An unsupervised method for learning to track tongue position from an acoustic signal

John Hogden*, Philip Rubin** & Elliot Saltzman**

*MS B265, Los Alamos National Laboratory, Los Alamos, NM 87545, USA
**Haskins Laboratories, 270 Crown St., New Haven, CT 06511, USA

ABSTRACT

A procedure is demonstrated for learning to recover the relative positions of simulated articulators from speech signals generated by articulatory synthesis. The algorithm learns without supervision, that is, it does not require information about which articulator configurations created the acoustic information in the training set. The procedure consists of vector quantizing short time windows of a speech signal, then using multidimensional scaling to represent quantization codes that were temporally close in the encoded speech signal by nearby points in a *continuity map*. Since temporally close sounds must have been produced by similar articulator configurations, sounds which were produced by similar articulator positions should be represented close to each other in the continuity map. Continuity maps were made from parameters (the first three formant center frequencies) derived from acoustic signals produced by an articulatory synthesizer that could vary the height and degree of fronting of the tongue body. The procedure was evaluated by comparing estimated articulator positions with those used during synthesis. High rank-order correlations (0.95 to 0.99) were found between the estimated and actual articulator positions. Reasonable estimates of relative articulator positions were made using 32 categories of sound and the accuracy improved when more sound categories were used.

RÉSUMÉ

Une procédure visant à inférer, à partir du signal acoustique, les positions correspondantes des articulateurs du conduit vocal est présentée dans cet article. Elle est évaluée sur des signaux de parole synthétique obtenus par synthèse articulatoire : le but est de retrouver les positions relatives des articulateurs qui ont effectivement été les commandes du modèle articulatoire utilisé dans la synthèse. L'algorithme est fondé sur un apprentissage non supervisé, qui ne requiert aucune information sur les dispositions articulatoires qui ont été utilisées dans le corpus d'apprentissage. La procédure consiste d'abord en la quantification vectorielle de courtes fenêtres temporelles du signal de parole ; puis grâce à une technique de mise à l'échelle multidimensionnelle (Multidimensional scaling) on représente les codes de quantification qui se succèdent temporellement dans le signal de parole ainsi codé par des points voisins dans une *carte de continuité*. Puisque les sons qui se succèdent dans le temps sont vraisemblablement produits à partir de configurations articulatoires similaires, les sons qui ont été produits par des positions articulatoires similaires devraient se situer à proximité les uns des autres dans la carte de continuité. Les carte de continuité ont été établies à partir de paramètres (les trois premières fréquences formantiques) obtenus par l'analyse de signaux acoustiques produits par un synthétiseur articulatoire qui peut agir sur la hauteur de la langue et le

Bulletin de la communication parlée n° 3, p. 101-116. 1996

degré d'avancement du corps de la langue. La procédure a été évaluée sur la base des erreurs entre les positions articulatoires inférées et celles qui ont été effectivement utilisées pour la synthèse. De fortes corrélations (0.95 to 0.99) ont été trouvées entre les positions des articulateurs estimées et celles qui sont effectivement utilisées lors de la synthèse. Des estimations satisfaisantes des positions relatives des articulateurs ont été obtenues en utilisant 32 catégories de son, et la précision de l'estimation croît si plus de catégories sonores sont utilisées.

1. Introduction

A growing body of research (e.g. Atal, 1975 ; Boë *et al.*, 1992 ; Hogden *et al.*, 1993 ; Jordan & Rumelhart, 1992 ; Kawato, 1989 ; Kuc *et al.*, 1985 ; Ladefoged *et al.*, 1978 ; McGowan, 1994 ; Papcun *et al.*, 1992 ; Rahim *et al.*, 1991 ; Schroeter & Sondhi, 1992 ; Shirai & Kobayashi, 1986) supports the hypothesis that information about articulator positions can be recovered from the acoustic speech signal. This conclusion is somewhat surprising since, when the acoustic properties of the vocal tract are modeled by lossless acoustic tubes, radically different vocal tract shapes can have identical transfer functions (Fant, 1970 ; Flanagan, 1972). Furthermore, although adding a glottal energy loss to the vocal tract model can make the mapping from acoustics to vocal tract shapes unique (Markel & Gray, 1976), adding energy losses is not always sufficient to eliminate vocal tract shape ambiguities (Atal *et al.*, 1978)

Energy losses or not, it is clear that the shape of an acoustic tube cannot be uniquely determined from information about formant frequencies of a single transfer function without incorporating additional constraints. This has been shown using articulatory synthesizers, both with and without energy losses (Atal, *et al.*, 1978 ; Maeda, 1989 ; Stevens & House, 1955). Linear prediction theory leads to the same conclusion by showing that formant frequencies and bandwidths must both be used to determine vocal tract shape. Finally, bite-block experiments confirm that people can produce vowels with nearly normal values of the first three formant frequencies using a « physiologically unnatural position of the mandible » (Lindblom *et al.*, 1979). It is difficult to argue that bite block vowels are acoustically identical to normally produced vowels – perceptual differences between normal and bite-block vowels have been noted (Fowler & Turvey, 1980) – but Lindblom *et al.*, found that the first three formants of bite block vowels were usually within 3 standard deviations of normal vowel formants with few systematic deviations.

Nonetheless, there has been some success at recovering articulation from acoustics. For example, given a training set consisting of acoustic signals generated by an articulatory synthesizer and the articulator positions used to produce them, Atal (1975) found a non-linear regression function that calculated seven vocal tract parameters (constriction location, constriction degree, lip protrusion, etc.) from twelve acoustic parameters (six formant frequencies and six bandwidths). The importance of using as much acoustic information as possible was reinforced in this study because Atal was able to determine vocal tract parameters from representations of the acoustic signal provided the acoustic representation included information about a sufficient number of formant frequencies and bandwidths.

Atal's success in the 1975 study was based, at least in part, on the fact that the model vocal tract used for synthesis had fewer degrees of freedom than the acoustic information used to recover the

tract shape. Conversely, the finding by Atal *et al.*, (1978) that many different vocal tract shapes can lead to the same acoustic signal was partly due to the fact that the number of articulatory parameters to recover was greater than the number of acoustic parameters measured. Clearly, vocal tract shape can be determined more accurately if the number of acoustic parameters used to determine vocal tract shape exceeds the number of parameters used to describe vocal tract shape.

