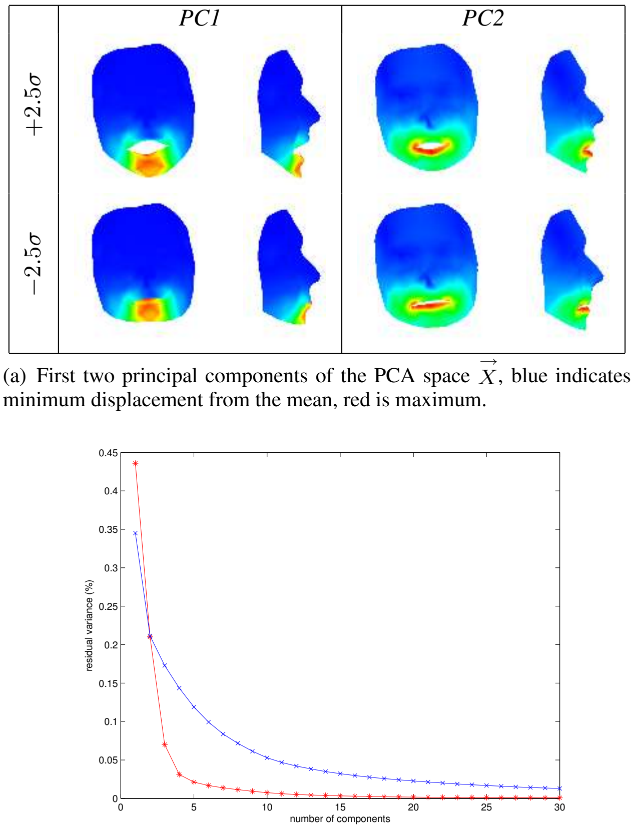
In the following sections we introduce our technique for constructing a low-dimensional task space which satisfies the properties described above. In Section 3 the construction of the task space and the associated geometric properties of the embedded speech manifold are described. In Section 4 we describe techniques for synthesising trajectories and recovering the mapping back to the original data domain. In Section 5 we briefly discuss clustering of phonetic classes on the generated speech manifold. Finally, in Section 6 we describe directions of future work and the application of our techniques to the problem of speech synthesis.

2. Data Capture

The data used in this paper consists of 8 minutes of 3D geometry captured using a commercial stereo face capture system (see [10].) Facial geometry is recovered using 2 camera stereo pairs (left/right); an infra-red speckle pattern is projected onto the face and stereo-photogrammetry is used to recover the 3D surface. A further two colour cameras also capture the appearance of the subject's face. The cameras we use operate at 60Hz, and audio is also captured synchronously. The raw data from this system is initially unregistered, i.e. given a point on the surface of the face in the initial frame we do not know the corresponding point in any of the following frames. To solve this correspondence issue, blue markers (both point and contour markers are used) are painted on the subjects face and tracked frame-to-frame in the colour images, these markers are used as the basis of a dense registration of all points in the geometry across all frames in the sequence (details are described in [11].) Because the colour appearance frames contain the blue markers, the processing described in the rest of this paper is performed on the recovered geometry alone.

Consonants	p	72	b	79	m	99	ch	31
	jh	34	s	313	z	109	sh	41
	zh	20	f	69	v	58	th	28
	dh	81	k	133	g	39	t	241
	d	187	r	136	w	68	n	254
	ng	28	hh	29	l	170	y	62
Vowels	aa	24	ae	85	ah	48	ao	49
	aw	23	ay	57	ax	299	ea	26
	eh	73	ey	65	ia	22	ih	198
	iy	126	oh	62	ow	47	oy	24
	ua	23	uh	30				

Table 1: Frequency of English phonemes in the captured data.



(a) First two principal components of the PCA space $\overrightarrow{X}$, blue indicates minimum displacement from the mean, red is maximum.

