Proc. of the 15th Int. Conference on Digital Audio Effects (DAFx-12), York, UK , September 17-21, 2012

IMPROVED PVSOLA TIME-STRETCHING AND PITCH-SHIFTING FOR POLYPHONIC AUDIO

Sebastian Kraft, Martin Holters, Adrian von dem Knesebeck, Udo Zölzer

Helmut-Schmidt-University
Department of Signal Processing
Holstenhofweg 85, 22043 Hamburg, Germany
sebastian.kraft@hsu-hh.de

ABSTRACT

An advanced phase vocoder technique for high quality audio pitch shifting and time stretching is described. Its main concept is based on the PVSOLA time stretching algorithm which is already known to give good results on monophonic speech. Some enhancements are proposed to add the ability to process polyphonic material at equal quality by distinguishing between sinusoidal and noisy frequency components. Furthermore, the latency is reduced to get closer to a real time implementation. The new algorithm is embedded into a flexible pitch shifting and time stretching framework by adding transient detection and resampling. A subjective listening test is used to evaluate the new algorithm and to verify the improvements.

1. INTRODUCTION

The phase vocoder is one of the oldest and most prominent techniques when it comes to time stretching and pitch shifting in the frequency domain. It was originally introduced in 1966 by Flanagan [1] and has then been realized as a parallel bank of bandpass filters. Portnoff [2] developed an efficient implementation by utilization of the fast fourier transform, which became the usual way to use the phase vocoder for signal processing in these days.

When it comes to the application of pitch shifting and time stretching, the phase vocoder has the advantage to easily handle polyphonic as well as monophonic audio but several drawbacks and annoyances are known from the original implementation. The typically arising artefacts are called phasing or phasiness and the sound can be described as becoming indirect or muffled. It also feels like the sound source moves away from the listener. Finding the cause of these artefacts and avoiding them is a research topic ever since the phase vocoder has been used in audio signal processing. A very effective and commonly used technique is the locked phase vocoder which has been first described by Puckette [3] and was further improved by Laroche and Dolson in [4] together with an in-depth analysis of phase vocoder artefacts.

Recently two new approaches to the reduction of phasing artefacts were published that combine SOLA, a common time domain technique, with a phase vocoder in the frequency domain. The shape invariant phase vocoder (SHIP) by Röbel [5] calculates the cross correlation to determine the phase shift that is needed to coherently overlap succeeding frames in the output. In contrast PVSOLA from Moinet [6] uses the cross correlation to regularly insert unmodified input frames directly into the output stream. This resets the phase and reconstructs perfect vertical phase coherence. Audio examples from Moinet[1] proof that this leads to a superior audio quality on speech signals even at high time stretch factors. Apart from that the results on polyphonic audio and complete music mixes are worse and superimposed by a distracting amplitude modulation.

As an introduction the basics of the phase vocoder and PVSOLA are initially described in section 2 and afterwards in section 3 two steps are presented to improve the PVSOLA algorithm. A first modification deals with a simplified calculation of the cross correlation between input and output frames that automatically compensates the occurring time drift during the resets. The other one reduces the artefacts that arise while processing polyphonic signals by limiting the reset to sinusoidal spectral components. The integration of the new algorithm into a pitch-shifting and time-stretching framework is depicted in section 4 and the overall results are discussed afterwards in section 5.

2. PVSOLA

The process of time stretching with the phase vocoder is based on the analysis of an input signal in overlapped frames and by resynthesizing them with a different overlap factor. For the common phase vocoder the analysis hop size R_a and synthesis hop size R_s are connected by the desired time stretch factor

$$\alpha = \frac{R_s}{R_a}. \tag{1}$$

Obviously this will likely create artefacts caused by discontinuities between the time shifted output frames. It is the task of the phase vocoder to reduce these artefacts and to make sure that succeeding frames overlap coherently.

2.1. The phase vocoder

The input signal is split at fixed analysis time instants $t_a^u = uR_a$ into overlapping frames x_u of length N, where u is a sequential integer index denoting the frame number. After applying the window function $w(n)$ and calculating the DFT we get the successive spectral representations

$$X(t_a^u, k) = \sum_{n=0}^{N-1} x(t_a^u + n)w(n) \cdot e^{-j\Omega_k n}, \qquad \Omega_k = \frac{2\pi k}{N}. \tag{2}$$

For small N and usual audio signals the content of x_u can be supposed to have a quasi-stationary characteristic. By this assumption the phase vocoder only needs to adjust the phase of every bin

[1] http://tcts.fpms.ac.be/~moinet/pvsola/

to properly match the phases at the shifted synthesis time positions $t_s^u = uR_s$.