Unfortunately, as Sondhi (1979) mentions, only a limited number of acoustic parameters can be accurately recovered from speech. This poses the serious question of whether the vocal tract shapes used during speech can be described with fewer parameters than the number of acoustic parameters that can be accurately recovered from speech. There is support for the contention that the articulator positions commonly used during vowel production can be described parsimoniously ; tongue shape can be adequately represented by only 2 or 3 parameters (Harshman, *et al.*, 1977 ; Morrish *et al.*, 1985) and vocal tract shapes in general can be represented by about 7 to 10 factors (Coker, 1976 ; Maeda, 1989 ; Rubin *et al.*, 1981). Evidence that human articulator positions can be recovered from acoustic information has been presented by Ladefoged *et al.*, (1978), who used multiple regression to find a relationship between the first three formants and two PARAFAC factors representing tongue shape. Tongue positions inferred from the first three formants of steady state vowels accurately reflected the tongue positions seen in X-ray tracings for several subjects, although there was some difficulty in estimating the tongue shapes used to produce the vowel [a]. Similarly, Hogden *et al.*, (1993) recovered articulator positions using a look-up table.

Some articulatory features can be more easily recovered from speech than others. For example, Boë *et al.*, (1992) used an articulatory synthesizer based on X-ray data (Maeda, 1979) to show that the location and area of the oral constriction used in vowel production could be determined from the first three formants alone, even though the complete shape of the vocal tract could not be recovered. This research demonstrates that even if the vocal tract shape is not entirely recoverable from the acoustic signal, aspects of articulation that are important for phonetic identification may be recoverable. Continued research in this direction may uncover other articulatory features that can be determined despite ambiguous mappings from acoustics to vocal tract shape.

Most techniques for solving the acoustic-to-articulatory mapping problem have not been rigorously tested on human articulatory/acoustic data because of the difficulties involved in measuring the articulator positions. Three exceptions to this rule are the studies by Ladefoged *et al.*, (1978) and Hogden *et al.*, (1993) that were already discussed, and also a study by Papcun *et al.*, (1992). The latter study found that neural networks, which perform a type of non-linear regression, can calculate X-ray microbeam pellet positions from spectral information. As in Atal's nonlinear regression study, Papcun *et al.*, supplied their recognition algorithm with more acoustic information than simple measurements of formant frequency. One difference between Atal's study and the study by Papcun *et al.*, is that Atal used acoustic signals from static vocal tracts while Papcun *et al.*, gave the neural network spectral information from successive short-time windows of speech, essentially providing acoustic information from successive vocal tract shapes. This difference is important because using information from several spectral slices can help overcome one-to-many mapping problems (Kuc, *et al.*, 1985 ; Rahim, *et al.*, 1991).

We will describe a novel method, the *continuity mapping* technique (Hogden, 1991 ; Hogden *et al.*, 1992a), for computing articulator information from the speech wave. The goal of the continuity mapping algorithm is to produce a map, called a continuity map (CM), in which acoustic signals that are produced close together in time are represented by points that are close to each other in

the continuity map. The reasoning behind this is that speech sounds mapped to nearby locations in the continuity map (those produced close together in time) must have been produced by similar articulator configurations. We know that temporally proximate acoustic signals were produced by similar articulator configurations because the articulators move continuously, i.e. they do not move from one position to another without occupying intermediate positions. Since sounds produced by similar articulator configurations are mapped close together in the continuity map, the continuity map should give topologically accurate information about articulator positions.

CMs differ from other topological maps of acoustic signals (Kohonen, 1988) in that for CMs acoustic signals are not placed close together on the basis of acoustic similarity. Unlike other topological mapping procedures, the CM algorithm is trying to recover information about articulator positions, and acoustic signals which are completely different can be produced from very similar articulator configurations. For example, the tongue only needs to move a small distance to change from producing a non-fricative to a fricative – drastically different sounds. To recover articulatory information, acoustically dissimilar sounds need to be able to be placed close to each other in the map. By placing acoustics signals close to each other in the CM if they were produced close to each other in time, drastically different acoustic signals can be represented next to each other, something that is not possible when using an acoustic distance measure.

Unlike previous techniques for recovering articulator positions, which determine the absolute positions of the articulators, continuity mapping only determines their relative positions. However, the relative articulator positions are estimated by an unsupervised algorithm, i.e. without giving the algorithm access to explicit information about the articulator positions used to generate the acoustic signals in the training set. Understanding the difference between a supervised and an unsupervised learning algorithm is essential for evaluating the advantages and disadvantages of the continuity mapping algorithm, thus we will discuss it in a little more detail.

Regression can be thought of as a supervised learning technique for estimating y values from x values. To estimate values of y from values of x we find the regression line relating y and x. To calculate the regression line, examples of (x, y) pairs are needed. The best fitting line cannot be found from x values alone. That is the defining characteristic of a supervised algorithm : examples of both the inputs and outputs are needed for learning. Being supervised algorithms, previous methods for determining articulator positions from acoustics require simultaneous measurements of articulator positions and the resulting acoustics.

The continuity mapping algorithm is an unsupervised algorithm. To continue the analogy to regression, using the continuity mapping technique is somewhat like finding the regression line relating x and y when given only the x values. An unsupervised algorithm is not given the desired output values – even during training. If the continuity mapping procedure is successful, it could learn to relate acoustics to articulation from a tape recording of an individual's speech – without any articulatory measurements. In the present work note that, although we do have simultaneous measurements of acoustics and articulation, the continuity maps are made from the acoustic data alone. The articulatory data is only needed to compare estimated articulator positions to the actual articulator positions.

From the above discussion, it should be clear that supervised learning algorithms will be difficult to apply to the problem of recovering articulator positions from acoustics. The difficulty lies in the fact that, to use a supervised learning algorithm to recover articulator positions, we need to gather a huge set of simultaneous articulatory and acoustics data. Without the simultaneous data, the supervised algorithms can not learn to relate acoustic to articulation. Needless to say, it is

still very difficult to gather such data, so supervised algorithms are not yet practical solutions to real world problems.

Supervised algorithms are also problematic if you believe that perceiving speech is tantamount to perceiving articulator gestures (Liberman *et al.*, 1967 ; Liberman & Mattingly, 1985). After all, when children perceive speech produced by others, the children are not told what articulator positions the other speakers are using. While the children do have access to information about their own articulations, a child's speech is acoustically different from adult speech, so it is difficult to imagine how the child could *learn* to relate adult acoustics to articulator positions given only examples of child speech (although innate knowledge, or possibly some kind of normalization, could be used to get around this problem). Similarly, it is difficult to understand how people could learn to perceive sounds which they cannot produce, as in sounds from foreign languages, or sounds that a child has not yet learned to produce (Smith, 1973).

Being an unsupervised algorithm, continuity mapping avoids the previously mentioned problems inherent in supervised algorithms ; however, some information is lost to gain the advantages of unsupervised learning. Unlike supervised algorithms, the continuity mapping algorithm is not able to recover the absolute positions of the articulators – only the *relative* positions of the articulators can be estimated. Any rotation, reflection, translation, scaling or other topological transformation of the estimated positions will be an equally acceptable solution as far as the continuity mapping algorithm is concerned.

