



GLOBAL JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY: G INTERDISCIPLINARY

Volume 16 Issue 1 Version 1.0 Year 2016

Type: Double Blind Peer Reviewed International Research Journal

Publisher: Global Journals Inc. (USA)

Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Time Stretching using Phase Vocoder

By Vignesh Ganesan

Introduction- Time Scale Modification (TSM) is a principle that is used to modify the time axis or time resolution. This concept can be used in Audio processing as well as Image processing. In this research work, TSM concept is used to perform Time Stretching of an audio signal. In this concept, the speed of the audio playback is changed without affecting audio signal's pitch. The time stretching concept has been used extensively in different application, such as in post production for effects, during music mixing. This is also used in the application to capture or play back a slow motion video. If we look at the current scenario, Time Stretching is extensively used in the field of entertainment, and creative arts.

My paper talks about using this Time Stretching in the Automobile Industry. Now a days we have different types of FM stations (Digital and Analog). There is one particular scenario which I have considered as the Problem statement.

GJCST-G Classification : I.4.1, I.4.0



Strictly as per the compliance and regulations of:



Time Stretching using Phase Vocoder

Vignesh Ganesan

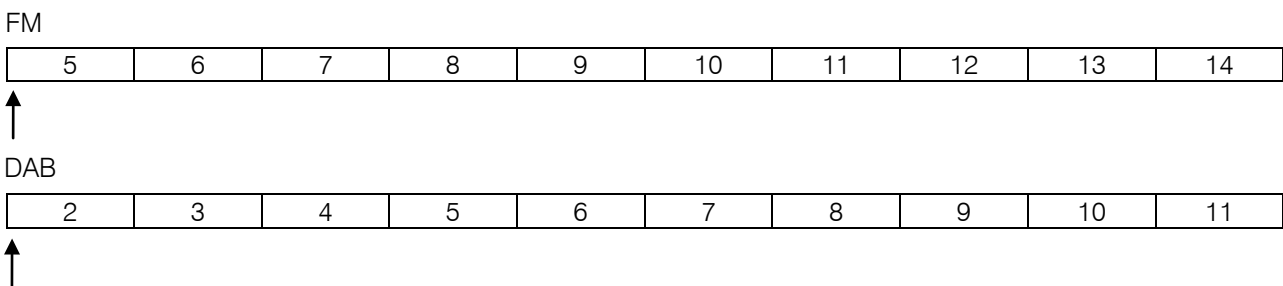
I. INTRODUCTION

Time Scale Modification (TSM) is a principle that is used to modify the time axis or time resolution. This concept can be used in Audio processing as well as Image processing. In this research work, TSM concept is used to perform Time Stretching of an audio signal. In this concept, the speed of the audio playback is changed without affecting audio signal's pitch. The time stretching concept has been used extensively in different application, such as in post production for effects, during music mixing. This is also used in the application to capture or play back a slow motion video. If we look at the current scenario, Time Stretching is extensively used in the field of entertainment, and creative arts.

My paper talks about using this Time Stretching in the Automobile Industry. Now a days we have different types of FM stations (Digital and Analog). There is one particular scenario which I have considered as the Problem statement.

II. USE CASE SCENARIO : FM-DAB LINKING

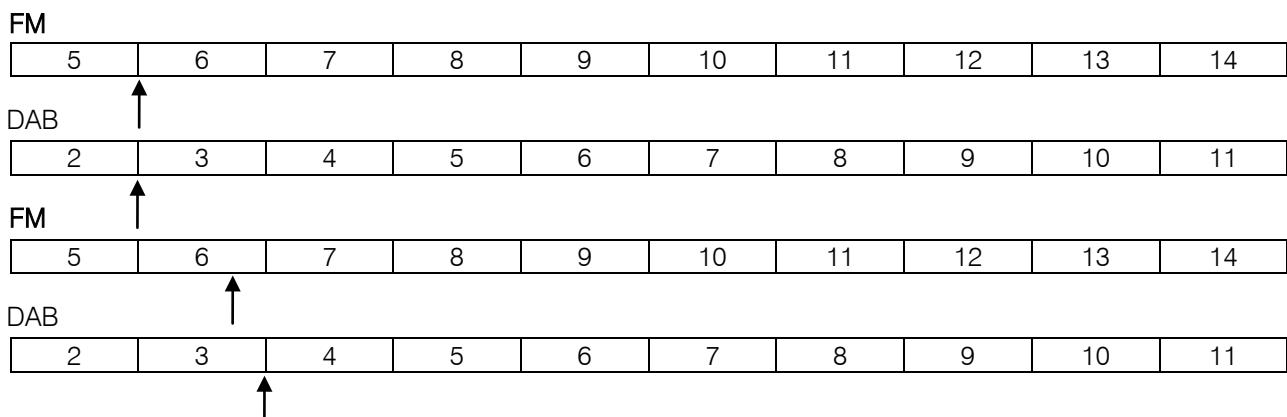
In the concept of Seamless Linking, the FM (Frequency Modulation) would be playing ahead of DAB (Digital Audio Broadcasting). In this case, when switching is necessary from FM to DAB, we would need to wait for an amount 3 seconds (assuming that the delay between FM and DAB is 3 seconds).