(b) Residual variance with increasing numbers of components: blue shows residual for $\overrightarrow{X}$; red shows residual for $\overrightarrow{Y}$.

Figure 1: Constructing a model of speech lip movement; the basis $\overrightarrow{X}$ is an initial *PCA* projection of the original data, $\overrightarrow{Y}$ is the reduced speech task space.

The 8 minutes of data consists of a single native British English subject speaking sentences selected from the TIMIT corpus. Sentences were selected to get a good sampling of all phonemes, see table 1. Unfortunately, due to data storage and processing issues it was not possible to get a good sampling of phonemes across all possible contexts. However, it is considered that the data does provide a good sampling of lip dynamics, the description of which is the purpose of this paper.

3. Parameterising Speech Lip Movements

3.1. Constructing a Task Space Embedding for Speech

In constructing a task space for the data described in the previous section, the registered geometry alone is used. The data consists of a sequence of frames, F, where the i^{th} frame $F_i = \{x_0, y_0, z_0, \ldots, x_i, y_i, z_i, \ldots, x_n, y_n, z_n\}$. A noise reduction step employing standard PCA directly on F is used to filter out low variance modes. By applying PCA we get a set of basis vectors, $\overrightarrow{X}$. The *EM* method for computing principal components [12] is used here due to the size of the data matrix, F, which holds $28,833$ frames $\times$ $12,784$ xyz coordinates. The first 100 basis vectors are computed, with the first 30 holding over 99% of the recovered variance. The percentage of the total variance accounted for will be lower, but the scree-graph shows that the important features of F are compressed in only a few dominant components (i.e. $\sim 95\%$ in the first 10 components shown in fig. 1(b), and $\sim 99\%$ in the first 30 components indicating a flattening of the scree-graph.)

F can be projected onto the basis $\vec{X}$ to produce the parameterisation F^x, i.e. $F_i \times \vec{X} \rightarrow F_i^x$. Broadly, the 1^{st} component of $\vec{X}$ can be described as jaw opening, the 2^{nd} is lip rounding/protrusion, and the 3^{rd} raises/lowers the upper lip. Lower variance components are not as easily contextualised in terms of observed lip-shape qualities. Figure 1(a) shows the first two principal components of $\vec{X}$.

A further projection of the data is performed to maintain the relative dynamic properties of lip movements during speech, i.e. in contrast to the formation of $\vec{X}$ which only takes into account static lip shapes. The first derivative for each frame is approximated as $F_i^{x\prime} = F_i^x - F_{i-1}^x$ (the parametric displacement of the lips in $1/60^{th}$ of a second.) Each pair $\{F_i^x, F_i^{x\prime}\}$ describes a distinct state in the physical space of lip movement. As the first derivative is at a different scale the parameters need to be normalised such that F_i^x does not dominate over $F_i^{x\prime}$. Thus, a matrix $G = \{\alpha(F_i^x - \mu), \beta(F_i^{x\prime} - \mu')\}$, where μ and μ' are the respective means of F^x and $F^{x\prime}$, is constructed where all parameters are scaled to the range $F_i^x, F_i^{x\prime} \in [-1, 1]$. This matrix is now processed in a manner similar to MDS/PCO[2]. A symmetric distance matrix D is formed where each element D_{ij} is the euclidean distance between G_i and G_j, i.e. $D_{ij} = \sqrt{(G_i - G_j)^2}$. The matrix D is then decomposed using another iteration of PCA forming a basis $\vec{Y}$, so for each of the initial frames F_i we have a corresponding projection into the task space F_i^y. The first three dimensions of $\vec{Y}$ account for $\sim 93\%$ of the recovered variance in D. Figure 1(b) shows that the eigenvalues corresponding to $\vec{Y}$ are more tightly clustered in the first several components than those of $\vec{X}$. Note that the F^y describe the relative position of the G_i, but there is no direct transformation from a point in $\vec{Y}$ back into $\vec{X}$ (i.e. because F^y is a relative embedding.)