The continuity mapping algorithm also faces normalization problems, i.e. a map relating acoustics to articulation created for one speaker may not be accurate for a different speaker. So, for the continuity mapping algorithm to be useful, we will either need to determine some way to normalize speech signals from different speakers (as is also the case for supervised algorithms), or we will need to make a variety of continuity maps to accommodate different speakers.

Because of the potential advantages of continuity mapping, several continuity maps were created and tested on acoustic data generated by an articulatory synthesizer. The following discussion describes these experiments.

2. Generating an articulator map

Since gathering simultaneous information about the entire set of articulator positions (especially the tongue) and speech acoustics is quite difficult, the articulatory speech synthesizer at Haskins Laboratories (Mermelstein, 1973 ; Rubin, *et al.*, 1981) was used to generate acoustic signals from static vocal tract configurations. Only the two-dimensional articulator space defined by the synthesizer's degrees of freedom for tongue body motion was investigated. The rest of the articulators were fixed at their neutral positions.

We chose to use two degrees of freedom purely for purposes of illustration. As long as the mapping from acoustics to articulation is not one-to-many, it should be possible to recover information about more than two degrees of freedom as well. However, with articulatory synthesizers, there are typically one-to-many mappings from acoustics to articulation in static synthesis. To stick to our main objective – illustrating the continuity mapping algorithm – our initial work has been limited to recovering two degrees of freedom.

To cover the full range of tongue body positions, the tongue body center was placed at each of 2500 equidistant points in a square grid. Excluding tongue positions that completely closed the

vocal tract left 2011 viable tongue positions. Fig. 1 gives a flavor of the range of tongue positions by showing some of the more extreme positions. A vector composed of the first three formants of the resulting acoustic signal was calculated for each tongue position.

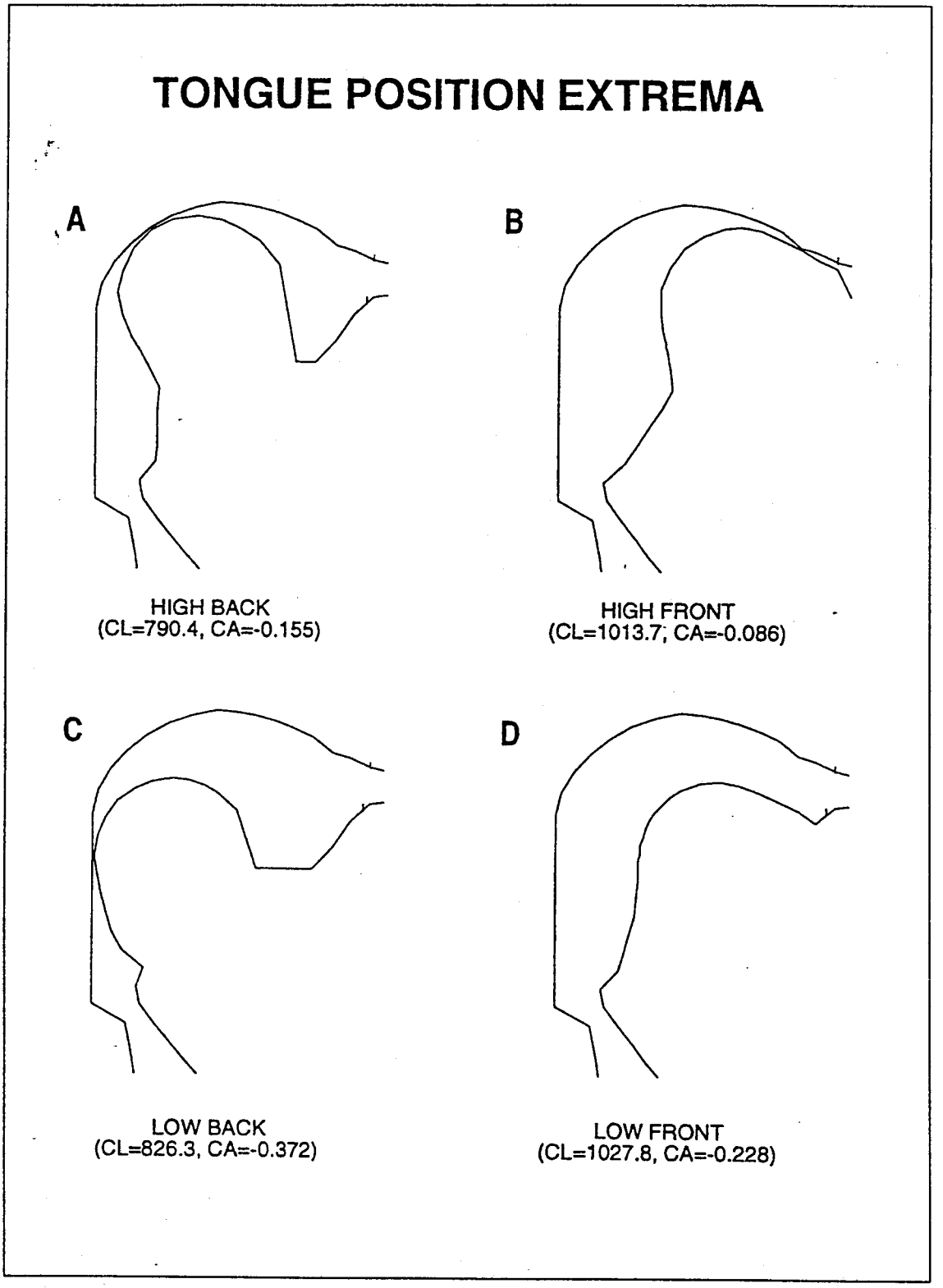

Figure 1 : Examples of vocal tract shapes created using extreme tongue positions. The articulatory synthesizer parameters used to make these tongue positions are given in parentheses below the shapes.

Each formant vector was replaced by a scalar code using a nearest neighbor coding technique. In nearest neighbor coding, the acoustic similarity between each formant vector and each of a set of prototypical formant vectors is calculated (by finding the Euclidean distance between formant vectors, for example), and the formant vector is replaced by the code representing the most similar prototype. We used a weighted Euclidean distance in formant space as the measure of acoustic similarity. The weight on any formant was the inverse of the standard deviation of the formant, calculated over all tongue positions. The weighted Euclidean distance measure is only one of a variety of distance measures that would all be reasonable. The appropriate distance measure to use for natural speech will likely be more complex (Schroeter *et al.*, 1990), but our goal is to illustrate the continuity mapping procedure, so a more complete discussion of possible distance measures is beyond the scope of this paper.

