Audio Signal Processing - Time-scaling phase vocoder

LACOMBE Yoach

MVA | ENS Paris-Saclay Paris

Abstract

Time-scaling vocoder are used for speeding up or slowing down audio signals. The phase vocoder (PV) is a specific audio signal modification paradigm that uses short-time Fourier transform (STFT) to modify sounds. PV are applied in a number of algorithm. This paper implements some of these algorithms and comments their pros and cons.

1. Introduction

Time-scale algorithm aims at adjusting the length of an audio signal while preserving its pitch and timbre. Ideally, it should sound as if the signal was performed at a different tempo. While being principally used for music production, sampling and remixing, time-scale modifications (TSM) are also useful for video production and streaming.

Given a contraction factor α, the most naive approach one could use would be to simply approximate the function $x \longmapsto f(\alpha x)$ where f is the sound. However, a contraction in the time space leads to a dilatation in the frequency space, which naturally changes the pitch of the sound.

While the classic TSM algorithms give relatively good results, they also introduce artifacts such as phasiness, reverberation and loss of presence, especially for polyphonic and rich sound. This paper aims at reproducing some classic phase-vocoder TSM algorithms and at commenting their efficiency qualitatively.

2. Literature review

TSM [2] can be divided into two main paradigms. [2] gives a good overview of the two paradigms and their main challenges.

Time-domain-based vocoders modify the audio on the time domain. The main idea behind these vocoders is Overlapp and Add (OLA), which means that it cuts the original audio into frames (they may be overlapping) and recombines them in such a way that the output signal is time-stretched by α. Basic OLA algorithm suffers from a number of downsides such as not being able to preserve local periodic structures, especially from periodic signals. A possible amelioration comes from Waveform Similarity OLA (WSOLA) [7] and its variations which reduce phase jump artifacts by allowing tolerance in the frames' positions. These ameliorations still suffer from some artifacts such as transient or doubling smearing (loss or duplication of percussiveness) and tempo modulation.

Frequency-domain-based vocoders modify the audio based on short-time Fourier Transform (STFT). Here, overlapping of frames is synchronized by modification of the STFT's phases. Transient smearing and phasiness (in which speakers appear further away from the mic) are typical artifacts of phase-vocoder. The most classic phase-vocoder (PV) is called PV-TSM and maintains horizontal (time coherence) at the cost of vertical coherence. [4] and [6] correct this lack of vertical coherence either by phase locking or by phase gradient estimation.

Some extensions based on both approaches also exist. [1] is an approach that apply PV-TSM to harmonic components and OLA to percussive components, taking the best of both word.

The rest of the paper focuses on explaining and applying PV-TSM based algorithms.

3. PV-TSM algorithms

3.1. PV-TSM basics

The typical TSM pipeline is made of three steps, that is synthesis, modification and resynthesis.

The **analysis** stage computes the STFT of the original signal x.

$$X(m,k) = \sum_{n=0}^{N-1} h(n)x(n+mR_a)e^{-2i\pi kn/N}$$

where $k \in \mathbb{N}_N$ is the frequency bin index, m is the frequency time index, R_a is the analysis hop, h the STFT window (typically the hanning window) and N is the window length.

The **modification** stage computes Y, the new STFT, from X and the dilation factor α.

The **resynthesis** stage typically uses the inverse STFT. Short-time signals $y_m(n)$ are obtained by computing the inverse FFT of Y . These signals are then weighted by

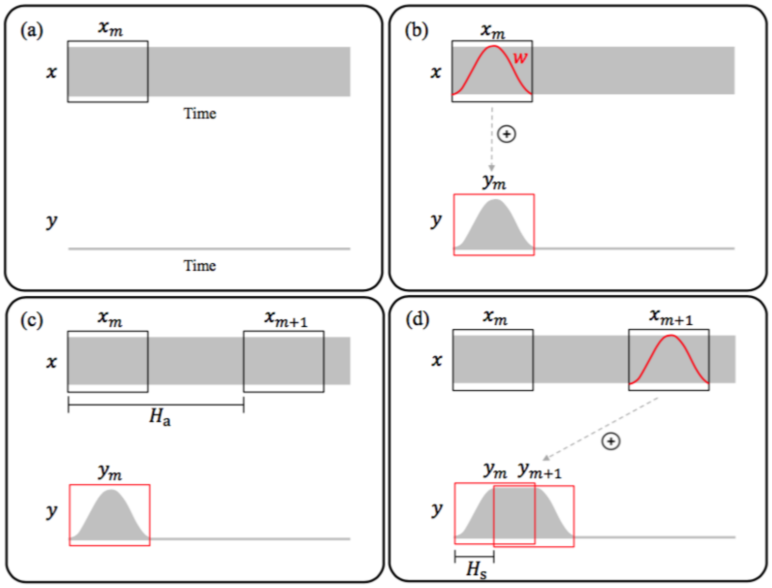

Figure 3. The principle of TSM based on overlap-add (OLA). **(a)** Input audio signal x with analysis frame x_m. The output signal y is constructed iteratively; **(b)** Application of Hann window function w to the analysis frame x_m resulting in the synthesis frame y_m; **(c)** The next analysis frame x_{m+1} having a specified distance of H_a samples from x_m; **(d)** Overlap-add using the specified synthesis hopsize H_s.