It can be seen in fig. 2(a) that the first 3 dimensions of the projected F^y form a manifold $\mathcal{M}$ embedded within the task space $\vec{Y}$ descriptive of lip movements during speech production. Some useful properties of $\mathcal{M}$ and speech trajectories in this space are described in the following sections.

3.2. The Structure of the Speech Manifold

As can be seen from fig. 2(a), the manifold, $\mathcal{M}$, constructed as described in the previous section, forms a paraboloid-type structure in 3D (the first 3 dimensions of $\vec{Y}$.) $\mathcal{M}$ is symmetric about a plane which coincides with zero-velocity, and either side of this plane we have the opening and closing halves of a speech cycle. In the opening half of the cycle lips move from closed states towards more open states, and in the closing half the lips move from more open states towards more closed states. The vector orthogonal to the zero-velocity plane corresponds to maximum change in velocity (i.e. maximum acceleration) and parallel to the zero-velocity plane we find a vector which corresponds to maximum change in shape (i.e. lip shape transitions from closed → open, see fig. 2(c).) We can define non-linear vectors which globally describe maximum change in lip shape, S, and maximum change in lip velocity, V. Of course the evident non-linear structure of the speech manifold means that it is more appropriate to define a local tangent space for each F_i^y, with s_i defining a local shape-like vector, and v_i defining a local velocity-like vector, with $s_i \times v_i$ defining the local orientation of the manifold (i.e. direction of thickness.)

Traces of speech movements will produce elliptical paths on $\mathcal{M}$. This is natural as S is effectively an ordering of lip shapes according to lip opening (the first component of $\vec{X}$, which will dominate over other components in F^x when computing the distance matrix D.) The same is true of V, i.e. the first component of $F^{x\prime}$ will dominate over components with smaller variance, leading to an ordering of V according to that first component.