The set of prototypical acoustic signals used in the nearest neighbor coding scheme were derived using a K-means vector quantization (VQ) algorithm (Gray, 1984 ; O'Shaughnessy, 1987). The VQ algorithm starts with some initial set of acoustic prototypes and moves them around in formant space to minimize the sum of the acoustic distances between the sounds being categorized and the prototypes they are closest to. Since the VQ minimization technique can run into local minima, it needs to be used with different sets of initial prototype positions. We generated three sets of 32 prototypes (each set is called a codebook because the prototypes are referred to by a number called a code) and used for further study the codebook which best minimized the error function.

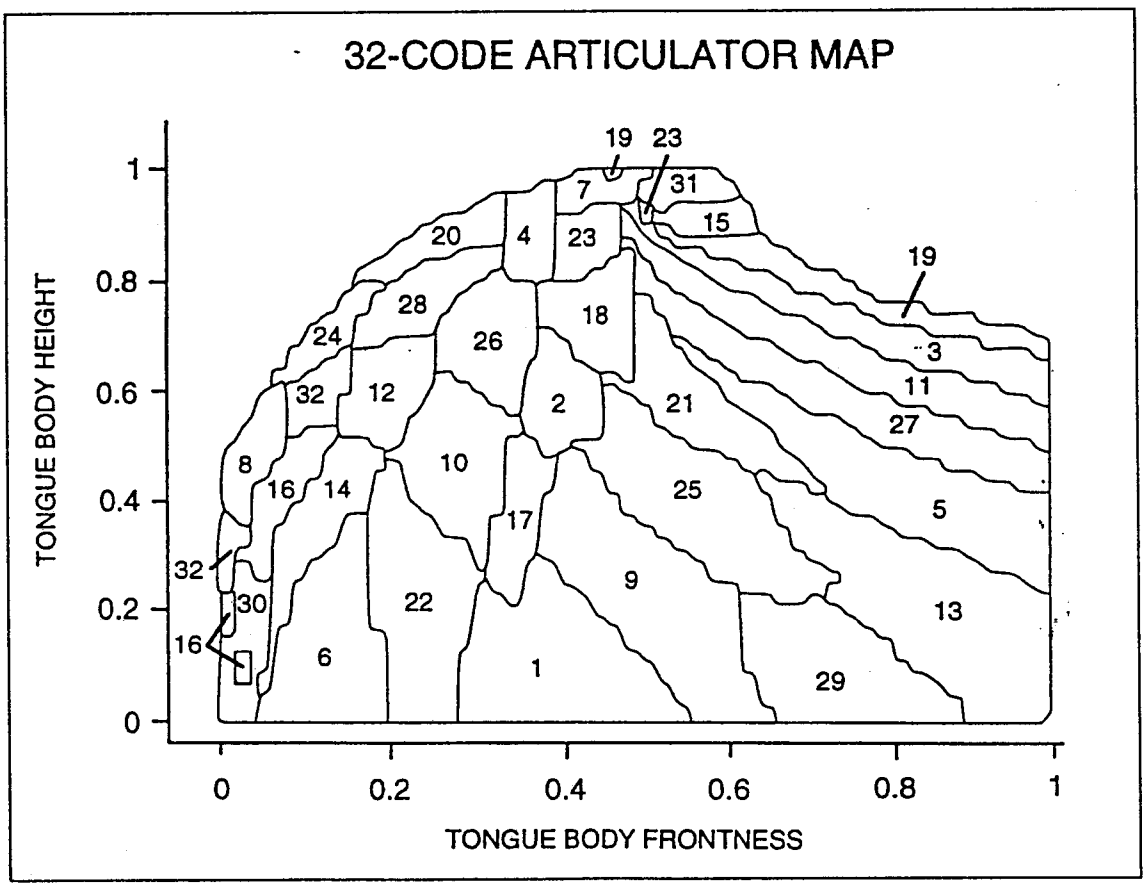

Figure 2 : Articulator map constructed using 32 codes. Each position in the map represents a tongue position. The numbers plotted are codes indicating which acoustic parameters are produced within the isocode regions.

The effect of quantizing the acoustic parameters can be seen in what we call an *articulator map* (AM), like that in Fig. 2. Fig. 2 shows which vector quantization prototype was most similar to the formant vector produced with each tongue position. The axes of the plot represent tongue body height and frontness and the numbers plotted in the figure are the codes representing the closest prototype. Each code is plotted near the center of a small region which we call an *isocode region.* As the name implies, all the acoustic signals produced with the tongue body positioned within a single isocode region are represented by the same code. It is important to realize that the VQ algorithm generates categories based on acoustics alone. We are able to make the map shown in Fig. 2 because, in this experiment, we know the both articulator positions and the resulting derived acoustic values. However, the articulator positions are not needed to perform VQ, and being able to draw the articulator map is not essential for creating a continuity map – the articulator map merely helps to visualize how the continuity mapping algorithm works.

Notice that some of the codes are produced in two or more distinct regions. Code 16 is one example. As in other studies (Stevens & House, 1955), the codes that occur in disjoint regions are found mostly when the tongue body center is low and back. When a code is found in more than one distinct region, the regions are fairly close together, so the first three formant frequencies are sufficient to determine two tongue position parameters with a relatively small error. As has already been discussed, for synthesized speech the extent of the one-to-many mapping problem depends on the relative number of articulatory and acoustic parameters. Thus, if the first three formants were used to recover more than two articulator parameters, there would probably be more cases where different articulator positions created similar acoustic signals. While we do not want to draw conclusions about the human acoustic to articulatory mapping from this example of synthesized add dashes speech, it will be seen that one-to-many mappings do not always prevent good articulator position estimates.

3. Generating a continuity map

To allow the continuity mapping algorithm to use information about which signals can be produced close together in time, sequences of codes were produced by taking random walks among the 2011 viable articulator positions. These walks were intended to provide examples of the sounds which could be produced by varying the tongue position in a continuous fashion, so the steps were made short enough to insure that transitions only occurred between adjacent regions. At each time step, the code produced using the current tongue position was output and the tongue was moved a short distance in some random direction. The random walk continued until at least N transitions were made to each code, with N taking on the value of 1, 2, 3, 4, 5, 10, or 50. Three random walks were made for each value of N, for a total of 21 random walks. As discussed below, random walks provide relatively poor information about the distances between isocode regions, and so give us a conservative means for determining how well the continuity maps will be able to estimate articulator positions.