Assuming a delay of 3 seconds, In the above example, FM is playing the 5th second of an audio, at the same instant DAB would be playing the 2nd second of the same audio. Now at this point if we need to switch from FM to DAB, we would again hear, 2,3 and 4th second audio before hearing the 5th second audio, So 3 seconds of audio will repeat. In alternate case, if we do not want to hear the same audio again, there would be 3 seconds of audio mute and playback would resume from 5th second in DAB.

In seamless linking this is avoided and a concept called Time Stretching is introduced. In the same example, without switching it immediately, the FM signal from 6th second till 11th second (total duration 6seconds) is stretched to 9 seconds.

The following buffer diagram depicts different stages of stretching and its corresponding pointer location of the DAB signal.



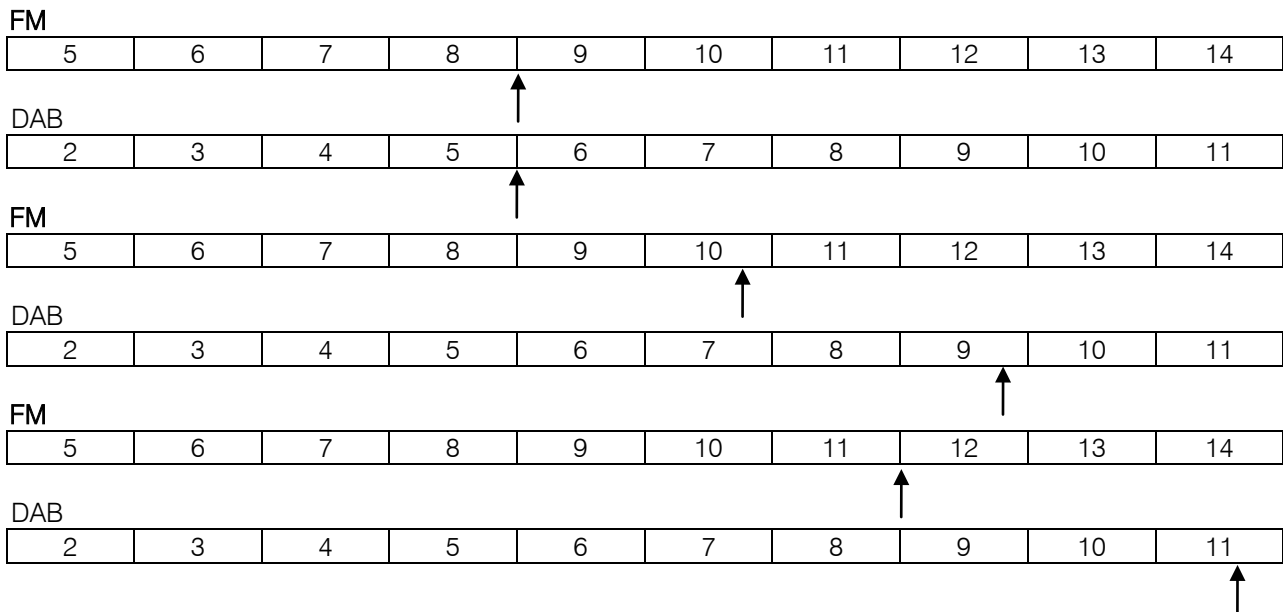


Figure 1 : Use Case

CASE

At the end of time stretching (last diagram) we can see that the pointer of FM and DAB are pointing to

the same time slot. Now if we switch from FM to DAB, there would not be any difference in the content being played.