The phase adjustment is based on the observed phase difference between the current and the previous frame and therefore, the spectrum is divided into amplitude and phase

$$r(t_a^u, k) = |X(t_a^u, k)| \tag{3}$$

$$\Phi(t_a^u, k) = \arg\left(X(t_a^u, k)\right). \tag{4}$$

The *heterodyned phase increment* is calculated between the time instants t_a^u and t_a^{u-1} of the input phases

$$\Delta\Phi(t_a^u, k) = \left[\Phi(t_a^u, k) - \Phi(t_a^{u-1}, k) - R_a\Omega_k\right]_{2\pi} \tag{5}$$

where the operator $[]_{2\pi}$ means to take the principal argument [7] to limit the phase values to a range of $\pm\pi$. Hence,

$$\omega(t_a^u, k) = \Omega_k + \frac{1}{R_a}\Delta\Phi(t_a^u, k) \tag{6}$$

is derived which is an estimation of the instantaneous frequency that refines the bin frequency Ω_k to a more accurate measure. Multiplying $\omega(t_a^u, k)$ by R_s yields the phase increment for the synthesis hop size and finally the output phase

$$\Psi(t_s^u, k) = \Psi(t_s^{u-1}, k) + R_s\omega(t_a^u, k) \tag{7}$$

is obtained by addition to the phase of the previous frame.

The synthesis frames y_u at time instants $t_s^u = uR_s$ are calculated from the synthesis spectra

$$Y(t_s^u, k) = r(t_a^u, k) \cdot e^{-j\Psi(t_s^u, k)} \tag{8}$$

$$y_u(n) = \frac{1}{N}\sum_{k=0}^{N-1} Y(t_s^u, k) \cdot e^{j\Omega_k n} \tag{9}$$

by an inverse fourier transformation. In a last step, the overall output signal

$$y(n) = \sum_{u=0}^{\infty} w(n - t_s^u) \cdot y_u(n - t_s^u) \tag{10}$$

is achieved by windowing and overlap-adding of the single frames.

While the processing described above perfectly reconstructs horizontal phase coherence on time axis, there usually also exists a fixed relation between the center bin of a sinusoid and its neighbouring bins. This vertical coherence in frequency direction is neglected by the standard phase vocoder though it needs to be preserved, as it otherwise leads to the well known phasing artefacts. The problem is partly approached with the phase locking described in [3][4] by setting the area around a sinusoidal peak to the phase of the center bin and inheriting the phase relations from the input frame.

2.2. Reset

Another solution comes from Moinet [6], who notes that the loss of vertical phase coherence rises with the number of processed frames and that phasing is only barely audible in the beginning. Based on that fact he proposes to regularly reset the phase vocoder after a small number of frames and thus reconstruct perfect vertical coherence by insertion of an unmodified frame from the input. The position to insert the reset frame is determined by calculating the cross correlation between input and output. This minimizes discontinuities and phase errors in time direction while maintaining perfect vertical phase coherence.

The processing is done in cycles of D ([6] chooses $D = 4$) frames and before performing a reset at the end of every cycle, the standard phase vocoder processes D-1 frames. This means a reset is triggered every D frames at time instants t_s^c where c is an integer multiple of D. During a reset the position $t_i = t_s^c + \Delta_n$ to insert the reset frame may vary by Δ_n inside a range of $\pm 2R_s$. To accomplish a proper symmetric computation of the cross correlation without the influence of the overlapped window envelope, the output signal $y(n)$ has to be completely processed up to a time position $t_s^F = t_s^c + N + 2R_s$. This means the phase vocoder has to produce q additional temporary frames until $t_s^{c+q} > t_s^F$ in advance of the reset procedure.

The cross correlation

$$Z(m) = \sum_{n=0}^{N-1} x_c(n)w^2(n) \cdot y(t_s^c + n - m) \tag{11}$$

is calculated only for m in the range $[-2R_s \ldots 2R_s]$ as the maximum possible time shift is also limited to $\pm 2R_s$. The index of the maximum value in this range is the desired optimal time shift

$$\Delta_n = \arg\max_m \left(Z(m)\right) \tag{12}$$

to insert the input frame at the output relative to t_s^c.