A set of intercode distance estimates was made for each random walk by calculating the average number of transitions between codes. The average number of transitions is calculated after first eliminating adjacent repetitions of codes, e.g., a sequence like « 25, 25, 25, 13, 13, 13, 5, 21, 21, 25, 29, 13, 29, 9 » is reduced to « 25, 13, 5, 21, 25, 29, 13, 29, 9 ». The next step is to count the number of transitions between each pair of codes in the sequence. To do this, we start with the

first code in the sequence and count the number of transitions to codes occurring later in the sequence. Then we start from the second code in the sequence and count the transitions from there, etc.

Counting from a particular starting code continues until *any* of the codes in this counting sequence is encountered twice. The justification for restarting at code repetitions is that, without restarting, intercode distance estimates would be overestimates. In the example given, the distance between code 25 and code 29 is overestimated by counting the number of transitions from the initial 25 to the 29 because the second occurrence of code 25 is adjacent to code 29. So, starting from the initial 25, we only count until we get to the occurrence of code 21, three transitions away. Similarly, counting from the second occurrence of code 25, we avoid a repetition of code 29 by only counting until we reach code 13, two transitions away. Thus, in the example sequence given above, we find that code 25 is one transition from code 13, then find that code 25 is two transitions from code 5 and three transitions from code 21. Next we count from code 13 (the second code in the sequence) and find that code 13 is one transition from code 5, two transitions from code 21, etc.

Notice that three estimates of the distance between code 25 and code 13 are obtained, since we count from the first example of code 25, then from the second example of code 25, and finally from the first example of code 13. All three estimates are averaged to get the mean number of transitions between code 25 and code 13. Note also, however, that this counting scheme gives no estimate of the distance between code 25 and code 9. This is because when counting from the first example of code 25, we see that code 25 is repeated before getting to code 9. Counting from the second 25 in the sequence, code 29 is repeated before code 9 is encountered. When the counting scheme does not give any estimates of the distance between two codes, a distance estimate equal to the number of codes in the codebook is used, effectively giving a maximum estimate of the distance.

Now that the method used to estimate the distances between isocode regions has been discussed, we can explain why code sequences were generated by random walks. Suppose we are trying to estimate the distance between isocode region 26 and region 22 from the sequence of VQ codes. If the tongue makes a relatively smooth downward motion from region 26 to region 22, we expect to see the VQ code sequence : 26, 10, 22. Notice that this code sequence gives a good estimate of the number of regions between region 26 and region 22. In contrast, if the tongue takes a random walk, it is fairly likely to travel to code 18, 23, 4, 28, 12 or even code 13 for that matter, before it gets to code 22. Typically, a random path is a longer than necessary way to get from one point to another. It is only by averaging the information from such random paths that we get distance estimates that should be monotonically related to actual distance estimates. Presumably the continuity mapping algorithm will work better given smoother tongue motions, as long as the tongue motions still travel through each of the isocode regions.

The relative positions of the isocode regions were estimated from the average intercode transition distances using non-metric multidimensional scaling (MDS). Multidimensional scaling calculates relative point positions from interpoint distances by starting with some initial configuration of points in space, and then moving the points until the distances between the points are nearly monotonically related to the desired interpoint distances (Dillon and Goldstein, 1984, provide more information about MDS). The MDS algorithm moves the points using gradient descent on an error measure, *stress*, which is a measure of the departure from a monotonic relationship between the interpoint distance as determined by MDS and desired interpoint distances.

While MDS is capable of producing solutions with different numbers of dimensions, the interpoint distances were generated from two-dimensional data, so solutions of more than two dimensions will only be fitting the noise in the intercode distance estimates. Because we know that the correct MDS solution is two-dimensional (i.e. the articulator map is two-dimensional), all the continuity maps have two dimensions.

The gradient descent minimizations performed by MDS can find local minima as well as global minima, where local minima are solutions that minimize stress in a local region, but which are not the best solution. The best way to avoid using a solution that is merely a local minimum is to run the MDS algorithm from a variety of different random starting configurations. So, to avoid local minima, five two-dimensional solutions were found for each set of interpoint distances, using a different initial configuration of points to get each solution. Since there were 21 random walks, 105 different solutions were found. For each random walk, the solution with the lowest stress value was used for further analysis, giving one best solution for each random walk. The resulting maps are called continuity maps (CMs) because they are made based on the fact that articulators move in a continuous fashion.

To show that different random walks lead to very similar CMs, CMs made from different random walks were compared using *generalized Procrustes analysis* (Gower, 1975), a technique for rotating, translating, reflecting, and scaling (only uniform scaling is allowed) configurations to make them maximally similar, and then calculating a measure of how similar the different configurations are. Fig. 3 illustrates Procrustes analysis (Lederman, 1984), the basic component of generalized Procrustes analysis. Two configurations of three points each are shown in Fig. 3a. As

Figure 3a

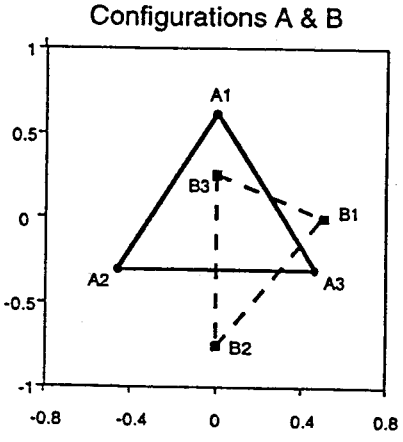

Figure 3b

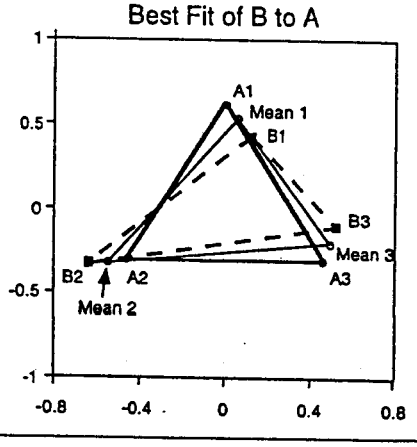

Figure 3 : Example of Procrustes analysis. Since only the relative configuration of points is of interest, the axes are not relevant and therefore are unlabeled. Plot 3A shows the two configurations that need to be compared. In plot 3B, configuration B has been rotated to best fit configuration A, and the mean configuration is shown.

you can see, the configurations are not aligned and would not be identical even if they were better aligned. Fig. 3b shows the result of using Procrustes analysis to rotate, reflect, scale, and translate configuration B to best fit configuration A. In a perfect fit, point A1 would be directly over point B1, A2 would be directly over B2, and A3 would be directly over B3, which is not the case for these two configurations. To calculate the deviation from a perfect fit, the configurations are compared to the mean configuration, also shown in Fig. 3b, by finding the square root of the mean squared distance between each point and the corresponding mean position. The mean configuration can also be used as the estimate of the true configuration. For the extension of this procedure to more than two configurations (the extension is called *generalized* Procrustes analysis), refer to Gower (1975).

