Journal of Basic and Applied Engineering Research
p-ISSN: 2350-0077; e-ISSN: 2350-0255; Volume 4, Issue 6; July-September, 2017, pp. 452-455

http://www.krishisanskriti.org/Publication.html

Manipulation of Voice Properties using PSOLA

Faiyaz Ahmad

Department of Computer Engineering, Faculty of Engineering and Technology,
Jamia Millia Islamia, New Delhi-110025, India
E-mail: ahmad.faiyaz@gmail.com

Abstract—*PSOLA (Pitch Synchronous Overlap and Add) is a digital signal processing technique used for speech processing and more specifically speech synthesis. In this paper we are exploiting the feature of Pitch Synchronous Overlap and Add to change voice properties. Here, we are taking three properties, namely vocal tract, pitch and time. On changing these properties original voice is changed. In PSOLA, we divide the speech waveform in small overlapping segments. Now, if we want to increase the pitch, we need to move speech segments apart otherwise we need to bring the segments together to decrease the pitch.*

Keywords*: PSOLA, pitch, formant frequencies, voice*

Introduction

The vocal cords periodically vibrate to generate glottal flow. Human pronounces a vowel or a voiced consonant which is composed of glottal pulses. Pitch period is the period of a glottal pulse. Fundamental frequency is reciprocal of the pitch period. The vocal tract acts as a time-varying filter to the glottal flow. The characteristics of the vocal tract include the frequency response, which depends on the position of organs, such as the pharynx and tongue. The peak frequencies in the frequency response of the vocal tract are formants, also known as formant frequencies. A speech signal is a convolution of a time-varying stimulus (Glottal Flow) and a time-varying filter (Vocal Tract).[1]

As far as PSOLA is concerned, it has many types, namely, time domain PSOLA, linear prediction PSOLA etc. Time domain PSOLA is the most popular algorithm. In time domain algorithm we basically duplicate or eliminate various frames. First of all, here we try to find out epochs in our speech signals. This epochs are at the instant of glottal closure for each period. But, one should keep in mind that all of these epochs should lie in the same relative position for every frame. Now, with the help of Hanning windows, the signal is separated into frames. These windowed frames can be combined and can be added to achieve overlap and add. When this is done, we get a signal, which is perceptually indistinguishable from the original. Actually, the waveform is not exactly the same, since the sinusoid multiplication carried out is not exactly reversed, but overlap-add helps here to hide any noticeable difference.

RELATED WORK

PSOLA ("Pitch-Synchronous Overlap and Add") is a method have been used to manipulate the pitch of a speech signal to match it to that of the target speaker. So, as far as related works are concerned PSOLA have been used as a voice conversion tool between a source and target speaker. Using PSOLA, the feature of target voice has been extracted and then imposed on target voices to bring the required changes. Our voice is mainly excitation plus message. So, message remains the same for each and every person, only excitation changes. So, the main task here is to get those excitation.

So, in the previous works, the excitation part is achieved using linear predictive coding, also known as LPC. Here our message signal is represented as a function of previous samples.

On the other hand, here we want to achieve something else. Here, instead of taking properties from a target speech, we want to get these feature manually. We get the three properties namely, vocal tract, pitch and time. Now, we will provide a slider, with the help of which one can change these features in the range of half to double. So, if the original speech is of 5 seconds, you can vary it from 2.5 seconds to 10 seconds. In the same way, you can vary other properties too.

ALGORITHM

The method used by us is TD-PSOLA. we shortly introduce the TD-PSOLA algorithm. Let

$v[n]$ = speech utterance
$w[n]$ = window
L = length
$T0$ = local pitch period

$$vi[n] = v[n] \cdot w[n - iT0] \qquad (1)$$

If $L = 2T0$, then spectrum preserves the spectrum envelope of the original signal.

We use Hamming windows and Triangular windows, most of the times.
TD-PSOLA can be used either for synthesis or for prosody modification.

Now, if

$u[n]$ = synthesized speech signal, then it can be obtained by overlap-adding the previously excised frames using the least-squares overlap-add synthesis scheme [3]:

$$u[n] = \frac{\sum_i \lambda_i v_i[n - i(T - T_0)]}{\sum_i w[n - iT_0]} \qquad (2)$$

Here,

T = desired pitch period

λi = a factor compensating for the dependence of the result on the pitch factor

If we ignore the denominator, it can be shown that no significant quality degradation appears. We can also ignore λi, we need to be sure that the data must be represented on a low number of bits. So, by ignoring the denominator the equation (2) can be rewritten as:

$$u[n] = \sum_{i=-\infty}^{\infty} v_i[n - i(T - T_0)] \qquad (3)$$

Now, here T – T0 gives the change is pitch. T here represents the new pitch period. Now, we can increase time scaling by repeating frames we have or compress them by discarding a few of them. While discarding them, one need to keep in mind that we are not discarding those pitches which affect our voice in greater extent.

