15th European Signal Processing Conference (EUSIPCO 2007), Poznan, Poland, September 3-7, 2007, copyright by EURASIP

MULTI-RESOLUTION SOUND TEXTURE SYNTHESIS USING THE DUAL-TREE COMPLEX WAVELET TRANSFORM

Deirdre O'Regan and Anil Kokaram

ABSTRACT

This paper presents a novel algorithm for Sound Texture Synthesis. Inspired by the well known Efros and Leung non-parametric 2-D Image Texture Synthesis algorithm, our 1-D interpretation is used to synthesise long, perceptually and statistically similar sound textures from much shorter real-world audio training examples including crowd noise, a baby crying and speech. The process employs the Dual-Tree Complex Wavelet Transform to reduce computational complexity without sacrificing spectral coherency in the synthesised audio. Our approach produces plausible and interesting sound textures that are comparable to the results of other state-of-the-art algorithms.

1. INTRODUCTION

Sound Texture Synthesis (hereafter STS) has many definitions [1, 2]. Generally, the goal is the synthesis of a longer, perceptually similar body of audio from a much shorter training example. The training sample may be noise-like or stochastic, quasi-periodic, or a mixture of these as in most real-world sounds. The biggest challenge of STS is the achievement of an acoustically plausible sound texture with an unpredictable temporal evolution. Simple repetition of the training example, however smoothly it is "tiled", is easily detectable acoustically and should be avoided. Training candidates for STS include natural (e.g. babbling water), human (e.g. baby crying), musical (e.g. piano), and mechanical (e.g. road traffic) sound samples. Each of these genres present unique challenges for STS. Some natural sounds seem stochastic (e.g. heavy rainfall), whereas human speech and polyphonic music have specific, complex structures. Applications of STS include audio compression, ambient music synthesis for computer games, installations and movies, re-synthesis of rare sounds (e.g. a rare bird call), and error correction or "hole filling" in existing, damaged audio tracks.

STS involves the *sample-wise* synthesis of a longer, novel sound track. This is subtly different to the idea of Audio Texture (hereafter AT) as defined in [3], which is concerned with the location of "transition points" for randomized playback of whole segments of the original training example [3, 4, 5]. AT is like a "patch-based" alternative to the unit-based STS. Our favouring of the synthesis approach is inspired by the interesting results and challenges emerging from the field of Image Texture Synthesis (hereafter ITS) in recent years [6, 7, 8].

Image texture has been successfully modeled as a Markov Random Field (hereafter MRF), meaning that *"the probability distribution of brightness values for a pixel given the brightness values of its spatial neighborhood is [assumed to be] independent of the rest of the image."* Efros & Leung [6]. This idea is derived from a statistical technique first used by Shannon to generate English-like text letter by letter. Using a large sample of training text, Shannon modeled language as a generalised Markov Chain enabling him to estimate the probability distribution for each new letter to be synthesised by measuring from the existing data. As discussed in [6], image texture can be pixel-wise synthesised using this technique adapted to image space.

Although image and audio are presented and perceived quite differently, we assume that a similar technique exists for sound texturing. Often with $44k$ samples per second, however, it is clear that sample-wise synthesis in sound space is not practical. We propose the use of wavelet space instead. Wavelet, or Multi-Resolution Analysis (hereafter MRA) involves the analysis of a signal with a finite energy basis or *mother* wavelet function under various translations and dilations. Wavelet decomposition is conducive to the spectral analysis of non-stationary, real-world signals due to its useful *time-scale* signal breakdown. This *octave* filtering is thought to bear similarity to that of the Human Auditory System [9]. Furthermore, this property can be exploited for computational efficiency in N-D sample-wise synthesis [2, 7, 8].

The Discrete Wavelet Transform (hereafter DWT) is used in [4] for the location of AT transition points in several ambient audio samples. A real-time demo applet (see URL in [4]) allows experimentation with different mother wavelets and parameter tuning. Similarly, Dubnov et al use the DWT for STS in [2]. Here, analysis of the training example results in a nodal MRA tree representing the levels of the DWT. A novel audio texture is created through the breath-first nodal synthesis of a new tree, such that each new node is chosen by the suitability of both its same-level "predecessors" and hierarchical "ancestors" for coherency with its same-level neighbour. Also drawing from an existing ITS technique [7], this algorithm produces good results for a variety of real-world audio training examples. The sound files are obtainable online at the URL associated with [2].