The results of generalized Procrustes analyses of the CMs generated by random walks of the same length are shown in Fig. 4. The error in Fig. 4 is given in the same units that are used on the axes of Fig. 5A. These errors are extremely small – for example, when there are at least 10 repetitions of each code, an error bar representing the standard deviation between a point in a CM and the corresponding point in the mean CM would be approximately the size of the characters used to label the codes in Fig. 5A. Clearly, by the time there have been fifty repetitions of each code, CMs generated from different random walks are nearly identical.

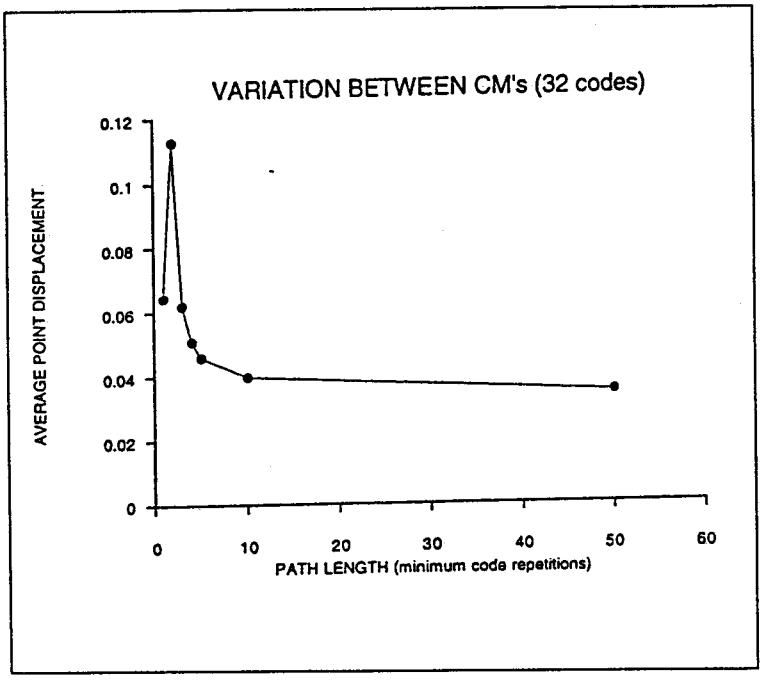

Figure 4 : This plot shows the average distance between the points in the continuity maps and their corresponding mean positions as a function of the minimum number of repetitions of each code in the path. A small average distance indicates that continuity maps made from different random walks are similar.

4. Evaluating the continuity map

The crucial comparison to be made is between the relative positions of the codes shown in the CM in Fig. 5A and the corresponding positions in the known AM shown in Fig. 5B. The position of a code, code 7 for example, in Fig. 5B is the mean of all the tongue positions (from the articulator map in Fig. 2) that produced a sound encoded as 7. The CM has been rotated and scaled to best fit the mean tongue positions, but the relative positions of the codes in the CM were not changed.

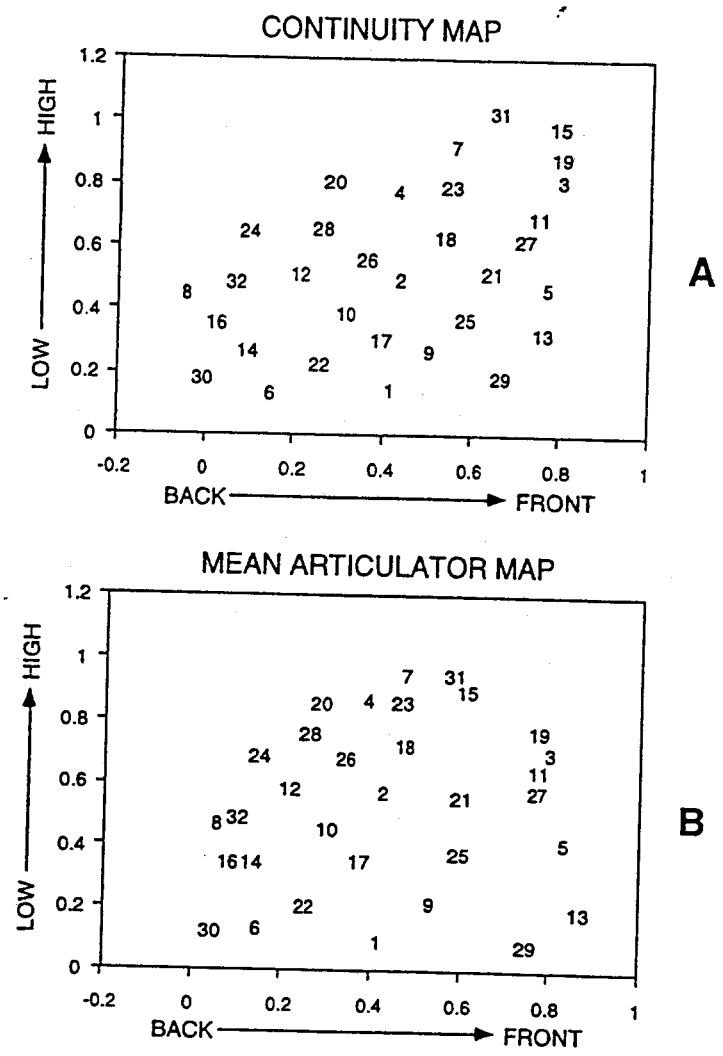

Figure 5 : A) The continuity map (CM) showing estimated tongue positions. The CM has been rotated, reflected, scaled, and translated to best fit the mean articulator map shown in 5B, but the relative positions of the points in the continuity map have not been changed.
B) The means of all the tongue positions that give rise to each code.

While the CM does show signs of non-uniform stretching relative to the plot of mean tongue positions, the relative positions of the codes are clearly similar in the two plots. The stretching can be attributed mostly to the thinness of the isocode regions when the tongue is extremely far forward. Since each isocode region is one transition away from its nearest neighbors, MDS tries to make the distances between neighboring isocode regions approximately equal. This means that the distance between neighboring large isocode regions should be about the same as between neighboring small isocode regions. Thus, the thin isocode regions that occur when the tongue is fronted are represented as taking up relatively larger regions in the CM than in the AM, distorting the CM relative to the AM.

Despite the distortions, the x-axis of the CM correlates well with the fronting axis of Fig. 5B as seen in Fig. 6A, which plots the position of the codes on the x-axis of the CM versus the fronting axis of Fig. 5B. The rank-order correlation between the positions is 0.98, showing that the position of a code in the CM can give us information about the relative fronting of the tongue. Similarly, the y-axis of the CM is compared to the height axis of Fig. 5B in Fig. 6B. The rank order correlation between the height given by the CM and the actual height is 0.97.

