

1. INTRODUCTION

Voice morphing means the transition of one speech signal into another. Like image morphing, speech morphing aims to preserve the shared characteristics of the starting and final signals, while generating a smooth transition between them. Speech morphing is analogous to image morphing. In image morphing the in-between images all show one face smoothly changing its shape and texture until it turns into the target face. It is this feature that a speech morph should possess. One speech signal should smoothly change into another, keeping the shared characteristics of the starting and ending signals but smoothly changing the other properties. The major properties of concern as far as a speech signal is concerned are its pitch and envelope information. These two reside in a convolved form in a speech signal. Hence some efficient method for extracting each of these is necessary. We have adopted an uncomplicated approach namely cepstral analysis to do the same. Pitch and formant information in each signal is extracted using the cepstral approach. Necessary processing to obtain the morphed speech signal include methods like Cross fading of envelope information, Dynamic Time Warping to match the major signal features (pitch) and Signal Re-estimation to convert the morphed speech signal back into the acoustic waveform.

2. AN INTROSPECTION OF THE MORPHING PROCESS

Speech morphing can be achieved by transforming the signal's representation from the acoustic waveform obtained by sampling of the analog signal, with which many people are familiar with, to another representation. To prepare the signal for the transformation, it is split into a number of 'frames' - sections of the waveform. The transformation is then applied to each frame of the signal. This provides another way of viewing the signal information. The new representation (said to be in the frequency domain) describes the average energy present at each frequency band.

Further analysis enables two pieces of information to be obtained: pitch information and the overall envelope of the sound. A key element in the morphing is the manipulation of the pitch information. If two signals with different pitches were simply cross-faded it is highly likely that two separate sounds will be heard. This occurs because the signal will have two distinct pitches causing the auditory system to perceive two different objects. A successful morph must exhibit a smoothly changing pitch throughout. The pitch information of each sound is compared to provide the best match between the two signals' pitches. To do this match, the signals are stretched and compressed so that important sections of each signal match in time. The interpolation of the two sounds can then be performed which creates the intermediate sounds in the morph. The final stage is then to convert the frames back into a normal waveform.

However, after the morphing has been performed, the legacy of the earlier analysis becomes apparent. The conversion of the sound to a representation in which the pitch and spectral envelope can be separated loses some information. Therefore, this information has to be re-estimated for the morphed sound. This process obtains an acoustic waveform, which can then be stored or listened to.