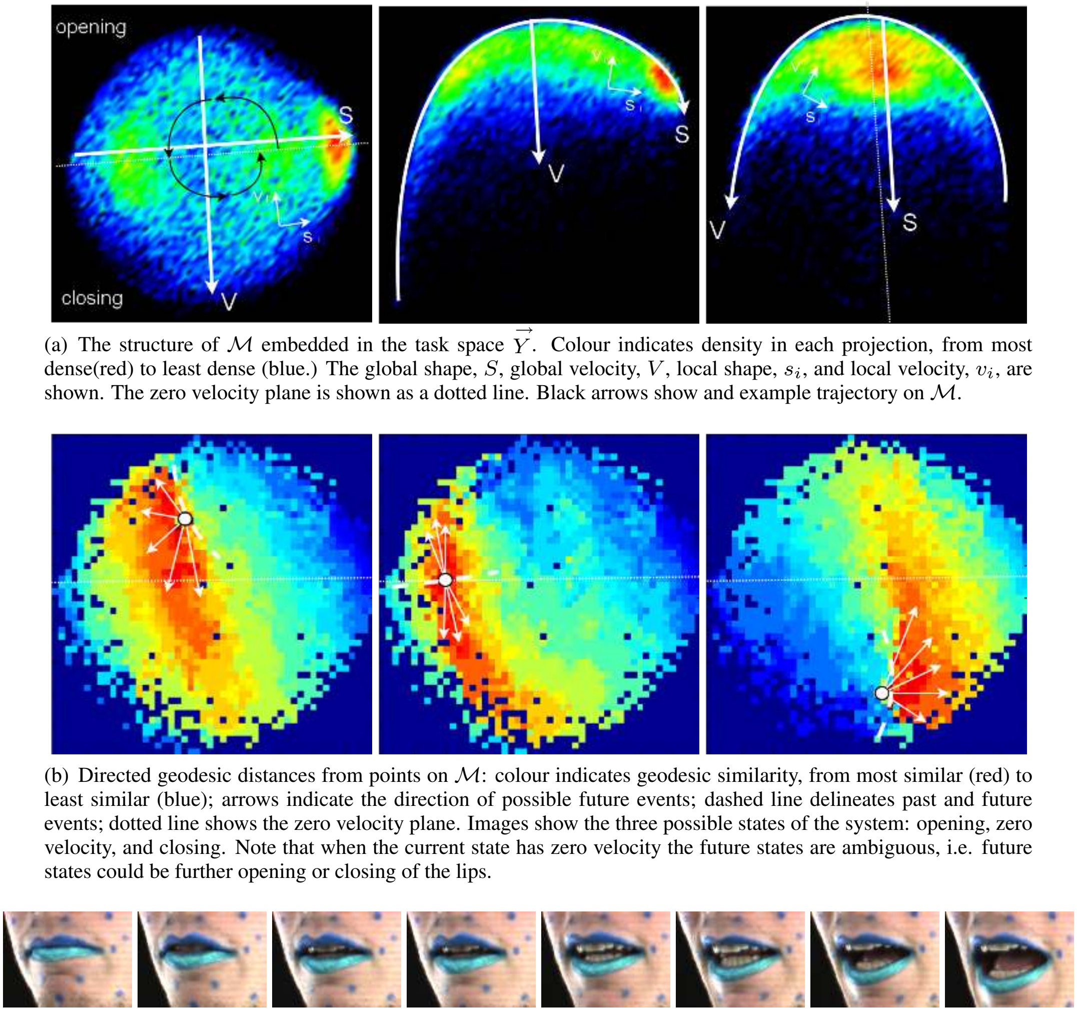
This local and global structure of $\mathcal{M}$ leads to several results. Firstly it is impossible to travel on the zero-velocity plane. This is obvious seeing as this would imply changing lip shape with zero velocity! Thus, when moving between the opening and closing halves of a speech cycle the trajectory will cross the plane of symmetry at right angles. Furthermore, at all other positions on the manifold, except on the plane of symmetry, the trajectory cannot travel parallel to v_i. This follows as to travel in the local tangential direction v_i implies maximum acceleration and minimum change in shape. In fact the trajectory can be described using the local tangent space described above, $\{s_i, v_i\}$. If you construct an offset between two consecutive frames, $z_i = (F_i^y - F_{i+1}^y)$, then $z_i.s_i$ and $z_i.v_i$ will categorise each frame into four different classes: opening/acceleration, opening/deceleration, closing acceleration, and closing/deceleration. These properties show that the speech manifold is highly structured, and potentially this structure can aid applications such as visual speech synthesis.

3.3. Distances on the Speech Manifold

Euclidean distances between pairs of points in the first several dimensions of $\vec{Y}$ approximate euclidean distances between pairs of points in G. However, the recovered structure of $\mathcal{M}$ indicates that a direct line between two non-consecutive states F_i^y and F_{i+n}^y does not give a good measure as to the time it would take to travel between them. In fact even a standard geodesic computation does not perform well because the manifold itself is directed, i.e. there is a temporal ordering of states, and thus distances are asymmetric. The natural way of understanding this is that given the lips are in a state F_i^y with velocity $F_i^{y\prime}$, the number of possible following states will be constrained by the physical properties of the lips (e.g. muscle properties, inertia etc.)

This evident asymmetry in distances can be accounted for using a modified geodesic computation. The F^y are initially grouped into states using K-means. A transition matrix, A, is then constructed with $A_{jk} = 1$ *iff* there is a transition from a member of state j to a member of state k, otherwise $A_{jk} = 0$. From this adjacency structure Dijkstra's algorithm can be used to calculate the *minimum distance* (i.e. minimum number of transitions $\approx$ minimum time in frames) between two states. Also, *minimum distance* paths between states can be calculated by propagating across the graph structure. There is no upper bound on the distance between two states as infinite cycles could occur in-between.