Before overlap-adding x_c to the output, the usual envelope that normally exists at the insertion point has to be reconstructed. Furthermore, some of the additional samples added during the reset procedure are set to zero. The expected envelope

$$e(n) = w^2(n + 3R_s) + w^2(n + 2R_s) + w^2(n + R_s) \tag{13}$$

is calculated from the squared and overlapped window functions and applied to $y(n)$ at position t_i. After adding x_c the next cycle starts.

It has to be noted that the periodic time shifts by Δ_n accumulate to an overall time shift Δ_T that usually is not equal to zero. As soon as Δ_T exceeds $\pm d \cdot R_s$ the phase vocoder creates $q \mp d$ additional frames before the next reset and the synthesis position t_s^c is shifted by $\mp d \cdot R_s$.

3. PROPOSED IMPROVEMENTS

The regular reset of the synthesis phase yields very good quality on speech signals even at high time stretch factors. Apart from that, the results with polyphonic audio have some considerable artefacts as already mentioned by [6]. Namely, on polyphonic or unvoiced signal parts an amplitude modulation occurs that is regulated by the frequency of the reset. This problem will be approached in section 3.2.

Initially a sliding compensation of the drift Δ_T by biasing the selection of Δ_n together with a modified calculation of the cross correlation in the frequency domain similar to [5] is described. This also avoids the calculation of additional frames during the reset and therefore, the need to have an input buffer holding future samples, which would in turn increase the overall latency.

3.1. Cross correlation and drift compensation

The cross correlation is intended to determine the time difference between the output frames created by the phase vocoder and an unmodified input frame. If the overlap during the synthesis is big enough, it would be sufficient to calculate only the cross correlation between input frame x_c and the corresponding phase vocoder generated output frame y_c. As both frames are already available as a spectral representation in the current step,

$$Z(m) = \sum_{k=0}^{2N-1} \left[X'(t_a^c, k)^* \cdot Y'(t_s^c, k)\right] e^{j\Omega_k m}, \qquad \Omega_k = \frac{2\pi k}{2N} \tag{14}$$

can be efficiently calculated in frequency domain. $X'(t_a^c, k)$ and $Y'(t_s^c, k)$ are the corresponding spectra interpolated by an FIR filter to twice the length N to avoid a cyclic convolution.

The cross correlation of two time shifted periodic signals is also periodic and so it is possible to maximize the correlation at several time instants. By biasing the selection of an area to search for local maxima, the range and direction of the time shift Δ_n can be controlled. To likely compensate the cumulated drift Δ_T, a bias should lead the selection in the opposite direction.

As the spectra in the calculation of $Z(m)$ are windowed in time domain, the cross correlation is superimposed by the auto correlation $Z_w(m)$ of the window function. This would introduce another bias that has to be compensated in advance like it is also done in [5]. The compensation curve $K(m)$ simply is $Z_w(m)$ limited to an area where $Z_w(m) > Z_w(N/3)$ to avoid rounding errors at both ends. The actual compensation is done by dividing $Z(m)$ with $K(m)$:

$$K(m) = \max\left(Z_w(m), Z_w(N/3)\right) \tag{15}$$

$$Z'(m) = \frac{Z(m)}{K(m)} \tag{16}$$

y_c is added to the output to make $y(n)$ complete up to $t_s^c + R_s$ and by now the maximum possible time shift during a reset is $T_h = R_s$ in positive direction, while the maximum negative shift is still $T_l = -2R_s$. Because of the asymmetric time shifts, the mean drift will not converge to zero but to a value $\Delta_{T_0} = T_h - 0.5(T_h - T_l)$ which leads to a zero offset $\Delta_{T_0} = -0.5R_s$. Finally the selection is achieved by multiplying the compensated cross correlation $Z'(m)$ with a sine window $w_x(m)$. The width of the window is $L = T_h - T_l$ and its position is determined by the borders k_l and k_h:

$$\begin{aligned} w_x(m) &= \sin\left(\frac{\pi m}{L}\right), \qquad m = [0 \dots L-1] \\ k_l &= \Delta_{T_0} - L/2 - \Delta_T \\ k_h &= \Delta_{T_0} + L/2 - \Delta_T \\ \hat{Z}(m) &= \begin{cases} Z'(m) \cdot w_x(m - k_l), & k_l < m < k_h \\ 0, & \text{elsewhere} \end{cases} \end{aligned} \tag{17}$$