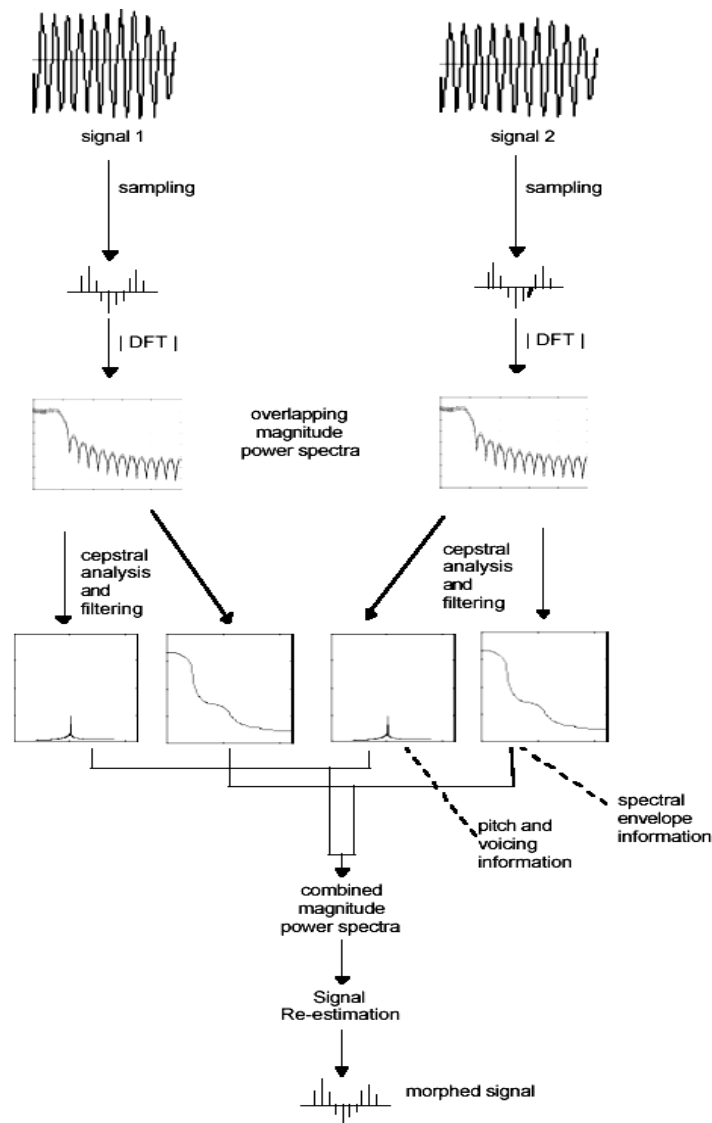


Figure 2.1 Schematic block diagram of the speech morphing process

3. MORPHING PROCESS: A COMPREHENSIVE ANALYSIS

The algorithm to be used is shown in the simplified block diagram given below. The algorithm contains a number of fundamental signal processing methods including sampling, the discrete Fourier transform and its inverse, cepstral analysis. However the main processes can be categorized as follows.

- I. Preprocessing or representation conversion: This involves processes like signal acquisition in discrete form and windowing.
- II. Cepstral analysis or Pitch and Envelope analysis: This process will extract the pitch and formant information in the speech signal.
- III. Morphing which includes Warping and interpolation.
- IV. Signal re-estimation.

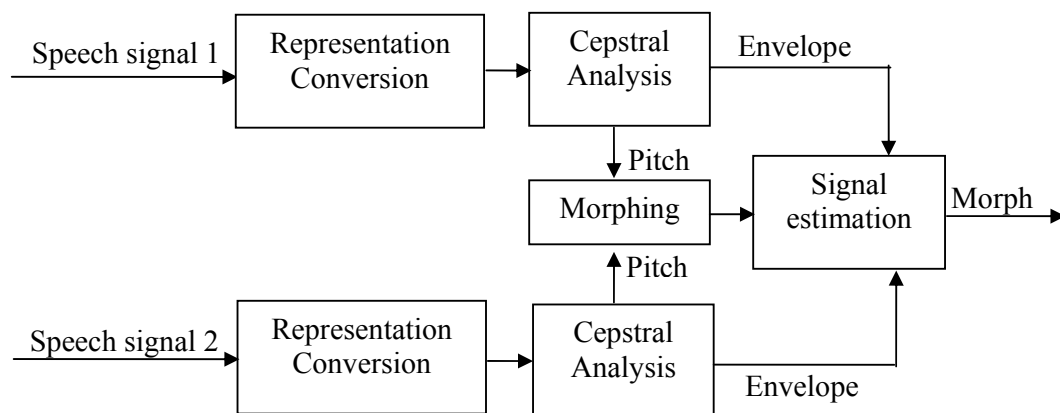


Fig 3.1: Block diagram of the simplified speech morphing algorithm.

3.1 Acoustics of speech production

Speech production can be viewed as a filtering operation in which a sound source excites a vocal tract filter. The source may be periodic, resulting in voiced speech, or noisy and a periodic, causing unvoiced speech. As a periodic signal, voiced speech has a spectra consisting of harmonics of the fundamental frequency of the vocal cord vibration; this frequency often abbreviated as F_0 , is the physical aspect of the speech signal corresponding to the perceived pitch. Thus pitch refers to the fundamental frequency of the vocal cord vibrations or the resulting periodicity in the speech signal. This F_0 can be determined either from the periodicity in the time domain or from the regularly spaced harmonics in the frequency domain.

The vocal tract can be modeled as an acoustic tube with resonances, called formants, and anti resonances. (The formants are abbreviated as F_1 , where F_1 is the formant with the lowest center frequency.) Moving certain structures in the vocal tract alters the shape of the acoustic tube, which in turn changes its frequency response. The filter amplifies energy at and near formant frequencies, while attenuating energy around anti resonant frequencies between the formants.

The common method used to extract pitch and formant frequencies is the spectral analysis. This method views speech as the output of a liner, time-varying system (vocal tract) excited by either quasiperiodic pulses or random noise. Since the speech signal is the result of convolving excitation and vocal tract sample response, separating or “deconvolving” the two components can be used. In

general, deconvolution of the two signals is impossible, but it works for speech, because the two signals have quite different spectral characteristics. The deconvolution process transforms a product of two signals into a sum of two signals. If the resulting summed signals are sufficiently different spectrally, they may be separated by linear filtering. Now we present a comprehensive analysis of each of the processes involved in morphing with the aid of block diagrams wherever necessary.

3.2 Preprocessing

This section shall introduce the major concepts associated with processing a speech signal and transforming it to the new required representation to affect the morph. This process takes place for each of the signals involved with the morph.

