VOICE MORPHING USING LINEAR PREDICTIVE CODING

Guided by: Prof. V. M. Gadre

-Shruti Hiray(14D070016)
-Komal Saroya (14D070023)
-Nisha Dhake (14D070024)
-Himani (14D070047)

Abstract:

Aimed at configuring one's speech into words and voice so that we can morph the voice for several applications like security purpose, and implemented using linear predictive coding(LPC).

Speech is a collection of words shaped by voice.

Speech =Words + Voice

The components of speech are produced in different organs. To speak, air is first released over the vocal cords, which expand and contract to give the air column structure. This is termed as biological concept of words. The words are then passed through the vocal tract where they are shaped, giving them intonation.This shaping of the words is the biological concept of voice.

Source Filter Model states that:
The source,x(t) is simply a signal input to the filter and is called the **excitation signal** since it excites the vocal tract. Vocal tract is filter of a linear time-invariant system with impulse response h (t). This is called the **transfer function** of speech since it is what transfers the excitation signal to speech - it adds voice to words.
The output is given by y (t) = x (t) ∗ h (t)
Where,
x(t)---->Excitation signal
h(t)---->Transfer function

Above two components can be modelled using several audio signal processing techniques like cepstrum, linear predictive coding, etc.

Linear Predictive Coding (or LPC) is a method of predicting a sample of a speech signal based on several previous samples. Predicting the nth sample in a sequence of speech samples is represented by the weighted sum of the p previous samples:

$$s'[n] = \sum_{k=1}^{p} a_k * s[n-k]$$

The number of samples (p) is referred to as the order of the LPC. As p approaches infinity, we should be able to predict the nth sample exactly. However, p is usually on the order of ten to twenty, where it can provide an accurate enough representation with a limited cost of computation. The weights on the previous samples (a_k) are chosen in order to minimize the squared error between the real sample and its predicted value. Thus, we want the error signal e[n], which is sometimes referred to as the **LPC residual**, to be as small as possible:

$$e[n] = s[n] - s'[n] = s[n] - \sum_{k=1}^{p} a_k * s[n-k]$$

Taking z-transform both sides will give:

$$E(z) = S(z)*(1 - \sum_{k=1}^{p} a_k * z^{-k})$$

$$E(z) = S(z)*A(z) \text{ or}$$
$$S(z) = E(z)*(1/A(z))$$

We can represent our original speech signal S(z) as the product of the error signal E(z) and filter 1 / A(z).
Therefore, components of speech signal, s[n] are given as:
Excitation signal =e[n]
Transfer function=(1/A(z))

The transfer function 1/A(z) represents an all-pole digital filter, where the a_k coefficients correspond to the poles in the filter's z-plane.

Note that the roots of the A(z) polynomial must all lie within the unit circle to ensure stability of this filter.

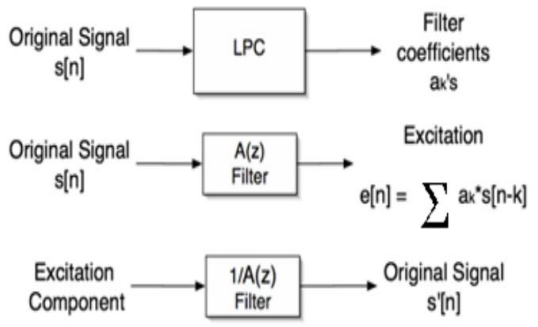


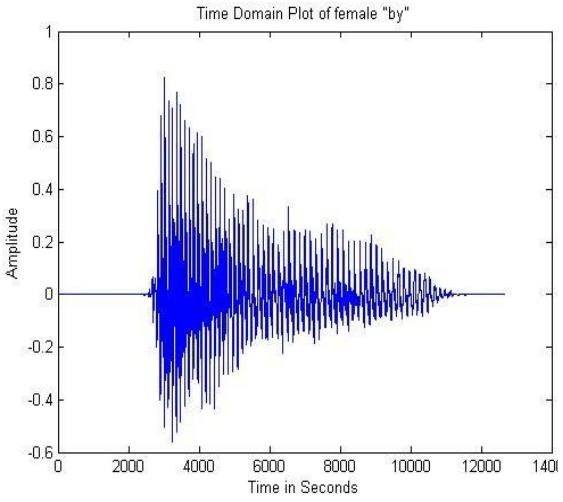
Fig1. Voice Conversion Algorithm

In speech processing, a pre-emphasis filter should be applied to the input signal before the LPC analysis and during the reconstruction following the LPC analysis, a de-emphasis filter should be applied in order to reverse the effects of pre-emphasis.