To generate high quality synthetic speech, TD-PSOLA requires a large recorded speech database.

METHODS USED

Our implementation focus on following properties:

a. Variable Speed Replay

This method of pitch shifting is very straightforward and works by playing back the original sound at an increased or decreased rate, thus creating a shift in pitch. For example

x(n) , replay = x(n),in * c

Where c < 1 is time expansion and c > 1 is time compression.

b. Delay-Line Modulation

This method was described in several publications and can be implemented in several ways. The first principle of the proposed methods was to implement a pitch shift using two saw tooth waves to control the time varying delay line which were set half a period apart. The resulting output waveforms were multiplied by a cross fade filter and divided in to blocks. When the blocks were read faster or slower the pitch would go up or down accordingly. The downside is a fair amount of distortion in the signal and the output signal becomes more noise prone. Alternatively an overlap and add scheme that does not require estimation of the fundamental frequency can be employed using three in phase time varying delay lines. Each line is used on a block that overlaps 2/3 of the next full block length. The end result gives the same desired effect.[4]

c. SOLA Time Stretch and Resample

Basically this method takes the original signal uses the below SOLA algorithm and does a linear resample to get an output signal of the same time duration but with a shifted pitch. Re-sampling is done at the rate of alpha*fs, where alpha is the time stretch or constant.

IMPLEMENTATION

Background

The goal of pitch shifting is to modify up or down the pitch of an audio signal without losing its information, which is preserved in the frequency information and the harmonic ratios. If done correctly the new audio signal will be of the same length, sound like the original signal, but at a desired pitch.

Pitch Detection/Marking

Detection and marking of pitches for the input sound are crucial to the next two algorithms. For input signals of constant pitch the desired pitch marks can be found at the time index location where the signal reaches its maximum amplitude. However for more complicated signals involving multiple instruments and vocals this becomes a much more involved task. The main problem to solve requires then lends itself to finding a way to separate the different pitch periods of the in order to accurately determine the pitch marks for each segment.

Pitch Synchronous Overlap Add (PSOLA)

So, after pitch detection, we proceed towards the implementation of pitch synchronous overlap and add. Here, we will divide the message signals in various chunks. After division, we will Voice and speech processing fall in to the category of applications that this particular algorithm excels at. Based on the assumption that the input can be characterized by a series of pitches, PSOLA remains a two-step process. First the input sound is segmented in to its harmonic, non-harmonic and transient parts then characterized by pitches, known as “analysis”. The second part is known as “synthesis” whereby various transformations can be then applied to the

signal by a parameter set. [5] These two phases are done as follows, with illustrations below for clarification:
I. Analysis:
a. Determine the pitch period. Divide the signal in to small blocks where the pitch is considered constant. Finally do pitch detection on each block in succession.
b. Use a Hanning window centred on the pitch mark to extract each block length of
two individual pitch periods. Thus providing for a smooth transition between blocks using a fade-in/fade-out effect between blocks [6].

II. Synthesis:

a. Choose the analysis segment identified by its corresponding time marking.

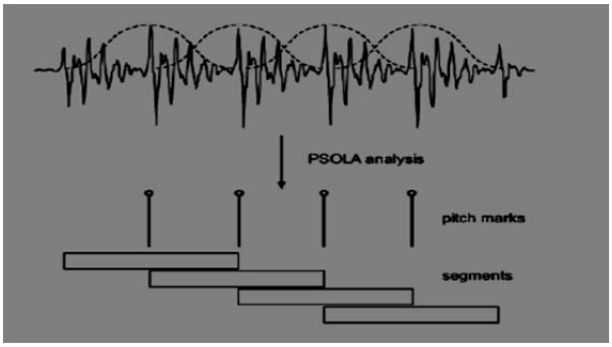

Figure 1: Pitch marking in PSOLA

b. Use the Overlap and Add algorithm where the scaling factor (alpha) decides if the time signal is to be expanded or compressed. Scaling factors less than 1 will result in discarding of segments resulting in time compression. While a scaling factor greater than 1 will cause segments to be repeated resulting in time expansion.

c. Finally the new time index is found in order to centre the next synthesis segment and preserve the pitch.