3.2.1 Signal Acquisition

Before any processing can begin, the sound signal that is created by some real-world process has to be ported to the computer by some method. This is called sampling. A fundamental aspect of a digital signal (in this case sound) is that it is based on processing sequences of samples. When a natural process, such as a musical instrument, produces sound the signal produced is analog (continuous-time) because it is defined along a continuum of times. A discrete-time signal is represented by a sequence of numbers - the signal is only defined at discrete times. A digital signal is a special instance of a discrete-time signal - both time and amplitude are discrete. Each discrete representation of the signal is termed a sample.

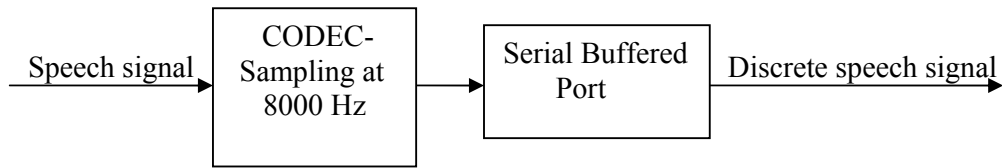


Fig 3.2: Signal acquisition

The input speech signals are taken using MIC and CODEC. The analog speech signal is converted into the discrete form by the inbuilt CODEC TLC320AD535 present onboard and stored in the processor memory. This completes the signal acquisition phase.

3.2.2 Windowing

A DFT (Discrete Fourier Transformation) can only deal with a finite amount of information. Therefore, a long signal must be split up into a number of segments. These are called frames. Generally, speech signals are constantly changing and so the aim is to make the frame short enough to make the segment almost stationary and yet long enough to resolve consecutive pitch harmonics. Therefore, the length of such frames tends to be in the region of 25 to 75 milli seconds. There are a number of possible windows. A selection is:

The Hanning window

$$W(n) = 0.5 - 0.5 \cos(2\pi n/N) \text{ when } 0 \leq n \leq N,$$

$$= 0 \text{ otherwise} \dots\dots\dots 3.1$$

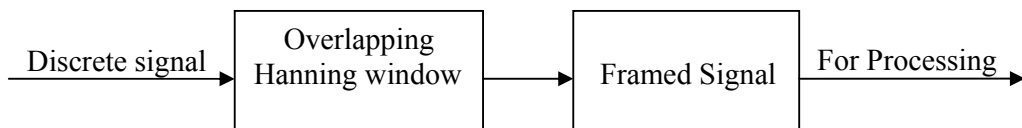


Fig 3.3: Windowing

The frequency-domain spectrum of the Hamming window is much smoother than that of the rectangular window and is commonly used in spectral analysis. The windowing function splits the signal into time-weighted frames.

However, it is not enough to merely process contiguous frames. When the frames are put back together, modulation in the signal becomes evident due to the windowing function. As the weighting of the window is required, another means of overcoming the modulation must be found. A simple method is to use overlapping windows. To obtain a number of overlapping spectra, the window is shifted along the signal by a number of samples (no more than the window length) and the process is repeated. Simply put, it means that as one frame fades out, its successor fades in. It has the advantage that any discontinuities are smoothed out. However, it does increase the amount of processing required due to the increase in the number of frames produced.

3.3 Morphing

3.3.1 Matching and Warping: Background theory

Both signals will have a number of 'time-varying properties'. To create an effective morph, it is necessary to match one or more of these properties of each signal to those of the other signal in some way. The property of concern is the pitch of the signal - although other properties such as the amplitude could be used - and will have a number of features. It is almost certain that matching features do not occur at exactly the same point in each signal. Therefore, the feature must be moved to some point in between the position in the first sound

and the second sound. In other words, to smoothly morph the pitch information, the pitch present in each signals needs to be matched and then the amplitude at each frequency cross-faded. To perform the pitch matching, a pitch contour for the entire signal is required. This is obtained by using the pitch peak location in each cepstral pitch slice.

Consider the simple case of two signals, each with two features occurring in different positions as shown in the figure below.