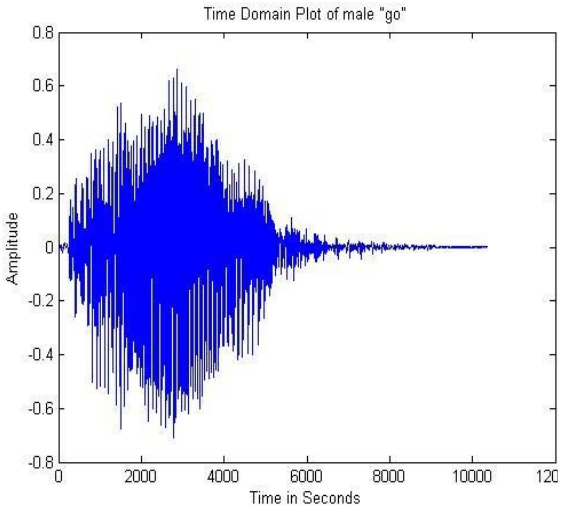
Pre- and de-emphasis are necessary because, in the spectrum of a human speech signal, the energy in the signal decreases as the frequency increases. Pre-emphasis increases the energy in parts of the signal by an amount inversely proportional to its frequency. Thus, as the frequency increases, pre-emphasis raises the energy of the speech signal by an increasing amount. This process therefore serves to flatten the signal so that the resulting spectrum consists of formants of similar heights. The flatter spectrum allows the LPC analysis to more accurately model the speech segment. Without pre-emphasis,
the linear prediction would incorrectly focus on the lower-frequency components of speech, losing important information about

Simulation Plots:

Source speech signal:
word *"by"* in female voice



Target speech signal:
word *"go"* in male voice

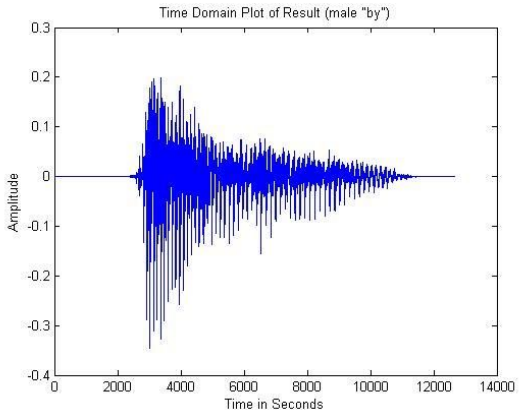


Implementing voice morphing algorithm developed on above source and target signal will result in:

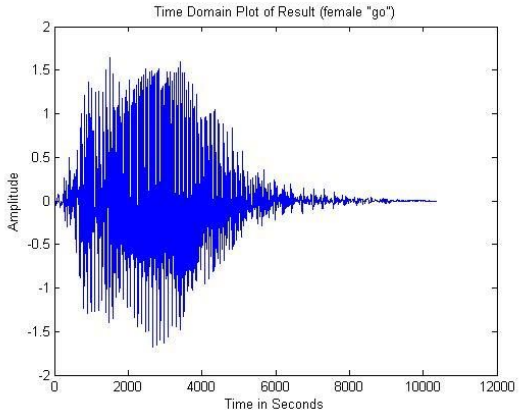
--word *"by"* in male voice
--word *"go"* in female voice

certain sounds.

Word *"by"* in male voice



Word *"go"* in female voice



The simulation plots clearly illustrate that the information of speech signal(i.e. the chunk of words) is present in the shape of its magnitude spectrum whereas the voice is judged by it's frequency(how densely packed the samples are).

From application point of view,there are numerous fields where voice morphing can be extremely helpful.
It is widely used for security and privacy objectives, where the motive is to conceal the identity of a person. A text-to-speech system which is integrated with voice morphing is capable of producing speech of more than one type. Apart from it's use in recreational purposes for voice dubbing in videos,it is widely applied in voice therapy.

References:

[1] *plaza.ufl.edu/guru1984/files/LPC%20based%20Voice%20Morphing.doc*
[2] *http://www.ijecce.org/administrator/components/com_jresearch/files/publications/IJECCE_2520_Final.pdf*
[3] *www.ijsps.com/uploadfile/2014/1210/20141210044237752.pdf*
[4] *https://en.wikipedia.org/wiki/Linear_predictive_coding*