Similar results were found for three different codebooks with 64 codes and one codebook with 128 codes. Both the 64- and 128-code books had disjoint isocode regions when the tongue was low and very far back, in nearly the same areas as in the 32-code book. In the cases where disjoint isocode regions did occur, they were still relatively close together. As before, the correlations between estimated and actual articulator positions were high (0.95 or above, median correlation = 0.98) and the particular random walk used to make the continuity maps made very little difference as long as the random walk contained at least 10 repetitions of each code.

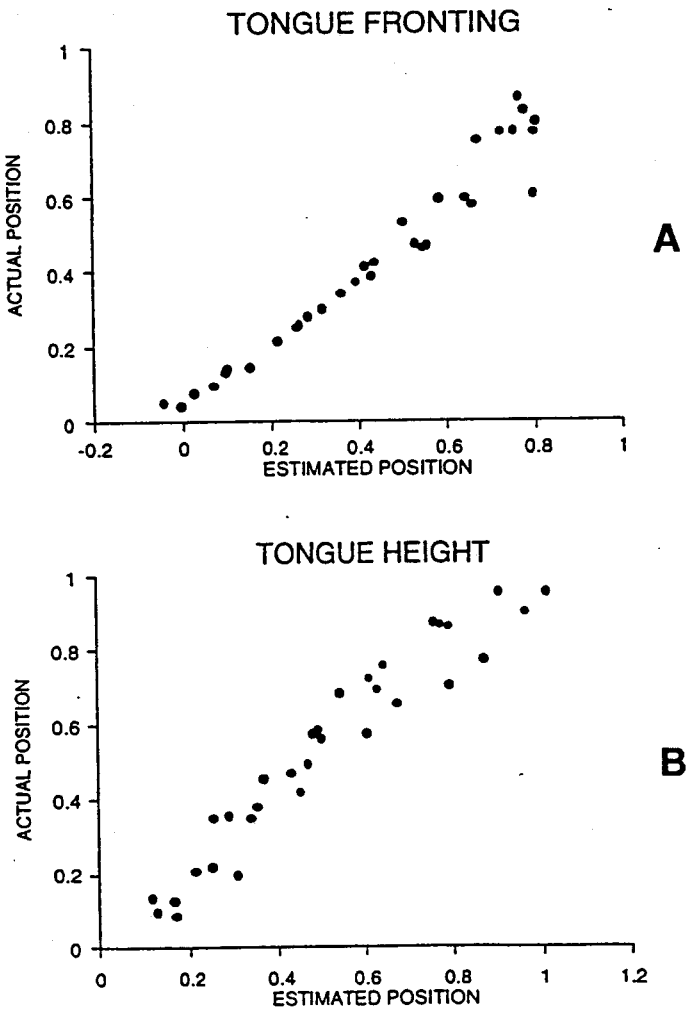

Figure 6 : A) Each point represents a single code ; the x-axis is the position of the code on the fronting axis of the continuity map and the y-axis shows the position of the code on the fronting axis of the mean articulator map. B) The continuity map height positions are compared to the mean articulator map height positions.

5. Discussion

All thirty of the continuity maps created from random walks with at least 10 repetitions of each code (this includes those with at least 50 repetitions of each code) gave good estimates of the relative locations of the mean articulator positions. The high correlations between the continuity maps and the average tongue positions clearly show that the continuity maps can be used to estimate the relative locations of the mean tongue positions for this synthesized data set. Of course, the ability of the continuity maps to represent the relative tongue positions also depends on how well the centroids of the isocode regions approximate the actual tongue positions. Since

the ability to estimate the mean tongue positions stays approximately constant as the number of codes increases, but the mean tongue positions become better estimates of the actual tongue positions, the accuracy of the estimates of relative tongue positions increases as the number of codes increases.

The consistently high correlations found with different VQ codebooks were surprising because, although the positions of codes in the continuity maps should be topologically similar to the positions of the centers of gravity of the isocode regions, the positions can be uncorrelated even if the two maps are topologically identical. Non-uniform stretching of one map relative to the other can decrease the correlation while maintaining topological similarity. In this study, some non-uniform stretching was found, particularly for front tongue positions, but the effect on the overall relative positions was small. The non-uniform stretching may be more prominent in continuity maps of natural speech.

Once a continuity map (CM) is created from training data, it can be used to give relative articulator position estimates for subsequent speech, without the algorithm ever getting any information about the absolute positions of the articulators. One possible use for the continuity mapping technique would be training the deaf to speak. For example, the algorithm could be used to create a continuity map from recordings of an instructor's voice. Once the continuity map is made, new speech sounds made by the instructor could be vector quantized, and the position of the vector quantization code in the CM could be used as an estimate of the instructor's articulator configuration. The instructor's articulation could then be displayed on a computer screen for the students to imitate. While only the relative positions are recovered from the technique described here, the absolute positions of the articulators can presumably be determined from only a few examples of acoustic signals created from known articulator positions, because only rotation and scaling information is needed to get the absolute positions from the relative positions.

A weakness of continuity mapping is that it only uses information from one short-time window of speech to determine articulator positions. This will make the technique less robust under noisy conditions. By treating the CM as a hidden Markov model (Huang *et al.*, 1990), it should be possible to use information from several windows of speech. One way to do so would be to treat the VQ codes as hidden Markov model states, then estimate transition probabilities between each of the codes in the CM and find the probability distributions of the observed acoustic vectors around the VQ reference vectors (the prototype vectors used in the nearest neighbor categorization). After making these extensions, it should be possible to calculate the path through the CM with the highest probability of creating an observed acoustic sequence. Research in this direction will have to address the computational problems of learning the transition probabilities for such a large network (normal hidden Markov models have many fewer possible transitions).

There are two main conclusions to draw from these results. The first is that, even though the data set contained a few cases where different articulator positions created the same derived acoustic parameters, there was enough information in the data set to find a rough mapping from acoustic information to the simulated articulator positions. If this were not the case, the continuity mapping procedure could not have found the mapping. The second conclusion is about the technique itself: using only unsupervised learning, the continuity mapping technique was able to recover information about the positions of moving objects. This suggests that continuity mapping may have applications beyond speech (Hogden *et al.*, 1992b give an example), since objects in the world move continuously and we often need to obtain knowledge about their physical positions from sensory information.

References

Atal, B.S. (1975). Towards determining articulator positions from the speech signal. In G. Fant (Ed.), *Proceedings of the 1974 Stockholm Speech Communication Seminar* (p. 1-9). New York : Wiley.

Atal, B.S., Chang, J.J., Mathews, M. V. & Tukey, J.W. (1978). Inversion of articulatory-to-acoustic transformation in the vocal tract by a computer-sorting technique. *Journal of the Acoustical Society of America, 63* (5), 1535-1555.