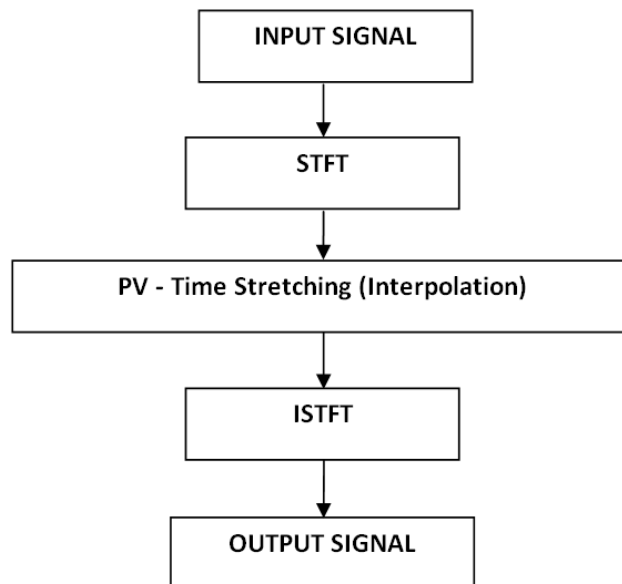
III. ALGORITHM OVERVIEW

Figure 2 : Software Architecture

IV. ARCHITECTURE DIAGRAM

The Fig 2 represents the overall block diagram of the entire time stretching process. The algorithm is developed in such a way that the quality of the output audio is good and at the same time the computational complexity is minimum, considering the fact that this algorithm needs to be used in a real time embedded system for an automobile.

V. ALGORITHM IMPLEMENTATION

Phase Vocoder is the basic principle behind Time Stretching. This concept involves conversion of time domain into frequency domain. The implementation involves totally 3 stages.

Stage 1: STFT

Input to this stage is a pure Time Domain signal, it can be any format, mono, stereo or surround

channel etc. The algorithm has been designed in such a way to accommodate different number of channel. For every channel the same process would be applied. Being the 1st stage, it is responsible for converting the time domain signal to a frequency domain signal. The concept that has been applied here is Short Time Fourier Transform. In general cases a Fourier Transform would be sufficient to convert the time domain signal to a frequency domain signal. This would only result in a single conversion and we would not have much details about what frequency is present at what instant of time. To get more precise details about which frequency is present at which time slot STFT is carried out. This is done taking FFT to the signal at different time slots. The output of this stage would be a Multi Dimensional array with each column representing frequency component at a particular time slot.

Stage 2: Phase Vocoder

The output of the STFT stage will act as the input to this stage. This module performs the core of time stretching. Phase Vocoder stretches time in frequency domain. The input data to this stage is in complex form and is in frequency domain. Interpolation is the backbone concept of this Phase Vocoder. The

algorithm estimates a phasor, which is used to modify the existing sample at a different time position. In this algorithm, I have used interpolation using two columns. The output of this stage will be in frequency domain, it would be a multi dimensional array (2D Matrix) having same number of rows as the input but number of columns would vary according to the stretching desired.

Stage 3: ISTFT

This stage performs Inverse STFT. Input to this module is the output of the Phase Vocoder. Here IFFT has to be performed several times to get back the stretched time domain signal. The only major difference in applying the IFFT is that the output of every IFFT has to follow overlap addition process. This overlap method will avoid the phase discontinuity and amplitude reduction at the frame border.

VI. SIMULATION RESULTS

Implementation of this algorithm has been successfully evaluated in MATLAB Platform. Both graphical evaluation and psychoacoustic listening tests were also conducted. Some of the simulation results are described below.

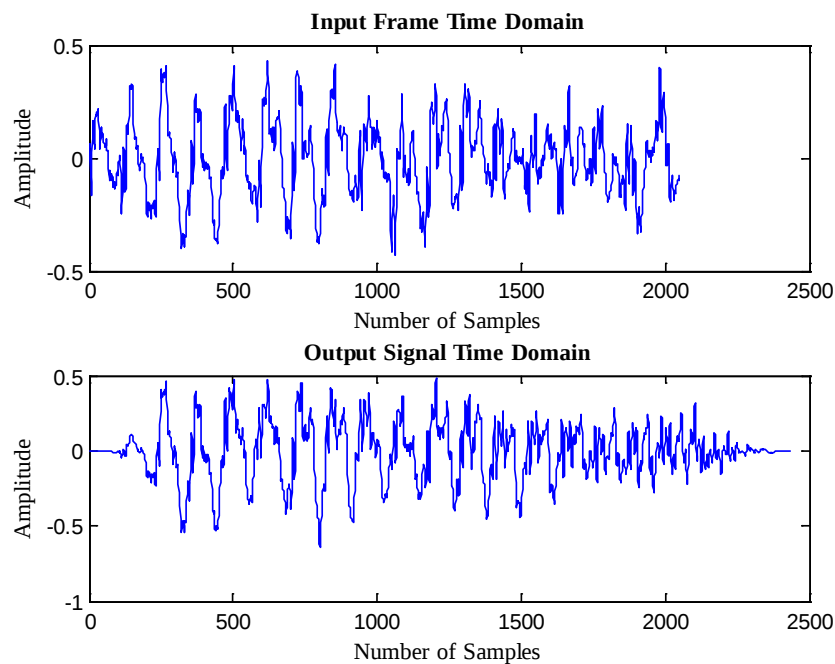


Figure 3 : Audio Time Stretching (Frame by Frame): Input Frame vs Stretched Output Frame

Fig.3. represents the analysis of just one frame iteration. According to the above iteration, the input is of 2048 samples and from the output graph it is evident that it has been stretched close to 2450 samples. This output will not be the final data written to the file. There will be overlap addition taking place, so that the end segment will be compensated and there would be no data loss.