[2] Classical metric-MDS, PCO and Isomap will all produce related parametric spaces to the one described here. The techniques described in this paper are general enough that they could be used with any of these choices of projection.



(a) The structure of $\mathcal{M}$ embedded in the task space $\overrightarrow{Y}$. Colour indicates density in each projection, from most dense(red) to least dense (blue.) The global shape, S, global velocity, V, local shape, s_i, and local velocity, v_i, are shown. The zero velocity plane is shown as a dotted line. Black arrows show and example trajectory on $\mathcal{M}$.

(b) Directed geodesic distances from points on $\mathcal{M}$: colour indicates geodesic similarity, from most similar (red) to least similar (blue); arrows indicate the direction of possible future events; dashed line delineates past and future events; dotted line shows the zero velocity plane. Images show the three possible states of the system: opening, zero velocity, and closing. Note that when the current state has zero velocity the future states are ambiguous, i.e. future states could be further opening or closing of the lips.

(c) Lip shapes indexed by increasing S, the lip shape vector.

Figure 2: Properties of the visual speech manifold $\mathcal{M}$.

By examining *minimum distance* plots on $\mathcal{M}$, see fig. 2(b), the causal nature of the visual speech manifold becomes apparent. It can be clearly seen that given a current state F_i^y the possible set of future states is a subset of F^y, i.e. those for which $dist(F_i^y, F_p^y) = 1, F_p^y \in F^y$. A hemisphere of possible future events is evident at each point on the manifold, and likewise if the adjacency matrix is constructed using links between current and previous states (i.e. $A_{jk} = 1$ *iff* there is a transition from a member of state k to a member of state j) a hemisphere of possible past events can be seen. This indicates that a temporal ordering exists at each point on $\mathcal{M}$ pointing in the general direction of future events in the speech cycle. For states with zero-velocity, there are *two* directions of possible future states, this is due to the fact that with the lips in a state with zero velocity the lips can proceed into a further opening *or* closing cycle equally. Also, for states closer to the boundary of $\mathcal{M}$ the set of possible next states is much reduced.

4. Trajectory Synthesis

The speech manifold, $\mathcal{M}$, as discussed in the previous section, is useful in visualising what is happening in terms of the dynamics of speech production for the lips. However, interesting applications can be tackled if the inverse problem can be solved, i.e. can trajectories be derived directly from the structure of $\mathcal{M}$. The observations in Section 3.3 indicate that there is a temporal ordering of states in $\mathcal{M}$ which can potentially aid the generation of speech trajectories. An obvious application for this is speech synthesis, supposing that we define a set of states that the lips must pass through to articulate an utterance - the complete trajectory can be inferred from $\mathcal{M}$. This can be seen as a form of data-driven interpolation lying between traditional spline-based techniques (e.g. [13]) and trajectory concatenation (e.g. [14].)

4.1. Calculating Paths Between Pairs of Points

Given two targets on the speech manifold, the graph structure from Section 3.3 can be used to generate paths passing across $\mathcal{M}$. If we have a sequence of states $Q = \{q_1, q_2, \ldots, q_{n-1}, q_n\}$, we can say that each state is roughly $1/60^{th}$[3] of a second slice in time, and that the probability of this sequence of states is

[3] The same temporal resolution as the original data.