Figure 1. OLA principle explained (from [2]).

a synthesis window h (typically the hanning window) and overlapp-add by a synthesis hop R_s to compute the output signal y.

$$y_m(n) = \frac{1}{N} \sum_{k=0}^{N-1} Y(m,k) e^{2i\pi kn/N}$$

$$y(n) = \sum_{m=-\infty}^{\infty} h(n - mR_s) y_m(n - mR_s)$$

3.2. PV-TSM modification stage

The modification stage of the original PV-TSM implementation consists in adjusting the output phase at time bin m according to the previous phase at time $m-1$ and the adjusted instantaneous frequency of the original signal at m.

In practice, we set Y such that:

$$|Y| = |X| \quad (1)$$

$$\Phi_Y[0,:] = \Phi_X[0,:] \quad (2)$$

$$\Phi_Y[m,:] = \Phi_Y[m-1,:] + R_s IF(m) \quad (3)$$

where Φ_X is the phase of X and $IF(m)$ the instantaneous frequency at time bin m, i.e:

$$IF(m) = \Omega + \frac{1}{R_a}[\Phi_X[m,:] - \Phi_X[m-1,:] - R_a\Omega]_{2\pi} \quad (4)$$

with $\Omega = \{k\frac{2\pi}{N}\}_{k\in\mathbb{N}_N}$ and $[]_{2\pi}$ denoting the conversion to the range $\pm\pi$.

1 sets the spectral amplitudes, which are not modified and 2 initializes the phase.

Note that $[\Phi_X[m,k] - \Phi_X[m-1,k] - R_a\Omega[k]]_{2\pi}$ is the difference between the mesured frequency at time-frequency bin [m,k] and the predicted frequency when assuming a frequency of $\Omega[k]$.

Remark: [4] proposes to replace 2 with $\Phi_Y[0,:] = \alpha\Phi_X[0,:]$ where α is the dilatation ratio. It is supposed to help with the phasiness issue of the PV-TSM.

3.3. How to improve PV-TSM

In the algorithm above, each frequency bin is processed independently. However, [4] observes that a sinusoidal component may affect multiple adjacent frequency bins of a single analysis frame. In order to be improved, the new algorithms thus must incorporate vertical (frequency) phase coherence.

3.3.1 Phase-locking

[4] assumes that a peak in a frame's magnitude is representative of a particular sinusoidal component and that the surrounding bins with lower magnitude are affected by this very same sinusoidal component. Thus, it proposes to lock

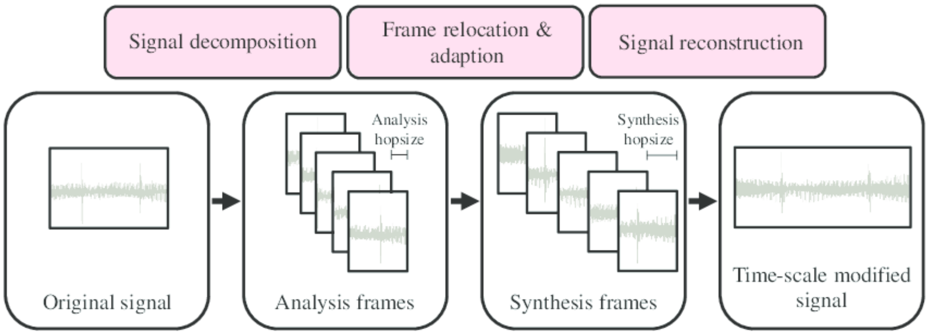

Figure 2. TSM process (from [2]).

the surrounding bins to the phase of the given peak in order to conserve the local vertical phase coherence.

In practice, it proposes several phase-locking procedure, the main ones being *loose phase-locking* and *identity phase-locking*.

Loose phase-locking means computing a vertical rolling sum on the resulting STFT i.e $Y_{new}[:, k] = Y[:, k-1] + Y[:, k] + Y[:, k+1]$

Identity phase-locking consists in the following steps, for each analysis frame.