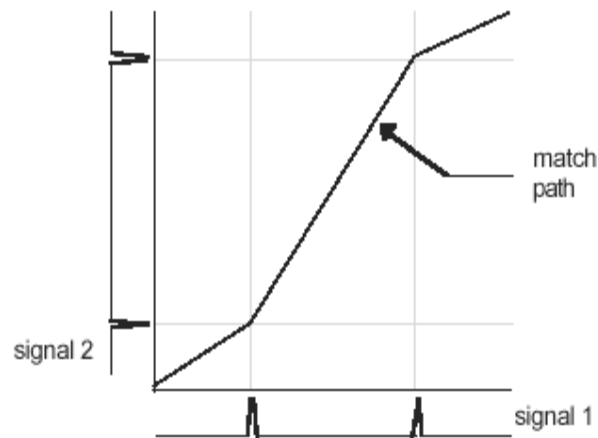


Figure 3.4: The match path between two signals with differently located features

The match path shows the amount of movement (or warping) required in order aligning corresponding features in time. Such a match path is obtained by **Dynamic Time Warping (DTW)**.

3.3.2 Dynamic Time Warping

Speaker recognition and speech recognition are two important applications of speech processing. These applications are essentially pattern recognition problems, which is a large field in itself. Some

Automatic Speech Recognition (ASR) systems employ time normalization. This is the process by which time-varying features within the words are brought into line. The current method is time-warping in which the time axis of the unknown word is non-uniformly distorted to match its features to those of the pattern word. The degree of discrepancy between the unknown word and the pattern – the amount of warping required to match the two words - can be used directly as a distance measure. Such time-warping algorithm is usually implemented by dynamic programming and is known as Dynamic Time Warping. Dynamic Time Warping (DTW) is used to find the best match between the features of the two sounds - in this case, their pitch. To create a successful morph, major features, which occur at generally the same time in each signal, ought to remain fixed and intermediate features should be moved or interpolated. DTW enables a match path to be created. This shows how each element in one signal corresponds to each element in the second signal.

In order to understand DTW, two concepts need to be dealt with:

Features: The information in each signal has to be represented in some manner.

Distances: some form of metric has to be used in order to obtain a match path. There are two types:

1. **Local:** a computational difference between a feature of one signal and a feature of the other.
2. **Global:** the overall computational difference between an entire signal and another signal of possibly different length.

Feature vectors are the means by which the signal is represented and are created at regular intervals throughout the signal.

In this use of DTW, a path between two pitch contours is required. Therefore, each feature vector will be a single value. In other uses of DTW, however, such feature vectors could be large arrays of values. Since the feature vectors could possibly have multiple elements, a means of calculating the local distance is required. The distance measure between two feature vectors is calculated using the Euclidean distance metric. Therefore the local distance between feature vector x of signal 1 and feature vector y of signal 2 is given by,

$$d(x, y) = \sqrt{\sum_i (x_i - y_i)^2} \dots\dots\dots 3.3$$

As the pitch contours are single value feature vectors, this simplifies to,

$$d(x, y) = |x - y| \dots\dots\dots 3.4$$

The global distance is the overall difference between the two signals. Audio is a time- dependent process. For example, two audio sequences may have different durations and two sequences of the sound with the same duration are likely to differ in the middle due to differences in sound production rate. Therefore, to produce a global distance measure, time alignment must be performed - the matching of similar features and the stretching and compressing, in time, of others. Instead of considering every possible match path which would be very inefficient, a number of constraints are imposed upon the matching process.

3.4.3 The DTW Algorithm

The basic DTW algorithm is symmetrical - in other words, every frame in signals must be used. The constraints placed upon the matching process are:

- Matching paths cannot go backwards in time;
- Every frame in each signal must be used in a matching path;
- Local distance scores are combined by adding to give a global distance.

If $D(i,j)$ is the global distance up to (i,j) and the local distance at (i,j) is given by $d(i,j)$

$$D(i,j) = \min[D(i-1,j-1), D(i-1,j), D(i,j-1)] + d(i,j) \quad \dots\dots\dots 3.$$

5

Computationally, the above equation is already in a form that could be recursively programmed. However, unless the language is optimized for recursion, this method can be slow even for relatively small pattern sizes. Another method, which is both quicker and requires less memory storage, uses two nested for loops. This method only needs two arrays that hold adjacent columns of the time-time matrix. In the following explanation, it is assumed that the array notation is of the form $0 \dots N-1$ for an array of length N . The only directions in which the match path can move when at (i, j) in the time-time matrix are given in figure 3.8 below.

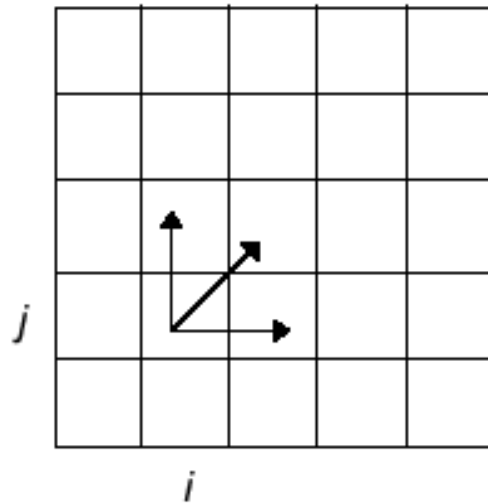


Figure 3.5: Time –Time matrix

The three possible directions in which the best match path may move from cell (i, j) in symmetric DTW.