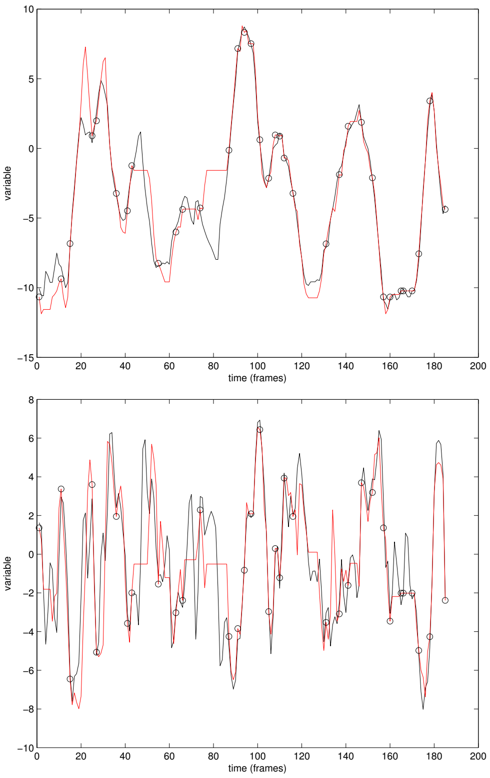


Figure 3: Generated trajectories in the first two components of $\overrightarrow{Y}$: black line is the original trajectory; red line is the generated trajectory; circles indicate phone mid-points.

$$P(Q) = \prod_{i=1}^{n} P(q_i|q_{i-1})$$

given that each transition $P(q_i|q_{i-1})$ is conditionally independent of previous nodes in the trajectory (i.e. a first-order *Markov* process.) In this case each edge in the graph is labelled with a probability derived from the relative frequency of transitions between neighbouring graph nodes (i.e. different to the binary edge costs used in Section 3.3.) The elements of the transition matrix, A, now become $A_{jk} = P(q_j|q_k)$. Thus we can calculate the probability of any possible path on $\mathcal{M}$. *Maximum probability* paths can be found between pairs of nodes on $\mathcal{M}$, which are distinct from *minimum distance* paths as discussed in section 3.3. However, it is more useful to be able to compute the *maximum probability* path between two states of a predetermined length n frames. If we have a trajectory defined by some key-frames (e.g. phoneme/viseme mid-point targets) given timings derived from the phonetic structure of an utterance, the structure of $\mathcal{M}$ can be used to generate the entire trajectory. A Viterbi-like algorithm is used to calculate these paths. Given a starting state, F^y_{start}, path probabilities are forward propagated across the graph structure for n frames. The optimal path from state $F^y_{start} \rightarrow F^y_{end}$ is then determined by backtracing to find the path corresponding to *maximum probability* in a manner similar to *dynamic programming*. This is efficient as, where phone mid-points are used as targets, the distance in time between targets is rarely greater than 10 frames.

An example of a trajectory generated by interpolating phone mid-points, for the sentence *”Herb’s birthday occurs frequently on Thanksgiving”*, on the speech manifold is shown in fig. 3. As can be seen, where the phone mid-points strongly define the trajectory this method produces a good approximation to the original data. Where the interpolation fails this is generally due to two reasons: firstly, the discretisation of the manifold means that the trajectory may not exactly be matched; secondly, where there are higher frequency components to the original signal (i.e. far higher frequency than the targets being interpolated.) The second issue is the more important and implies that the number of required targets may be greater than the number of phone mid-points in the target utterance. Even so this form of trajectory synthesis performs far better than a direct interpolation of phone targets.

4.2. Calculating the Inverse Mapping

It is important to provide a mapping back from the embedded manifold $F^y \rightarrow F^x$ for the parameterisation to be useful for many applications, particularly for synthesis. There is no direct mapping because $\overrightarrow{Y}$ is constructed from the distance matrix D, and only preserves relative distances between states. In $\mathbb{R}^n$ barycentric coordinates can be defined using a simplex consisting of $n + 1$ points. Thus, in $3D$ for any given point in the embedded space $\overrightarrow{Y}$, the projection of a point p^y back into $\overrightarrow{X}$ can be determined using the bounding tetrahedron Δ^y_{ijkl}. *Delaunay* tetrahedralisation can be used to find an appropriate structure for the speech manifold to provide this mapping, and similar techniques can be used even when the number of dimensions exceeds 3.