Figure 1 shows the described movement of the selection window depending on the cumulated drift. The index of the maximum value of $\hat{Z}(m)$ directly yields the time shift

$$\Delta_n = \arg\max_m \left(\hat{Z}(m)\right). \tag{18}$$

When the correct time shift Δ_n is found, the output signal envelope has to be prepared. Because no additional frames have

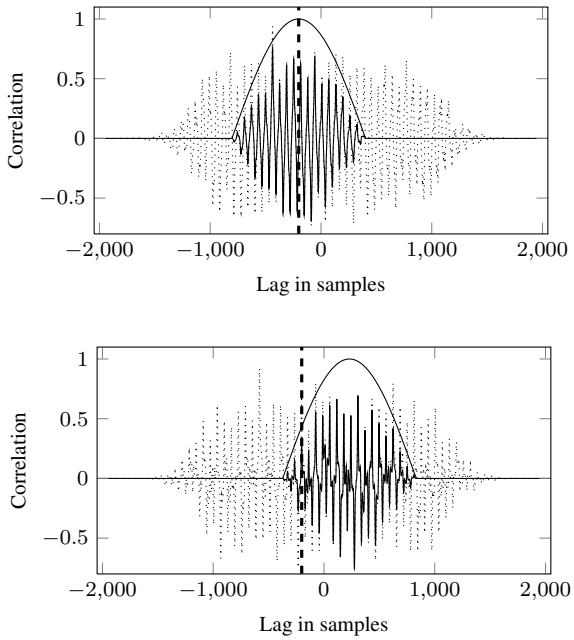

Figure 1: Windowing of $Z'(m)$ for $\Delta_T = 0$ (top) and $\Delta_T = -541$ (bottom). The aimed center of the drift Δ_{T_0} is drawn as vertical dashed line.

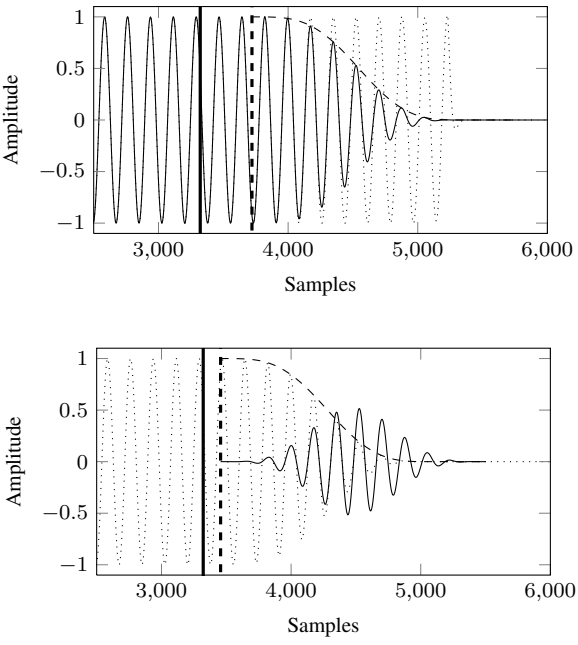

Figure 2: Top shows the output signal before compensation (solid) and with compensated output envelope (dotted). Position t_s^c is marked by the vertical line and dashed line marks the output envelope at position $t_s^c + R_s$. Bottom depicts the reapplied envelope at $t_i = t_s^c + \Delta_n$ (dashed) and finally the overlapped frame x_c at new position t_i ($\Delta_n = 134$).

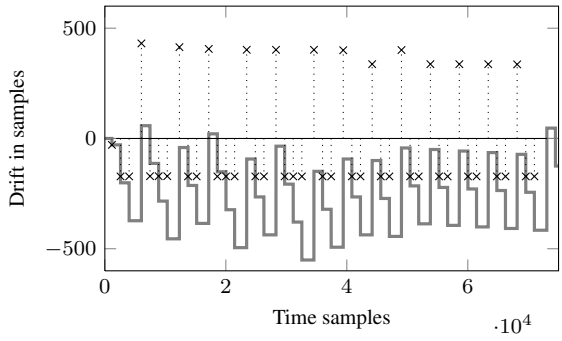

Figure 3: Cumulated drift Δ_T (thick gray) and time shifts Δ_n (dotted) at reset instants for an audio sample ($R_s = 400$).