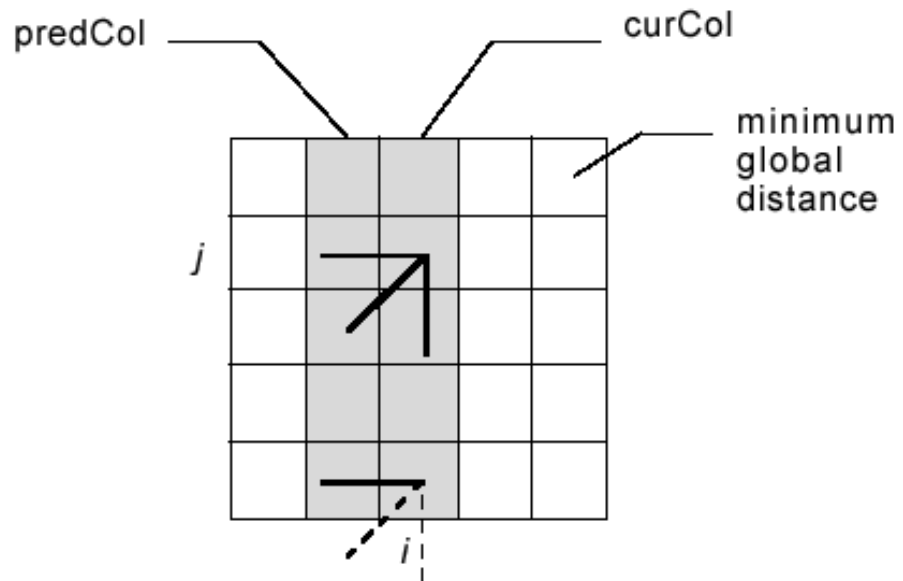


Figure 3.6: Minimum cost path

The cells at (i,j) and $(i,0)$ have different possible originator cells. The path to $(i, 0)$ can only originate from $(i-1, 0)$. However, the path to (i,j) can originate from the three standard locations as shown in the figure 3.9 above.

The algorithm to find the least global cost is:

- I. Calculate column 0 starting at the bottom most cell. The global cost to this cell is just its local cost. Then, the global cost for each successive cell is the local cost for that cell plus the global cost to the cell below it. This is called the predCol (predecessor column).
- II. Calculate the global cost to the first cell of the next column (the curCol). This local cost for the cell plus the global cost to the bottom most cell of the previous column.
- III. Calculate the global cost of the rest of the cells of curCol. For example, at (i,j) this is the local distance at (i,j) plus the minimum global cost at either $(i-1,j)$, $(i-1,j-1)$ or $(i,j-1)$.
- IV. curCol is assigned to predCol and repeat from step 2 until all columns have been calculated.
- V. Global cost is the value stored in the top most cell of the last column.

However, in the case of audio morphing, it is not the minimum global distance itself, which is of interest but the path to achieve. In other words, a back trace array must be kept with entries in the array pointing to the preceding point in the path. Therefore, a second algorithm is required to extract the path.

The path has three different types of direction changes:

- Vertical
- Horizontal
- Diagonal

The back trace array will be of equal size to that of the time-time matrix. When the global distance to each cell, say (i,j) , in the time-time matrix is calculated, its predecessor cell is known - it's the cell out of $(i-1,j)$, $(i-1,j-1)$ or $(i,j-1)$ with the lowest global cost. Therefore, it is possible to record in the backtrace array the predecessor cell using the following notation (for the cell (i,j)):

- 1) $(i-1, j-1)$ -- Diagonal
- 2) $(i-1, j)$ -- Horizontal
- 3) $(i, j-1)$ -- Vertical

4	3	1	2	1	1
3	3	3	2	1	2
2	3	1	2	3	1
1	3	2	1	2	1
0	0	2	2	2	2
	0	1	2	3	4

Fig 3.7: A sample back trace array with each cell containing a number, which represents the location of the predecessor cell in the lowest global path distance to that cell.