Our algorithm makes use of the Dual-Tree Complex Wavelet Transform (hereafter DT-CWT) [10] for MRA. Our transformed signal is represented as an "inverse pyramidal" structure with large-scale feature coefficients at the bottom, these features decreasing in scale as we move up the levels of the pyramid. Sound texture is first synthesised at the largest scale, which represents the coarsest level of detail in the signal. Other levels are textured by means of "coarse-to-fine" scale coefficient propagation. A variety of audio training examples are tested, and the resulting sound textures evaluated for spectral coherency and acoustic desirability. Due to its similarity to our approach, the training examples used by Dubnov et al [2] are included in our test set for benchmarking purposes.

2. SINGLE RESOLUTION STS (SR-STS)

The idea of SR-STS is necessary in the comprehension of its multi-resolution extension (MR-STS) using the DT-CWT. Essentially the 1-D application of [6], a brief explanation of SR-STS is now needed.

Suppose that our unit of synthesis, y_s, is a *single sample* of audio, with $22kHz$ resolution for example. Let Y_s be a long body of audio with N samples to be synthesised from Y_e, which is a shorter audio example of n samples. It is assumed that Y_e is long enough to approximate the statistical distribution of the underlying, infinite texture from which both Y_e and Y_s are derived. Initialisation of an empty Y_s involves the copying of a short series of samples, or "seed" from Y_e to a region in Y_s. In keeping with [6], this region might be placed at the mid-point of Y_s, although this is somewhat counter-intuitive for audio, as we shall see later.

Proceeding from the boundaries of the seed outwards, let $y_s \in Y_s$ be the next sample to be synthesised, and $W(y_s)$ be the neighbourhood of samples of length w centered at y_s. An approximation to the conditional probability distribution $P(y_s|W(y_s))$ must now be constructed. We begin by identifying the neighbourhoods in Y_e that are similar to $W(y_s)$. Let $d(W(y_s),W(y_e))$ represent the *perceptual distance* between some $W(y_s)$ and the same-sized neighbourhood centred at y_e at some point in Y_e. In keeping with [6], d is defined as the Sum of Square Differences:

$$d_{y_s \in Y_s, y_e \in Y_e}(W(y_s),W(y_e)) = \sum_{i=0}^{w} \frac{G_i V_i \sqrt{[W_i(y_s) - W_i(y_e)]^2}}{\sum_{i=0}^{w} G_i V_i} \quad (1)$$

where G is a 1-D Gaussian kernel of length w and variance $\sigma = w/6.4$ whose purpose is to maintain local temporal coherency in the synthesised texture, and V is a binary vector that is only non-zero where samples Y_s have already been filled. Note that the length of the initialising seed must be at least equal to $w/2-1$ to ensure a valid synthesis of the first case of y_s, but need not be any longer theoretically.

The neighbourhood $W(y_e)$ most similar to that of $W(y_s)$ corresponds to $W_m = MIN_{y_e \in Y_e}(d(W(y_s),W(y_e))$. All neighbourhoods in Y_e with $d < (1+\varepsilon)d(W(y_s),W_m)$ are used to construct $P(y_s|W(y_s))$ in the form of a histogram which is then randomly sampled, yielding y_s. According to [6], the value of ε can be varied to encourage or suppress variation in the synthesised texture.

We assume that $P(\cdot|\cdot)$ is valid if the length of w (which defines the MRF neighbourhood) is chosen to incorporate the longest repeating temporal feature in Y_e. Here, we are inspired by the spatial case of this assumption for image texture [6]. The unit of a sample is analogous to that of a pixel for image in the 2-D case. Unfortunately, the computational burden is potentially heavier than that for image due to the difference in perception and associated higher sampling rate of audio.

3. MULTI-RESOLUTION STS (MR-STS)

The DT-CWT [10] is an excellent means of multi-resolution signal analysis that has already been exploited for ITS [8]. It is known for its shift-invariant property that is also valid in audio application. In other words, the DT-CWT relates identical audio features occur

The DT-CWT uses a dual tree of wavelet filters which decompose the signal into multi-level (i.e. multi-resolution) complex wavelet coefficients. Analysis with the mother wavelet in one tree results in the real values, while analysis with the Hilbert transform produces imaginary values in the other. At each level of decomposition, the DT-CWT produces a high-pass complex signal of *detail* coefficients and a low-pass real signal that is passed on to the next level. Our MR-STS makes use of Type C wavelet filters in [10] which are deemed optimal in terms of shift invariance. The Q-Shift structure of the decomposition implies that each level k coefficient has two complex "children" located symmetrically above it at level $k-1$. Thus we end up with an "inverse pyramidal" structure whose increasing levels capture coarse-to-fine scales of detail in the signal.