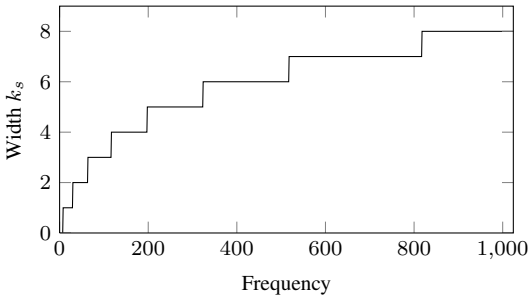

Figure 4: Peak width k_s with a maximum value $k_N = 8$ in dependency of the frequency for a $N = 2048$ bin spectrum.

been generated as in section 2.2, the envelope in the range needs to be compensated to one before the artificial output envelope from equation (13) is applied. Therefore, in a first step the samples in the range $[t_s^c + R_s \dots t_s^c + R_s + N[$ are divided by the envelope $e(n)$. To avoid rounding errors at the low end, the division is only accomplished for $e(n) > 10^{-3}$. Afterwards $e(n)$ is reapplied onto $y(n)$ at the shifted output position t_i and the reset frame x_c is overlap-added. Figure 2 depicts the steps from compensation until addition of x_c. Remember that after adding x_c, the output signal was complete up to $t_s^c + R_s > t_i$. So the operations described above only affect the envelope and not the signal content itself.

It is possible for a potential Δ_n that the sum $\Delta_T + \Delta_n$ will exceed $[T_l \dots T_h]$ although the process favours time shifts in a way to keep the drift in that range. If this occurs, the reset will be postponed and is retried immediately at the next time frame. Theoretically this could even inhibit the reset for a long time but testing with many audio signals proofed that in reality the reset is only delayed in rare situations and by few frames.

In figure 3 the cumulated drift and time shifts at reset instants are plotted for a short audio sample. It can be seen that the cumulated drift always remains in a small area around the zero offset $\Delta_{T_0} = -200$ and is smoothly compensated as desired.

3.2. Polyphonic signals

The next section focuses on an improved performance when processing polyphonic signals. By carefully listening to some test signals it could be noticed that the amplitude modulation is limited to non-voiced or weak components, whereas strong sinusoids are not influenced. One possible explanation is that the cross correlation achieved during the reset is dominated by strong sinusoidal components while it cannot give a good prediction of noisy parts. This leads to the idea to limit the reset to sinusoidals only, as the resulting Δ_n is a valid measure to guarantee a coherent overlap for these. Noisy or unvoiced parts, for whom Δ_n is not a good measure, are better processed by the standard phase vocoder as it already handles these kind of signals at sufficient quality.

3.2.1. Sinusoidal peak detection

Initially the bins in the signal spectrum have to be classified into being sinusoidal or not. This is not a trivial task and lot of research has already been done in that area. But for a first proof of concept a quite common and easy detection seems to be sufficient that could still be improved afterwards if necessary. By the assumption that sinusoidal components have a peak-like character, one of the simplest solutions is a local maximum detection that labels a bin as peak when its amplitude is bigger than the k_s neighbouring bins. The approach from [8] expands this principle and introduces a frequency dependant k_s for a scaled locking phase vocoder. This is motivated by the nature of the human hearing that also has a similar frequency dependant resolution. The same idea applies to the proposed peak detection but instead of using an arbitrary scale, we decided to calculate the width k_s by the Mel-Scale. There is no single Mel-Scale formula and several different variations exist, but they all define a similar logarithmic curve and all share the mapping of exactly 1000 Hz to 1000 Mel. In case of a discrete spectrum the conversion from a bin number k to the corresponding Mel value

$$M(k) = 2595 \cdot \log_{10}\left(1 + \frac{f_s \cdot k}{N \cdot 700\text{Hz}}\right) \tag{19}$$

is obtained from scaling with the sample frequency f_s and the FFT length N. Further scaling is needed to limit the output range to a desired maximum peak width k_N, which directly leads to

$$k_s(k) = \text{round}\left(\frac{M(k)}{M(0.5N)} \cdot k_N\right) \tag{20}$$

expressing the peak search width in dependency of the frequency bin k. The change of $k_s(k)$ over frequency for a maximum width of $k_N = 8$ is shown in figure 4. For small values of k the search width is zero and so every bin is selected and not only peaks. But as the FFT resolution is too low to distinguish single sinusoidals at low frequencies, it is reasonable to select every bin as a potential peak in this area. The collection of detected peaks might be further reduced by an absolute threshold slightly above the expected noise floor.