For the example in Figure above, the 2D array would be

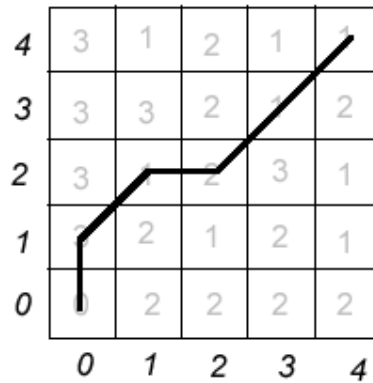


Figure 3.8: The sample back trace array with the calculated path overlaid

At this stage, we now have the match path between the pitches of the two signals and each signal in the appropriate form for manipulation. The next stage is to then produce the final morphed signal.

4. MORPHING STAGE

Now we shall give a detailed account of how the morphing process is carried out. The overall aim in this section is to make the smooth transition from signal 1 to signal 2. This is partially accomplished by the 2D array of the match path provided by the DTW. At this stage, it was decided exactly what form the morph would take. The implementation chosen was to perform the morph in the duration of the longest signal. In other words, the final morphed speech signal would have the duration of the longest signal. In order to accomplish this, the 2D array is interpolated to provide the desired duration.

However, one problem still remains: the interpolated pitch of each morph slice. If no interpolation were to occur then this would be equivalent to the warped cross-fade which would still be likely to result in a sound with two pitches. Therefore, a pitch in-between those of the first and second signals must be created. The precise properties of this manufactured pitch peak are governed by how far through the morph the process is. At the beginning of the morph, the pitch peak will take on more characteristics of the signal 1 pitch peak - peak value and peak location - than the signal 2 peak. Towards the end of the morph, the peak will bear more resemblance to that of the signal 2 peaks. The variable l is used to control the balance between signal 1 and signal 2. At the beginning of the morph, l has the value 0 and upon completion, l has the value 1. Consider the example in Figure 4.6. This diagram shows a sample cepstral slice with the pitch peak area highlighted. Figure 4.7 shows another sample cepstral slice, again with

the same information highlighted. To illustrate the morph process, these two cepstral slices shall be used.

There are three stages:

1. Combination of the envelope information;
2. Combination of the pitch information residual - the pitch information excluding the pitch peak;
3. Combination of the pitch peak information.

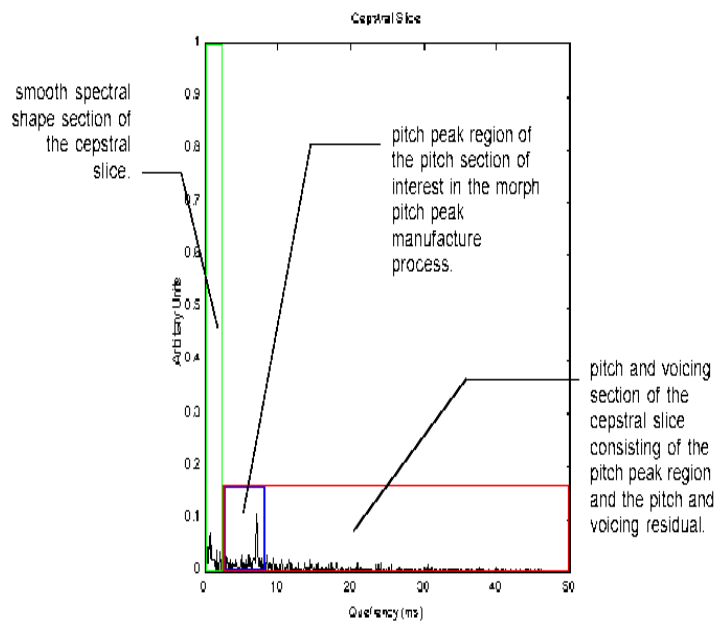


Figure 4.1. A second sample cepstral slice with the pitch p

4.1 Combination of the envelope information

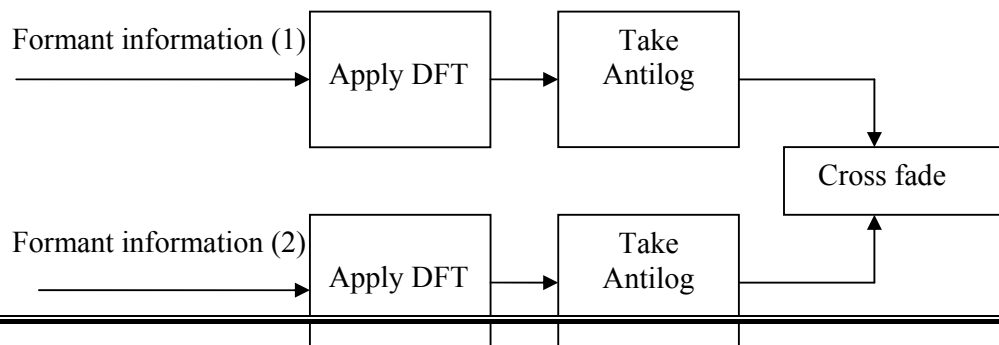


Figure 4.2: Cross fading of the formants.