Given p^y is in the tetrahedron Δ^y_{ijkl} then the projected point p^x is calculated as a barycentric combination of the surrounding vertices.

$$p^x = F^x_i.B_i + F^x_j.B_j + F^x_k.B_k + F^x_l.B_l, p^y \in \Delta^y_{ijkl}$$

The barycentric weights, B_*, are defined as the ratio of subvolumes within the tetrahedron,

$$B_i = \frac{V_\triangle(p^y, F^y_j, F^y_k, F^y_l)}{V_\triangle(F^y_i, F^y_j, F^y_k, F^y_l)}, B_j = \frac{V_\triangle(p^y, F^y_i, F^y_k, F^y_l)}{V_\triangle(F^y_i, F^y_j, F^y_k, F^y_l)},$$

$$B_k = \frac{V_\triangle(p^y, F^y_i, F^y_j, F^y_l)}{V_\triangle(F^y_i, F^y_j, F^y_k, F^y_l)}, B_l = \frac{V_\triangle(p^y, F^y_i, F^y_j, F^y_k)}{V_\triangle(F^y_i, F^y_j, F^y_k, F^y_l)},$$

where $V_\triangle$ is the volume of a tetrahedron spanning four vertices.

$$V_\triangle(p_0, p_1, p_2, p_3) = \frac{|(p_0 - p_3).((p_1 - p_3) \times (p_2 - p_3))|}{6}$$

Likewise, the mapping from points in $\overrightarrow{Y}$ back to the space of the original data F can be constructed by replacing the F^x_* in the barycentric combination with the original frames F_*.

Combined with the trajectory interpolation described in the previous section we can now take a set of discrete targets on $\mathcal{M}$, determine a path interpolating these targets, and map this generated trajectory back to the data domain F. This accounts for half of the problem of synthesis, the important missing piece being how the targets are chosen according to the phonetic structure of a target utterance.

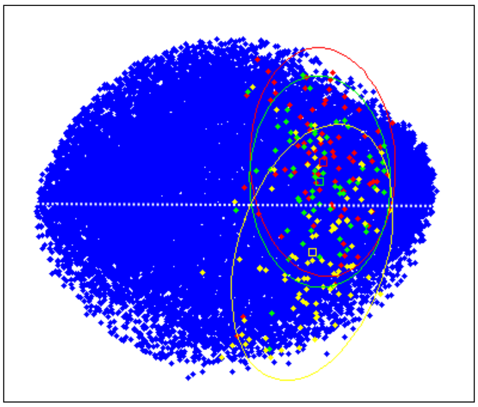


Figure 4: Clustering of bilabial consonants centres on the speech manifold $\mathcal{M}$: /b/ centres are shown in red, /p/ centres are shown in green, /m/ centres are shown in yellow. Ellipses show 2 s.d. distribution of states, squares show mean state. The /p,b/ clusters are mainly located on the opening half of the speech cycle, and the nasal /m/ clusters are mainly within the closing half of the cycle.

5. Phonetic Clustering on the Speech Manifold

Of further interest is how lip shapes corresponding to phone mid-points cluster on $\mathcal{M}$. An example of clustering for bilabials can be seen in fig. 4. This group demonstrates an example of dynamics separating groups which are similar when only lip-shape is taken into account. The nasal /m/ cluster is located mostly in the closing half of the speech cycle whereas the voiced/voiceless /b/ and /p/ clusters are located mostly on the opening half of the cycle. Spatially all three classes cluster about narrow/closed lip shapes, but are widely spread across all velocities - likely a result of the surrounding phonetic context. This fits with the plosive nature of /p,b/, and the nasal stop description of /m/. By implication the nasal /m/ group should be treated separately from the plosive group /p,b/, which closely overlap, whilst in many cases where dynamics are not taken into account these are all treated as one group (e.g. in viseme models.)