The salient quasi-periods of a signal are often visible as "peaks" in the resulting band-pass detail signals, particularly in coarser levels of the DT-CWT. Recall that the length of w in SR-STS (Sec. 2) is chosen to reflect the longest repetitive temporal feature in the training example. In the DT-CWT of the signal, a fairly small w on the coarsest level of detail can be used to capture this large-scale period. Given an audio example of n samples, Y_e, and desiring a longer audio texture of N samples, Y_s, we can perform a reduced-complexity MR-STS as follows:

- A K-level DT-CWT is performed on Y_e. Knowing N, the dimensions of the K band-pass complex and final low-pass real signals needed to reconstruct Y_s are calculated and used to initialise appropriate empty signal containers (e.g. vector arrays).
- A sample seed of the decomposed Y_e is placed in the low pass, and band-pass signal containers at each level. The size and position of this seed follows the parent-child relationship described earlier, such that the seed placed at level $k-1$ is twice the length of that placed at level k, and symmetrically above it. Again, strictly adhering to [6] implies placing these seeds centrally. This idea was later questioned, as mentioned in Sec. 4.
- All null coefficients for each level k must now be synthesised. At the coarsest scale (i.e. level $k = K$) the algorithm described in Sec. 2 is used to synthesise the detail coefficients. As we are now dealing with complex numbers, Eqn. 1 is modified:

$$d = \sum_{i=0}^{w} \frac{G_i V_i \sqrt{[Re\{W_i(c_s) - W_i(c_e)\}]^2 + [Im\{W_i(c_s) - W_i(c_e)\}]^2}}{\sum_{i=0}^{w} G_i V_i} \quad (2)$$

  where are c_e and c_s are the complex wavelet coefficients at level $k = K$ of the wavelet decomposition of Y_e and Y_s respectively, and W, w, G and V are as previously described.
- Once a value for a particular c_s has been chosen, the levels $k = K-1..1$ are updated in keeping with the parent-child relationship. In other words, the two coefficients above c_s at level $k = K-1$ are copied from those above the location of the c_e used to synthesise c_s, and so on until level $k = K$ (and therefore all levels above) has been completely filled.
- Finally, the synthesised structure is inverse transformed yielding the longer sound texture, Y_s.

This coarse-to-fine coefficient propagation has a huge computational advantage over the SR-STS described in Sec. 2. This saving depends on both the the depth of the DT-CWT decomposition, K and associated reduction in neighbourhood size, w. Due to our MRF assumption, w should capture the dominant tempo (which may be very small if the signal is nearly stochastic). We try to choose the greatest possible value of K with corresponding w that does not compromise the quality of the resulting sound texture Y_s.

Referring to the similar work of Dubnov et al [2] for comparison, we note that our MR-STS algorithm is only concerned with predecessor (i.e. same level) coherency on level $k = K$, and propagates local ancestral (i.e. parent-child) coherency up the pyramidal hierarchy. Our algorithm does not, however, guarantee local temporal coherency on levels above K. Dubnov et al are concerned with both dimensions of local spectral coherency, echoing the ITS technique of Wei and Levoy [7] in their approach. Although our algorithm seems more simplistic, a comparison of our results with those of Dubnov et al is interesting.

4. RESULTS AND DISCUSSION

Training examples obtained from the URL associated with Dubnov et al [2] were tested first with our algorithm, followed by some other sound samples of interest. Our sound textures were synthesised to be much longer than the training examples to allow for the temporal emergence of variation and novelty, or undesirable looping, tiling, and uncomfortable artifacts (known as "clicks").

4.1 Dubnov et al Training Examples and Textures

Dubnov et al create plausible sound textures from a selection of real-world training samples. These samples, their sampling rates, Fs, original time durations, t1, and the durations of Dubnov et al's sound textures, t2_d, are listed in Table 1. It is clear that some of these textures are actually *shorter* than the training examples.

Dubnov et al's textures of training examples 1 and 2 sound interesting, with few clicks and good variety. The texture of 3 - traffic jam - has short, repetitive loops, some clicks and abrupt silences. The latter effect could occur because a period of recorded silence at

	training example	Fs[Hz]	t1[s]	t2_d[s]
1	drum loop	22k	3	5
2	baby crying	11k	13	11
3	traffic jam	22k	22	35
4	shore, splashing	11k	18	11
5	formula 1 race	11k	16	23

Table 1: Training examples used, and sound texture durations, t2_d, achieved by Dubnov et al. Training set and results are obtainable online [2].

the start of the original training example. Although generally plausible, the domineering presence of long-term car-horn "honking" is not reflected well in the synthesised texture.