We can say that that the best morphs are obtained when the envelope information is merely cross-faded, as opposed to employing any pre-warping of features, and so this approach is adopted here. In order to cross-fade any information in the cepstral domain, care has to be taken. Due to the properties of logarithms employed in the cepstral analysis stage, multiplication is transformed into addition. Therefore, if a cross-fade between the two envelopes were attempted, multiplication would in fact take place. Consequently, each envelope must be transformed back into the frequency domain (involving an inverse logarithm) before the cross-fade is performed. Once the envelopes have been successfully cross-faded according to the weighting determined by l , the morphed envelope is once again transformed back into the cepstral domain. This new cepstral slice forms the basis of the completed morph slice.

4.2 Combination of the pitch information residual

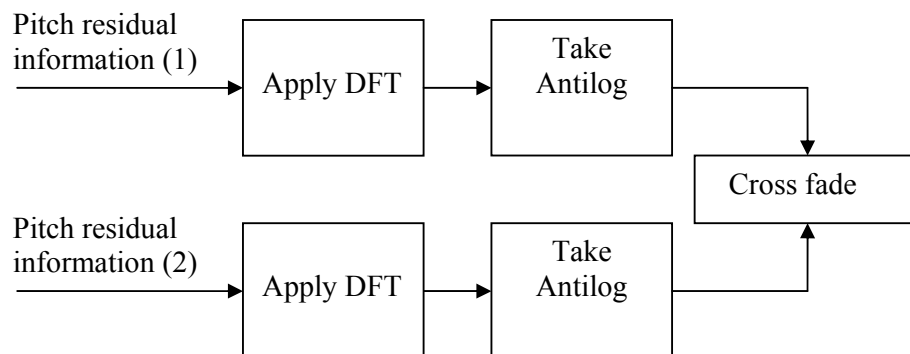


Figure 4.3: Cross fading of the Pitch information.

The pitch information residual is the pitch information section of the cepstral slice with the pitch peak also removed by liftering. To produce the morphed residual, it is combined in a similar way to that of the envelope information: no further matching is performed. It is simply transformed back into the frequency domain and cross-faded with respect to 1. Once the cross-fade has been performed, it is again transformed into the cepstral domain. The information is now combined with the new morph cepstral slice (currently containing envelope information). The only remaining part to be morphed is the pitch peak area.

4.3 Combination of the Pitch peak information

As stated above, in order to produce a satisfying morph, it must have just one pitch. This means that the morph slice must have a pitch peak, which has characteristics of both signal 1 and signal 2. Therefore, an artificial' peak needs to be generated to satisfy this requirement. The positions of the signal 1 and signal 2 pitch peaks are stored in an array (created during the pre-processing, above), which means that the desired pitch peak location can easily be calculated.

In order to manufacture the peak, the following process is performed,

- I. Each pitch peak area is liftered from its respective slice. Although the alignment of the pitch peaks will not match with respect to the cepstral slices, the pitch peak areas are liftered in such a way as to align the peaks with respect to the liftered area (see Figure 4.8).
- II. The two liftered cepstral slices are then transformed back into the frequency domain where they can be cross-faded with respect to

1. The cross-fade is then transformed back into the cepstral domain.

III. The morphed pitch peak area is now placed at the appropriate point in the morph cepstral slice to complete the process.

The morphing process is now complete. The final series of morphed cepstral slices is transformed back in to the frequency domain. All that remains to be done is re-estimate the waveform.

5. SUMMARIZED BLOCK DIAGRAM

The whole morphing process is summarized using the detailed block diagram shown below (figure 6.1).

6. FUTURE SCOPE

There are a number of areas in which further work should be carried out in order to improve the technique described here and extend the field of speech morphing in general. The time required to generate a morph is dominated by the signal re-estimation process. Even a small number (for example, 2) of iterations takes a significant amount of time even to re-estimate signals of approximately one second duration. Although in speech morphing, an inevitable loss of quality due to manipulation occurs and so less iteration are required, an improved re-estimation algorithm is required.