Boë, L.J., Perrier, P. & Bailly, G. (1992). The geometric vocal tract variables controlled for vowel production : proposals for constraining acoustic-to-articulatory conversion. *Journal of Phonetics, 20,* 27-38.

Coker, C. (1976). A model of articulatory dynamics and control. *Proceedings of the IEEE, 64* (4), 452-460.

Dillon, W. & Goldstein, M. (1984). *Multivariate Analysis : Methods and Applications.* New York : John Wiley & Sons.

Fant, G. (1970). *Acoustic Theory of Speech Production* (2nd ed.). The Hague : Mouton & Co.

Flanagan, J. (1972). *Speech Analysis, Synthesis, and Perception* (2nd ed.). New York : Springer-Verlag.

Fowler, C. & Turvey, M. (1980). Immediate compensation in bite-block speech. *Phonetica, 37,* 306-326.

Gower, J. (1975). Generalized Procrustes analysis. *Psychometrika, 40* (1), 33-51.

Gray, R. (1984). Vector Quantization. *IEEE Acoustics, Speech, and Signal Processing Magazine,* 4-29.

Harshman, R., Ladefoged, P. & Goldstein, L. (1977). Factor analysis of tongue shapes. *Journal of the Acoustical Society of America, 62* (3), 693-707.

Hogden, J. (1991) *Low-dimensional phoneme mapping using a continuity constraint.* Doctoral Dissertation, Stanford University.

Hogden, J., Lofquist, A., Gracco, V., Oshima, K., Rubin, P. & Saltzman, E. (1993). Inferring articulator positions from acoustics : an electromagnetic midsagittal articulometer experiment. *Journal of the Acoustical Society of America, 94* (3), 1764 (A).

Hogden, J., Rubin, P. & Saltzman, E. (1992a). An unsupervised method for learning to track tongue position from an acoustic signal. *Journal of the Acoustical Society of America, 91* (4), 2443 (A).

Hogden, J., Saltzman, E. & Rubin, P. (1992b). Unsupervised neural networks that use a continuity constraint to track articulators. *Journal of the Acoustical Society of America, 92* (4), 2477 (A).

Huang, X.D., Ariki, Y. & Jack, M. (1990). *Hidden Markov Models for Speech Recognition.* Edinburgh : Edinburgh University Press.

Jordan, M. & Rumelhart, D. (1992). Forward models : supervised learning with a distal teacher. *Cognitive Science, 16,* 307-354.

Kawato, M. (1989). Motor theory of speech perception revisited from minimum torque-change neural network model. In *8th Symposium on Future Electron Devices,* (p. 141-150).

Kohonen, T. (1988). The neural phonetic typewriter. *Computer,* 11-22.

Kuc, R., Tutuer, F. & Vaisnys, J.R. (1985). Determining vocal tract shape by applying dynamic constraints. In *Proceedings of the International Conference on Acoustics Speech & Signal Processing,* 1101-1104, New York : IEEE.

Ladefoged, P., Harshman, R., Goldstein, L. & Rice, L. (1978). Generating vocal tract shapes from formant frequencies. *Journal of the Acoustical Society of America, 64* (4), 1027-1035.

Lederman SSS (1984). Orthogonal Procrustes Analysis. In E. Lloyd (Ed.), *Handbook of Applicable Mathematics* (p. 761-781). New York : John Wiley & Sons.

Liberman, A., Cooper, F., Shankweiler, D. & Studdert-Kennedy, M. (1967). Perception of the speech code. *Psychological Review, 74* (6), 431-461.

Liberman, A. & Mattingly, I. (1985). The motor theory of speech perception revised. *Cognition, 21,* 1-36.

Lindblom, B., Lubker, J. & Gay, T. (1979). Formant frequencies of some fixed-mandible vowels and a model of speech motor programming by predictive simulation. *Journal of Phonetics, 7,* 146-161.

Maeda, S. (1979). An articulatory model of the tongue based on a statistical analysis. *Journal of the Acoustical Society of America, 65,* S22.

Maeda, S. (1989). Compensatory articulation during speech : evidence from the analysis and synthesis of vocal-tract shapes using an articulatory model. In *NATO AI Series* Kluwer Academic Publishers.

Markel, J. & Gray, A. (1976). *Linear Prediction of Speech.* New York : Springer-Verlag.

Mcgowan, R. (1994). Recovering articulator movement from formant frequency trajectories using task dynamics and a genetic algorithm : Preliminary tests. *Speech Communication, 14,* 19-48.

Mermelstein, P. (1973). Articulatory model for the study of speech production. *Journal of the Acoustical Society of America, 53* (4), 1070-1082.

Morrish, K., Stone, M., Shawker, T. & Sonies, B. (1985). Distinguishability of tongue shape during vowel production. *Journal of Phonetics, 13*, 189-203.

O'shaughnessy, D. (1987). *Speech Communication : Human and Machine.* New York : Addison-Wesley.

Papcun, G., Hotchberg, J., Thomas, T., Laroche, F., Zacks, J. & Levy, S. (1992). Inferring articulation and recognizing gestures from acoustics with a neural network trained on X-ray microbeam data. *Journal of the Acoustical Society of America, 92* (2), 688-700.

Rahim, M.G., Kleijn, W.B., Schroeter, J. & Goodyear, C.C. (1991). Acoustic to articulatory parameter mapping using an assembly of neural networks. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 485-488.

Rubin, P., Baer, T. & Mermelstein, P. (1981). An articulatory synthesizer for perceptual research. *Journal of the Acoustical Society of America, 70* (2), 321-328.

Schroeter, J., Meyer, P. & Parthasarathy, S. (1990). Evaluation of improved articulatory codebooks and codebook access distance measures. *Proceedings of the IEEE International Conference on Acoustics, Speech & Signal Processing*, 393-396.

Schroeter, J. & Sondhi, M. (1992). Speech coding based on physiological models of speech production. In S. Furui & M. Sondhi (Eds.), *Advances in Speech Signal Processing* (p. 231-267). New York : Marcel Dekker, Inc.

Shirai, K. & Kobayashi, T. (1986). Estimating articulatory motion from speech wave. *Speech Communication, 5*, 159-170.

Smith, N. (1973). *The acquisition of phonology : A case study.* Cambridge : Cambridge University Press.

Sondhi, M. (1979). Estimation of vocal tract areas : the need for acoustical measurements. *IEEE Trans. ASSP, 27* (3), 268-273.

Stevens, K. & House, A. (1955). Development of a quantitative description of vowel articulation. *Journal of the Acoustical Society of America, 27* (3), 484-493.