The end effect of this process is a shift in pitch. This is accomplished using a linear interpolation on the time stretched signal to recreate samples between the samples and then re-sampling to get the desired pitch. This approach is used rather than a simple re-sampling as seen in the SOLA algorithm and should offer much improved sound quality over the previously discussed methods.

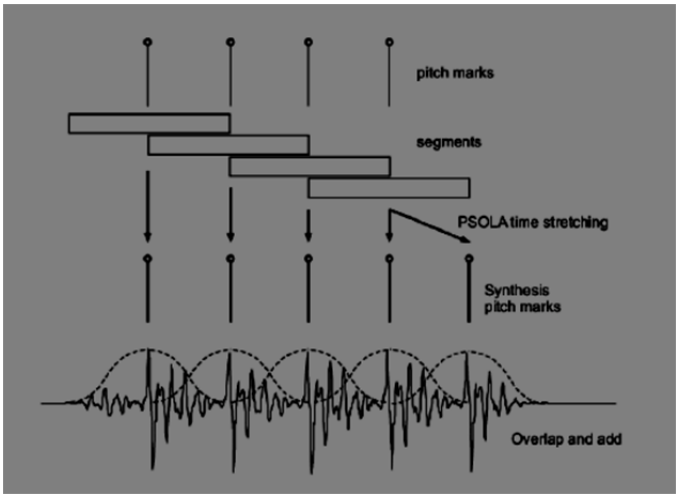

Figure 2: Time stretching in PSOLA

RESULT

Implementation of above method has been done in MATLAB. This GUI actually takes as input recorded source and three sliders input and carry out the conversion as suggested. A screenshot of the implementation is given below:
We have three sliders:
Vocal Tract Modification Slider: This slider enables the user to set the amount of change in formant frequencies.
Pitch Scale Modification Slider: This slider enables the user to set the amount of change in pitch scale of current speech.
Time Scale Modification Slider: Similar to the above slider, the value set here alters the amount of time taken by speaker.

Apart from sliders, we have axes to draw the amlitude of the source and altered voice. Also, there is a reset button, which resets everything.

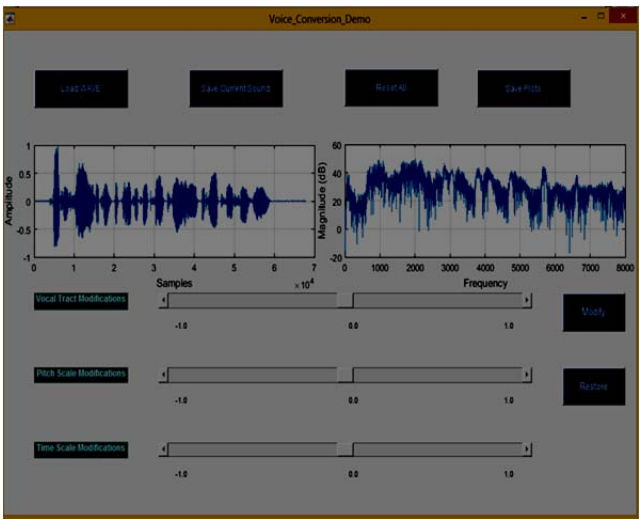

Figure 3: Final Implementation

VII. CONCLUSION

This paper was an extension of the work done in [7] research paper. Here instead of voice morphing, we achieved changing in voice properties. It is evident from the produced sound files that the project is successful in realizing a system that can modify pitch and maintain the integrity of the original sound signal and source. We have changed only three properties of voice and by doing so, we have achieved a considerable amount of change in voice properties.

REFERENCES

[1] Vivek Vijay Nar, Alice N. Cheeran, Souvik Banerjee / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 3, Issue 3, May-Jun 2013, pp.461-465 [2]. http://dspbook.narod.ru/Pitch_shifting.pdf
[3] C. Hamon, E. Moulines and F. Charpentier, "A diphone synthesis system based on time-domain prosodic modifications of speech", in
Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'89, 1989, pp. 238-241 & overall performance evaluation of voice
[4] K. Bogdanowicz and R. Blecher. Using Multiple Processors for real-time audio effects.In AES 7th International Conference, pp. 336- 342, 1989.
[5] C. Hamon, E. Moulines and F. Charpentier. A diphone synthesis system based on time-domain prosodic odifications of speech. In Proc. ICASSP, pp.238-244, 1989.
[6] R. Bristow-Johnson. A detailed analysis of a time-domain format-corrected pitch shifting algorithm. J. Audio Eng. Soc., 43(5):340-353, 1995.
[7] Implementation morphing based on psola algorithm, International Journal of Advanced Engineering Technology , E-ISSN 0976-3945