3.2.2. Sinusoidal reset

Instead of inserting the reset frame x_c directly into the output data it is now at first processed by the standard phase vocoder algorithm with the synthesis hop size R_s in equation (7) replaced by $R_s' = R_s + \Delta_n$. Adding this modified frame to the output at $t_s^c + \Delta_n$ would seamlessly continue the waveform just with a locally modified time stretch factor. To perform the actual reset, the phases for the two nearest bins on both sides around a detected peak k_p are set to the input phase prior to the inverse FFT. Because of the applied time shift Δ_n this coherently resumes the si-

nusoidals while also maintaining proper overlap of all other components that were already processed by the phase vocoder.

As only sinusoidals are reset it is now feasible to limit the calculation of the cross correlation by a spectral mask to the detected peaks. This would yield a little more precise estimation of the time shift Δ_n.

4. OVERALL SYSTEM

The overall motivation was to integrate the modified PVSOLA algorithm into a variable pitch shifting and time stretching engine. As the original algorithm and the phase vocoder in general can only perform time stretching, an additional resampling stage has been added to perform pitch shifting.

To preserve the formants and character of speech when the pitch is shifted, the *True Envelope* algorithm [9] yielded better results compared to a simple linear predictor based envelope estimation at only slightly higher computational cost.

Transients are already preserved quite well with pure PVSOLA in contrast to other phase vocoder techniques. By the regular reset the probability is quite high that a reset occurs around a transient position. The reset copies the phase from the input and reintroduces the typical phase jump at the transient position in the output signal. Problems arise when the reset is triggered short before or after a transient which may cause an audible doubling or damping of the transient.

The proposed solution is to implement a transient detection and then synchronize the reset with the transient position. The detection is based on [10] that uses the center of gravity of a spectral peak together with a stochastic model to determine if it is part of a transient event or not. This fits well into the proposed framework because the phase calculation is already concentrated on single peaks when phase locking is used like it is described later. As soon as the beginning of a transient is detected the regular reset is suppressed until the end of the transient event is recognized. Then the reset is triggered immediately and this reliably transfers the phase change from the input to the output signal.

5. DISCUSSION

The enhanced algorithm has been tested with various audio signals containing speech, singing voice, monophonic and polyphonic instruments as well as transient material and complete song mixes. Pitch shifting and time stretching factors have been varied in a range from $[0.5 \dots 2]$. All test files were sampled at 44.1 kHz and a frame length of $N = 2048$ samples together with a Hann window have been used. The analysis hop size was set to $R_a = {}^{N}/_{8}$.

The amount of overlap, the pitch and stretch factors as well as the frame length have a direct influence on the maximum time shifts T_l and T_h. In this context it is important that the periodicity of the lowest occurring frequency is small enough to fit at minimum once in the range $[T_l \dots T_h]$. Otherwise, the maximum of the cross correlation would maybe indicate a high correlation for a harmonic instead of the fundamental frequency. As the selection range $[T_l \dots T_h]$ is windowed and shifted, the minimum frequency even needs to be higher to achieve a reliable detection in all cases.

5.1. Informal observations

With the new algorithm the overall artefacts for polyphonic material and complete songs were considerably reduced, although the limited reset re-establishes a little bit of phasing. The increase of phasiness is noticeable most with pure monophonic speech signals and these do not reach exactly the same quality as PVSOLA. Nevertheless, the general audio quality for these files is still far better than other advanced phase vocoder techniques, like for example the pure scaled locking phase vocoder from [4].

Concerning the reintroduced phasiness, a strong influence of the parameters of the sinusoidal peak detection described in section 3.2.1 has been identified. The selection of k_N, and by this the number of peaks detected in high frequency regions, revealed a strong dependency on the amount of phasing or amplitude modulation artefacts. When more peaks, and even weak peaks, are detected there is a higher chance to create phase discontinuities during a reset. This would produce the amplitude modulation known from the original PVSOLA. On the other hand when there are less detected peaks, a lower percentage of the spectrum is regularly reset leading to a higher amount of audible phasiness. A value of $k_N = 6$ was found as a good compromise in our case.

The integration of a scaled locking phase vocoder with sinusoidal trajectory estimation and frequency dependant peak detection like [8] further improved the results. *Scaled Phase Locking* is a technique originally described by Laroche [4] to better preserve vertical phase coherence. This is mainly achieved by limiting the phase calculation to spectral peaks and by locking the area around a peak to the central phase. Furthermore, the phase relation around a peak is transferred from the input to the output phase which preserves typical phase relations and establishes vertical coherence. Applied to the proposed PVSOLA variant this would decrease phasiness in non-sinusoidal areas that are usually not reset. While the sound becomes more clear for most test files when using phase locking, it has to be noted that it also adds a bit of a metallic character to some files. Again this is most perceivable with speech signals in comparison to the unmodified PVSOLA. As a side effect from using scaled phase locking the minimal analysis overlap could possibly be reduced from 75 % to 50 % which effectively cuts down the processing time to one half [4].