1. Identify the peaks which are identified as bins where the magnitude is larger than the 4 nearest neighbors.

2. Compute the synthesis frame phase of each peak according to 3.

3. Identify the closest channels to each peak.

4. For a channel k and its closest channel k_l, update the synthesis phase such that:

$$\Phi_Y[m, k] = \Phi_Y[m, k_l] + \Phi_X[m, k] - \Phi_X[m, k_l]$$

[4] also proposes a scaled phase-locking consisting in modifying step 2 such that the peaks' phases are updated according to the corresponding previous peaks of the same bins.

3.3.2 Other approaches

There are other approaches existing to improve vertical phase coherence that I haven't had the time to implement. To name but a few, PVSOLA [3] consists in resetting the output frames to the input frames every D processed frames. [6] dynamically change the phase either horizontally or vertically according to partial derivative. [5] automatically adapt the time-frequency resolution to analyse and resynthesize.

4. A harmonic-percussive approach

[2] also proposes a TSM 3 based on harmonic-percussive separation. The principle is simple. First, it separates the STFT into two STFT components (harmonic-percussive separation - HPS), one focusing on harmonic sounds and one focusing on percussive sounds. To the harmonic component, it applies PV-TSM as it works well with harmonic sounds, whereas it applies OLA to the percussive component as it works well with percussive and transient sounds. At the end of the day, the output signal is the superposition of the processed harmonic and percussive components.

The HPS is performed by first applying a vertical median filter and a horizontal median filter to the spectrogram. When the vertical spectrogram is larger than the horizontal spectrogram, the corresponding time-frequency bins are associated the percussive component. If not, they are associated to the harmonic component.

5. Evaluation

[4] proposes a consistency measure which basically measures the distance between the new STFT Y and the STFT of the reconstructed signal $y = ISTFT(Y)$. In other words, it compares Y and $STFT[ISTFT(Y)]$ where $ISTFT$ is the "inverse" of the STFT in the least square senses. However, note that no other articles I read used this consistency measure, and that no clear explanations are given as to why it would be a good measure.

The only good way of robustly measuring the TSM algorithms' efficiency would be to conduct a formal listening test such as MUSHRA, with several auditors and a robust evaluation.

However, here a some informal observations. I listened to several types of sound: speech sounds, singing voices, monophonic music and polyphonic music.

As expected in [4], phase-locked PV always corrects some of the artifacts of the classic PV, especially phasiness. However, in some cases, I've heard some distinct interruptions. Some of the attacks also lost their percussiveness.

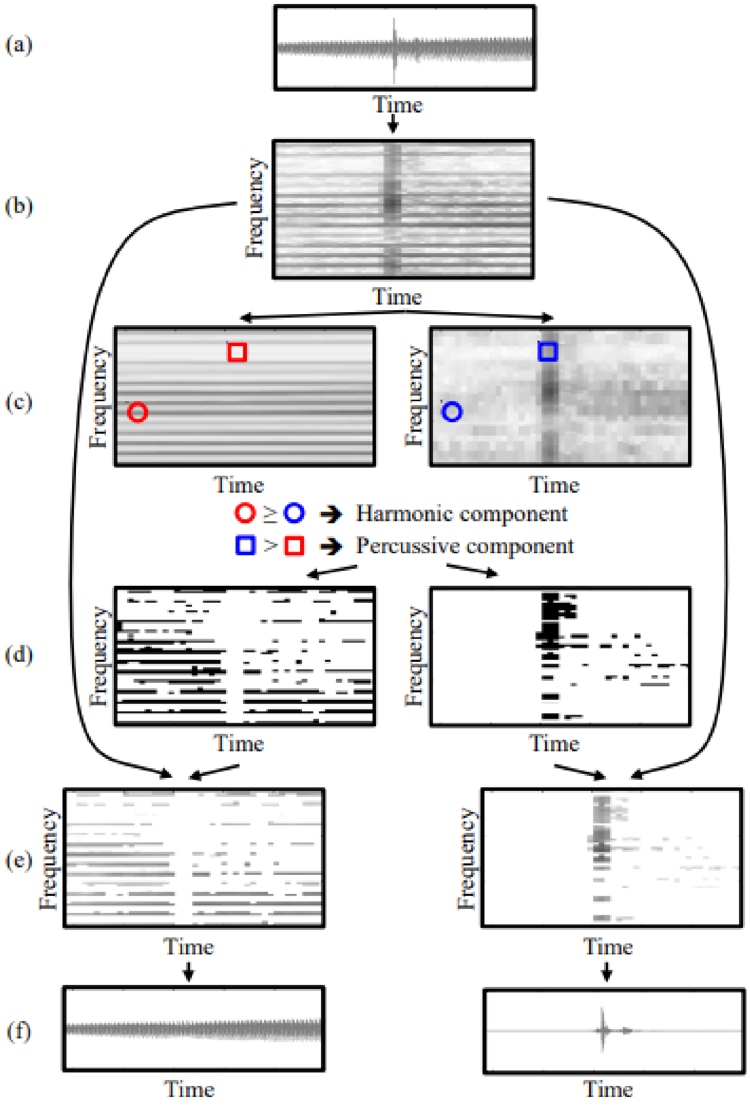

Figure 3. HPS approach's pipeline (from [2]).