The wide variation in the direction of V indicates the variety of possible trajectories that can produce bilabials. In terms of synthesis the question now becomes how do we choose states from phoneme clusters that are appropriate for interpolation (e.g. by the method described in Section 4.1.) This is the coarticulation problem, the choice of a sequence of states, according to context, (incorporating both lip shape and velocity) which describe the articulatory trajectory required to produce a particular utterance. The investigation of trajectories, how they pass through phoneme clusters, and grouping according to context are important future stages in the analysis of the structure of the visual speech manifold $\mathcal{M}$.

6. Conclusions

In this paper we have presented a novel parameterisation of $3D$ lip speech movements. This parameterisation is a constructed space in which lip movements can be visualised as elliptical paths on a non-linear manifold. Geodesic distances on the manifold, calculated according to the temporal ordering of the original data, allow us to extract an underlying causal structure to the task space. By exploiting this ordering of data, trajectories between discrete states on the speech manifold can be generated using a finite-state *Markov* model. This is a form of data-driven target interpolation which is not constrained by a pre-defined mathematical model of speech trajectories (such as spline-based models), or by the discrete set of pre-captured fundamental units (such as with concatenative models.) Some initial results for clustering of phone mid-points have been presented, which demonstrate the separation of phonetic groups when dynamics are included within the parameterisation.

The presented techniques are intended to aid the task of speech synthesis. Given that we can project sets of phone instances onto the speech manifold, and that these cluster, synthesis becomes a matter of selecting key targets from each of the clusters - between which data-driven interpolation generates the synthetic trajectory. Naturally this is obfuscated by coarticulation, the rôle of context in the realisation of target extrema. Furthermore, it is evident from captured trajectories that phonetic centres are not the only key features within speech utterances. Future work will concentrate upon the identification and selection of key targets from the speech manifold.

7. References

[1] Blanz, V., and Vetter, T., A morphable model for the synthesis of 3D faces, Proceedings of ACM Siggraph, 1999, pp.187-194.

[2] Cootes, T.F., Edwards, G.J., and Taylor, C.J., Active appearance models, Proceedings European Conference on Computer Vision, 1998, pp. 484-498.

[3] Theobold, B-J., and Wilkinson, N., A real-time speech-driven talking head using active appearance models. Proceedings of AVSP'07, 2007, pp.22-28.

[4] Gower, J.C., Some distance properties of latent root and vector methods used in multivariate analysis, Biometrika 53, 1966, pp.325-338.

[5] Cao, Y., Faloutsos, P., and Pighin F., Unsupervised learning for speech motion editing, Proceedings of Symposium on Computer Animation, 2003, pp.225-231.

[6] Roweis, S., and Saul, L., Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500), 2000. pp.2323–2326.

[7] Tenenbaum, J. B., de Silva, V., and Langford, J.C., A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500), 2000, pp. 2319-2323.

[8] Grochow, K., Martin, S.L., Hertzmann, A., Popovic, Z. Style-based inverse kinematics, Proceedings of ACM Siggraph, 2004, pp.522-531.

[9] Wang, J.M., Fleet, D.J., and Hertzmann, A., Gaussian process dynamical models, Proceedings of NIPS'05, 2005, pp. 1441-1448.

[10] http://www.3dmd.com/

[11] Edge, J.D., Hilton A., Nadtoka, N., 3D video face capture and registration, BMVA Symposium on 3D Video Analysis, Display and Applications, 2008.

[12] Roweis, S., EM algorithms for PCA and SPCA, Proceedings of NIPS'97, 1997, pp.626-632.

[13] Cohen, M.M., and Massaro, D.W., Modeling coarticulation in synthetic visual speech, In: Magnenat-Thalmann, N., Thalmann, D. (Eds.), Models and Techniques in Computer Animation, Springer, Berlin, 1993, pp. 139-156.

[14] Kshirsagar, S., and Magnenat-Thalmann, N., Visyllable based speech animation, Proceedings of Eurographics, 2003, pp.632-640.