This graphical representation is just for a basic understanding of how this algorithm works frame by frame. The next representation would show the overall input vs output audio signal with respect to the entire song duration of 6 seconds.

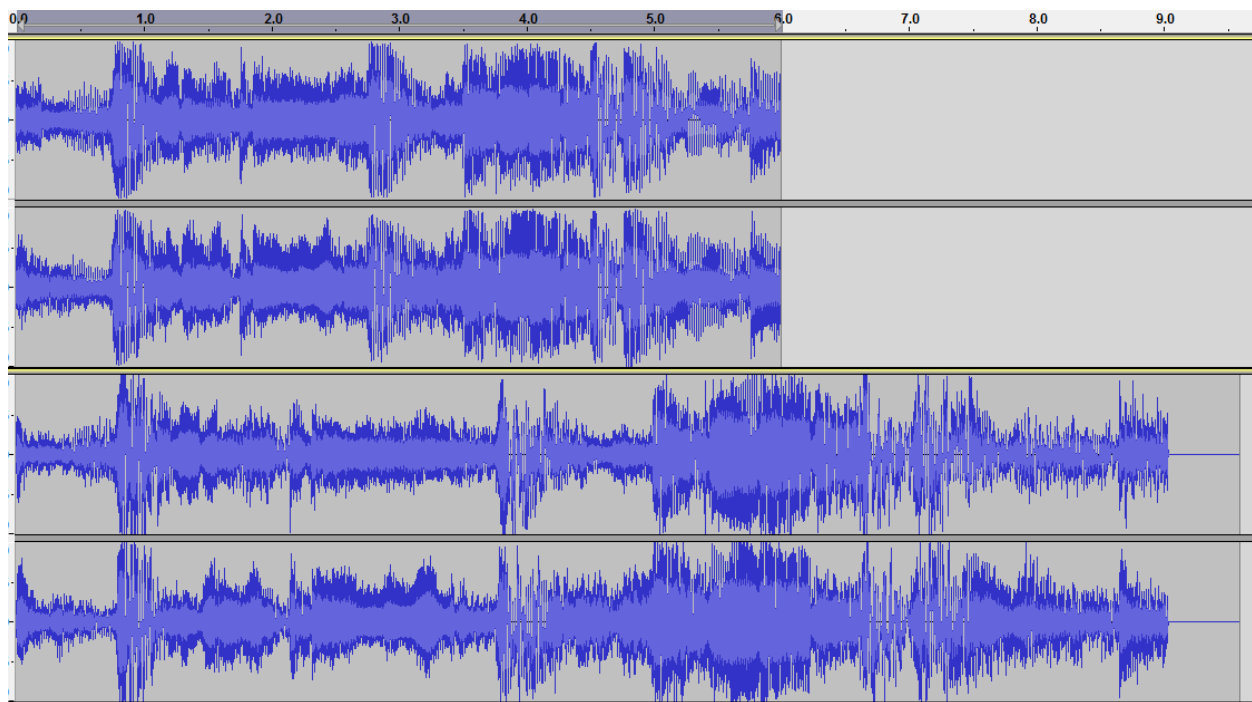


Figure 4 : Audio Time Stretching: Original Input (Top) vs Stretched Output (Bottom)

From Fig.4. one can observe that the audio has been stretched from 6 seconds to 9 seconds. The top audio graph is that of the original input and the below one represents the stretched output. Both the audio signals are in time domain. Almost 10 different types of audio was tested and the quality was verified.

VII. CONCLUSION AND FUTURE SCOPE

Though this algorithm was mainly created and developed to work for real time scenario, at present this algorithm has been successfully tested and verified under MATLAB platform. Once this algorithm is converted to Embedded C domain, then it would work in real time scenario. This algorithm needs to be linked to a seamless linking algorithm for maximum efficiency. To achieve that just certain parameters needs to be tweaked and nothing major needs to be done at the root level of the algorithm.

In general cases of Time Stretching, the audio is processed for the entire selection or a file, but in this concept it works on a frame by frame basis which is what makes it unique to work in a real time scenario such as an Automobile.

The phase discontinuity is suppressed almost to an unnoticeable change, and the phasiness effect has also been reduced. This algorithm has been designed for an optimal solution which is a good enough quality (the effect of time stretching is not too disturbing for the human ear), and a medium sized computational complexity.