The identity phase-locked PV always sounds better though.

HPS-based TSM corrects some of the interruptions and (most of the times) conserves the percussiveness of the attacks. The frequency and time lengths of the median filters highly influence the quality of the results. I found that a frequency median filter length of 25 and time median filter length of 100 give good results in all the audio samples I tested.

On the modification of equation 2: I found that applying the remark of section 3.2 never improves the quality of the output signal while most of the times highly decreasing the said quality. Thus, it doesn't seem to be a solution to the resolve phasiness issue.

On the FFT size and analysis/synthesis hop sizes: It is well know that the FFT size N is an important factor influencing the quality and precision of the STFT. It gives a trade-off between the time-space resolution and the frequency-space resolution. On some of the papers I read, N was set such that it takes frames of size 100 ms, i.e $N = 0.1F_s$. After having played a little bit with this parameters over a few sound samples, I decided to follow this recommendation, as it gave the more consistent results throughout the sample sounds.

As to the analysis and synthesis hop sizes R_a and R_s, most of the papers chose to set $R_s = N/2$ and thus $R_a = R_s/\alpha$. However, with $N = 0.1F_s$ and since most of the frequency rate are quite large, R_s and R_a are most of the times too large to give good results. Intuitively, it might be because when the hop size is too large, the STFT loses signal information, i.e the perfect reconstruction condition is not satisfied anymore, but I haven't dwell on it that much.

This observation might explain why the informal observation of [4] prefers the scale phase-locked PV to the identity phase-locked PV. Personally, after having set the hop sizes to more reasonable values ($R_a = 128$ and $R_s = \alpha R_a$), I've heard little to no difference between those two algorithms, and when I've heard some differences, I always preferred the identity phase-locking. Moreover, all of the TSM algorithms sounded better with those more reasonable values as compared to large hop sizes.

On HPS: Even without applying the TSM, I've found that the separation of harmonic and percussive components by applying median filters gives excellent results. It is a simple, yet effective way to separate both components. However, when adding both components after TSM, I found some unpleasant effects at the beginning of the audio sample due to the initial phase jump between the OLA-processed component and the PV-TSM-processed component. A good amelioration would be to correct that artifact.

6. Conclusion

In this paper, I've implemented some of the most well-known phase-vocoder based TSM. I found that some hyper-parameters are crucial to hear pleasant results. Moreover, the values recommended in the literature appeared to be not suited for the time-stretching task.

A limitation of my work is that I should have tested more the methods on polyphonic signals. Moreover, I haven't tested them yet on polychannel audio signals.

While most of the approaches are based on simple phase-vocoder approaches, my next line of work would be to implement [5] or [6] which both propose methods that goes beyond the usual PV-framework.

References

[1] Jonathan Driedger, Meinard Müller, and Sebastian Ewert. Improving time-scale modification of music signals using harmonic-percussive separation. *IEEE Signal Processing Letters*, 21:105–109, 2014. 1

[2] Jonathan Driedger and Meinard Müller. A review of time-scale modification of music signals. *Applied Sciences*, 6(2), 2016. 1, 2, 3, 4

[3] Sebastian Kraft, Martin Holters, Adrian von dem Knesebeck, and Udo Zölzer. Improved pvsola time-stretching and pitch-shifting for polyphonic audio. 09 2012. 3

[4] J. Laroche and M. Dolson. Improved phase vocoder time-scale modification of audio. *IEEE Transactions on Speech and Audio Processing*, 7(3):323–332, 1999. 1, 2, 3, 5

[5] Marco Liuni, Axel Roebel, Ewa Matusiak, Marco Romito, and Xavier Rodet. Automatic Adaptation of the Time-Frequency Resolution for Sound Analysis and Re-Synthesis. *IEEE Transactions on Audio, Speech and Language Processing*, 21(5):959–970, 2013. cote interne IRCAM: Liuni13a. 3, 5

[6] Zdenek Prusa and Nicki Holighaus. Phase vocoder done right, 2022. 1, 3, 5

[7] W. Verhelst and M. Roelands. An overlap-add technique based on waveform similarity (wsola) for high quality time-scale modification of speech. In *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 554–557 vol.2, 1993. 1