5.2. Formal listening test

A formal listening test based on the *MUlti Stimulus with Hidden Reference and Anchor* (MUSHRA) [11] method has been carried out. With the common MUSHRA test the participants rate the quality of a signal processed by different algorithms in comparison to the unmodified reference signal. Furthermore, the reference and an anchor signal is hidden in between the test signals. A qualified listener should reliably rate the hidden reference with a nearly equal quality compared to the real reference, while the anchor signal should be rated as the worst signal.

Because it is hard to judge the audio quality of a time-stretched or pitch-shifted signal compared to an unmodified reference, we decided to use the presented algorithm as the reference and let the participants rate the quality of the other algorithms relative to this. The rating was done on a nine point verbal scale ranging from *much worse* over *equal* to *much better* that later became linearly mapped to a numerical range of $[-4, 4]$ for statistical evaluation.

Altogether five algorithms were compared: the original PVSOLA, the presented variant of PVSOLA, a simple phase vocoder and two state of the art commercial algorithms. The 7 test signals transient drums, transient guitar, polyphonic voice, complete mix I/II, solo voice and male voice covered a wide range of audio signal types. To keep the whole duration of the test reasonable only

	Comm. 1 $\overline{x}$ / σ^2	Comm. 2 $\overline{x}$ / σ^2	Phase voc. $\overline{x}$ / σ^2	PVSOLA $\overline{x}$ / σ^2
Trans. Drums	1.53 / 1.41	3.13 / 1.27	-1.07 / 1.35	-2.93 / 1.78
Trans. Guitar	1.80 / 1.31	0.93 / 2.35	-1.40 / 1.40	-2.07 / 1.50
Polyph. Voice	2.00 / 1.86	1.47 / 0.98	0.33 / 4.10	-2.67 / 1.38
Compl. Mix I	1.33 / 1.10	1.33 / 2.81	-1.93 / 0.50	-3.07 / 1.07
Compl. Mix II	2.13 / 2.98	1.47 / 1.55	-0.13 / 3.27	-2.53 / 1.12
Solo Voice	1.07 / 2.07	1.27 / 3.50	-1.67 / 1.38	-2.87 / 1.27
Male Voice	-1.93 / 0.64	2.00 / 2.00	-2.47 / 0.70	0.87 / 3.12
Overall	1.13 / 3.23	1.66 / 2.40	-1.19 / 2.56	-2.18 / 3.17

Table 1: Mean value $\overline{x}$ and variance σ^2 of all single tests as well as an overall measure. All ratings are relative to the proposed algorithm.

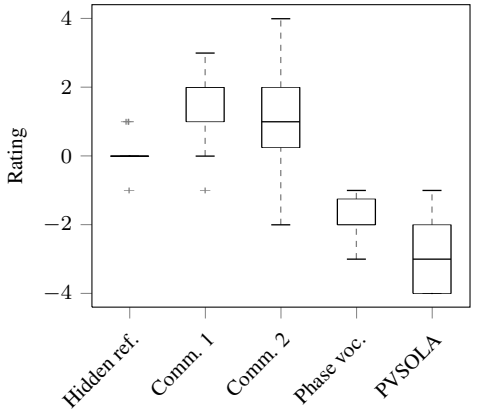

Figure 5: Exemplary box plot for *Complete Mix I* test file. The proposed algorithm is a clear improvement compared to PVSOLA though it does not reach state of the art commercial algorithms.

time stretching by a factor of 1.53 was investigated. The parameters from the proposed algorithm are chosen as mentioned before. Furthermore phase locking and transient preservation have been enabled.

A total number of 17 people with a background in audio processing took part in the test. Two of them regularly failed in finding the hidden reference and their results were not used in the final evaluation. The mean values and variances of the results from all single tests as well as an overall measure are shown in table 1 and figure 5 visualizes the *Complete Mix I* results in form of a box plot. The ratings have to be understood relative to the proposed algorithm.