A number of the processes, such as the matching and signal re-estimation are very unrefined and inefficient methods but do produce satisfactory morphs. Concentration on the issues described above for further work and extensions to the speech morphing principle ought to produce systems which create extremely convincing and satisfying speech morphs.

Further extension to this work to provide the above functionality would create a powerful and flexible morphing tool. Such a tool would allow the user to specify at which points a morph was to start and finish the properties of the morph and also the matching function. With the increased user interaction in the process, a Graphical User Interface could be designed and integrated to make the package more 'user-friendly'. Such an improvement would immediate visual feedback (which is lacking in the current implementation) and possibly step by step guidance. Finally, this work has used spectrograms as the pitch and voicing and spectral envelope

representations. Although effective, further work ought to concentrate on new representations which enable further separation of information. For example, a new representation might allow the separation of the pitch and voicing.

The Speech morphing concept can be extended to include audio sounds in general. This area offers many possible applications including sound synthesis. For example, there are two major methods for synthesizing musical notes. One is to digitally model the sound's physical source and provide a number of parameters in order to produce a synthetic note of the desired pitch. Another is to take two notes which bound the desired note and use the principles used in speech morphing to manufacture a note which contains the shared characteristics of the bounding notes but whose other properties have been altered to form a new note. The use of pitch manipulation within the algorithm also has an interesting potential use. In the interests of security, it is sometimes necessary for people to disguise the identity of their voice. An interesting way of doing this is to alter the pitch of the sound in real-time using sophisticated methods.

7. CONCLUSION

The approach we have adopted separates the sounds into two forms: spectral envelope information and pitch and voicing information. These can then be independently modified. The morph is generated by splitting each sound into two forms: a pitch representation and an envelope representation. The pitch peaks are then obtained from the pitch spectrograms to create a pitch contour for each sound. Dynamic Time Warping of these contours aligns the sounds with respect to their pitches. At each corresponding frame, the pitch, voicing and envelope information are separately morphed to produce a final morphed frame. These frames are then converted back into a time domain waveform using the signal re-estimation algorithm.

In this seminar, only one type of morphing has been discussed - that in which the final morph has the same duration as the longest signal. Also we discuss the case of speech morphing in this seminar. But the work can be extended to include audio sounds as well. The longest signal is compressed and the morph has the same duration as the shortest signal (the reverse of the approach described here). If one signal is significantly longer than the other, two possibilities arise. However, according to the eventual use of the morph, a number of other types could be produced.

8. BIBLIOGRAPHY

- Alex Luscos and Pedro Cano 'VOICE MORPHING'
CYBER SPEAK VOL 2, Dec 2002
- www.voicemorphing.com
- www.acoustics.com
- <http://www.nillymoser.com>

ABSTRACT

Voice morphing means the transition of one speech signal into another. The new morphed signal will have the same information content as the two input speech signals but a different pitch, which is determined by the morphing algorithm. To do this, each signal's information has to be converted into another representation, which enables the pitch and spectral envelope to be encoded on orthogonal axes. Individual components of the speech signal are then matched and the signal's amplitudes are then interpolated to produce a new speech signal. This new signal's representation then has to be converted back to an acoustic waveform. This project vividly describes the representations of the signals required to affect the morph and also the techniques required to match the signal components, interpolate the amplitudes and invert the new signal's representation back to an acoustic waveform.

CONTENTS

1. INTRODUCTION
2. AN INTROSPECTION OF THE MORPHING PROCESS
3. MORPHING PROCESS: A COMPREHENSIVE ANALYSIS
 - 3.1 Acoustics of speech production
 - 3.2 Preprocessing
 - 3.2.1 Signal Acquisition
 - 3.2.2 Windowing
 - 3.3 Morphing
 - 3.3.1 Matching and Warping: Background theory
 - 3.3.2 Dynamic Time Warping
 - 3.3.3 The DTW Algorithm
4. MORPHING STAGE
 - 4.1 Combination of the envelope information
 - 4.2 Combination of the pitch information residual
 - 4.3 Combination of the Pitch peak information
5. SUMMARIZED BLOCK DIAGRAM
6. FUTURE SCOPE
7. CONCLUSION
8. REFERENCES

ACKNOWLEDGEMENT

I extend my sincere gratitude towards **Prof. P.Sukumaran** Head of Department for giving us his invaluable knowledge and wonderful technical guidance

I express my thanks to **Mr. Muhammed Kutty** our group tutor and also to our staff advisor **Ms. Biji Paul** for their kind co-operation and guidance for preparing and presenting this seminar.

I also thank all the other faculty members of AEI department and my friends for their help and support.