It is apparent that the simple PVSOLA algorithm received a significant negative rating for 6 out of 7 test files and an overall mean rating of -2.18. This confirms that the proposed changes improve the sound quality for time stretching on a wide variety of signals compared to the previous PVSOLA implementation.

As mentioned before the authors expected a slight advantage for PVSOLA on monophonic speech, but this could not be verified by the test results. *Solo Voice* was clearly rated negative and though *Male Voice* has a slight positive tendency, the quite high variance of 3.12 shows that the listeners could not agree on a clear choice in this case. From these two files it is evident that even on signals were PVSOLA was expected to be superior, the proposed changes at least do not lead to any measurable degradation in quality.

The commercial algorithms are consistently rated better than both the original and improved PVSOLA, although they excel on different signal types and sometimes have a big variance in the results. The simple phase vocoder did receive worse ratings than the improved algorithm but was generally better than PVSOLA. So the goal to use the phase vocoder as an anchor signal to mark the lower bound of expected quality was not met. It seems that most listeners perceived the harsh amplitude modulation of PVSOLA as more disturbing than the phasing artefacts.

Overall the listening test successfully proofed the advantages of the modified PVSOLA algorithm. However, the quality of commercial engines is not reached. The complete results of the listening test can be accessed online[2] together with different audio examples to demonstrate the quality of the described algorithm.

[2]http://ant.hsu-hh.de/dafx12/modpvsola

6. CONCLUSION

In this paper a modified PVSOLA algorithm has been described that improves the quality on polyphonic audio signals. This was achieved by detecting sinusoidal peaks in the spectrum and establishing a separated processing for sinusoidal and residual components. This means to limit the regular reset to the detected peaks but also to integrate a scaled locking phase vocoder to calculate the phase propagation restricted to sinusoidal peaks. Compared to the original PVSOLA, the artefacts on polyphonic audio are significantly reduced while monophonic speech is still at high quality but with the addition of little more phasing. The formal listening test proofed the advantages of the described changes compared to the unmodified PVSOLA algorithm.

Furthermore, the computation of the cross correlation and the time shifts have been simplified. On the one hand this reduces the minimum amount of frames required in the input buffer and on the other hand permits a sliding compensation of the cumulating time drift.

The detection of sinusoidal peaks and the estimation of tracks was quite simple in the current form. Some glitches when using the scaled locking phase vocoder, in particular at note changes or vibratos, are probably caused by this simplicity and it would be interesting to see if a more complex detection and tracking algorithm could improve this. Also the strong influence of the parameters of the peak detection on the audio quality should be further investigated.

7. REFERENCES

[1] J. L. Flanagan and R. M. Golden, "Phase vocoder," *Bell System Technical Journal*, 1966.

[2] Michael R. Portnoff, "Implementation of the digital phase vocoder using the fast Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 3, pp. 243–248, 1976.

[3] Miller Puckette, "Phase-locked vocoder," in *Proceedings of 1995 Workshop on Applications of Signal Processing to Audio and Accoustics*, 1995, pp. 222–225.

[4] Jean Laroche and Mark Dolson, "Improved phase vocoder time-scale modification of audio," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 3, pp. 323–332, May 1999.

[5] Axel Röbel, "A shape-invariant phase vocoder for speech transformation," in *Proc. 13th Int. Conf. on Digital Audio Effects*, 2010, pp. 1–8.

[6] Alexis Moinet and Thierry Dutoit, "PVSOLA: A Phase Vocoder With Synchronized Overlapp-Add," in *Proc. of the 14th Int. Conference on Digital Audio Effects*, 2011, pp. 269–275.

[7] Udo Zölzer, *DAFX: Digital Audio Effects*, John Wiley & Sons, 2011.

[8] Thorsten Karrer, Eric Lee, and Jan Borchers, "PhaVoRIT: A phase vocoder for real-time interactive time-stretching," in *Proc. Int. Computer Music Conference (ICMC)*, 2006, pp. 708–715.

[9] Axel Röbel and Xavier Rodet, "Real time signal transposition with envelope preservation in the phase vocoder," in *Proc. Int. Computer Music Conference (ICMC)*, 2005.

[10] Axel Röbel, "A new approach to transient processing in the phase vocoder," in *Proc. of the 6th Int. Conference on Digital Audio Effects*. 2003, pp. 344–349, Citeseer.

[11] ITU-R, "RECOMMENDATION ITU-R BS.1534-1: Method for the subjective assessment of intermediate quality level of coding systems (01/03)," Tech. Rep., 2003.