The authors define a specific problem unsolved by their algorithm; the accurate re-synthesis of some quasi-periodic signals with both long-term patterns, and sporadic short-term periodic "events". Example 4 is in this category - it sounds like the relaxed tempo of water ebbing on a shore, punctuated with sudden, short "splashing" events. Here, the authors note an unrealistic "nervous splashing activity" in their texture synthesis. Perhaps this effect is due to the checking of only 5 predecessors at each level of the DWT. Example 5 - formula 1 race - presents a similar problem. A good balance between the "long sound phenomenon" of gradual engine acceleration and short-term "gear-shifting activity" is not well reflected in the resulting sound texture.

4.2 Comparative Results from our STS Algorithm

Table 2 summarizes the application of our algorithm to the Dubnov et al training examples, which we label as Y_e in keeping with the notation of Sec. 3. The values for parameters K, w, and ε which produced the best results are listed, along with the new durations, t2_o, of our sound textures.

Y_e	training example	K	w	ε	t2_o[s]
1	drum loop	8	41	0.3	63
2	baby crying	6	25	0.1	73
3	traffic jam	8	5	0.01	70
4	shore, splashing	8	51	0.1	74
5	formula 1 race	8	21	0.1	75

Table 2: Parameter values used, and durations, t2_o, of the best sound textures achieved by applying our STS algorithm to Dubnov et al training examples

Extensive trial-and-error experimentation was carried out until the best parameter combinations were found. In all cases, very short seeds (e.g. $0.4s$ for Y_e 2) initialized the levels of Y_s, and the mechanism of coarse-to-fine coefficient propagation from the coarsest level, K, did not seem to compromise spectral coherency. The best sound textures are generally plausible, longer than those of Dubnov et al, and sound smooth and varied (i.e. not tiled). All of the experimental sound files - including those listed in Table 2 - can be obtained from our webpage `http://www.deirdreoregan.com/STS_EUSIPCO.html`.

The best sound texture of Y_e 1 - drum loop - is varied and interesting, with cymbals appearing pseudo-randomly in time. This particular value for w appears to roughly correspond to the tempo of the piece at $K = 8$ in the DT-CWT, as can be seen in Fig. 1 (left). During experimentation, the origin and placement of the seed was important, with strange differences between the backwards and forwards synthesised results emerging for a central seed (e.g. a novel beat structure evolving and locking into tempo after the seed). The seed was later moved to the beginning of the texture. The waveforms of the training sample and $11s$ of our best drum texture are compared in Fig. 1 (centre and right).

A spectrogram of Y_e 2 - baby crying, and $30s$ of our best sound texture is shown in Fig. 2 (left) and (right) respectively. The spectral energy of the latter looks plausible and evolves temporally. The texture sounds equally plausible and varied, with little audible clicking. When experimenting with DT-CWT levels $K > 6$, smaller values of w produced repetitive looping of particular sections of the training example, whereas larger values resulted in tiling of the whole sample. Increasing the value of ε merely resulted in clicking and garbled sound texturing.

Recorded silence at the beginning and ends of Y_e 3, 4 and 5 were excluded to avoid tiling in Y_s. In general, our algorithm seems to favour amplitude troughs in Y_e as randomization points for Y_s. Silence has low amplitude, and so there is a tendency toward end-to-end tiling with Y_e 3, 4 and 5 in our STS. Once modified, however, good results could be achieved for these training examples with the optimal parameter values listed in Table 2.

The best sound texture of Y_e 3 - traffic jam - fully reflects the annoying ambience of long-term car-horn honking nicely overlayed with bursts of shouting from irate drivers. Careful tuning of the parameter w resulted in a plausible reflection of the quasi-periodic tempo in the textures of Y_e 4 and 5. Both the "long sound phenomena" and overlayed short-term "events" associated with these training examples are well represented and balanced.

4.3 Stochastic, Music and Speech Sound Textures

A further selection of training examples was chosen to test the robustness of our algorithm to near-stochastic (i.e. baseball game crowd chatter) and structured (i.e. speech, music) sound samples. Table 3 lists these samples, their durations, $t1$, and the algorithmic parameters used to produce the most plausible sound textures of duration $t2$.

Y_e	training example	t1[s]	K	w	ε	t2[s]
6	crowd chatter	13	7	11	0.1	71
7	piano phrases	26	8	51	0.3	146
8	german speech	12	6	51	0.001	60
9	english speech + music	10	8	201	0.1	70

Table 3: Further training examples, Y_e, their durations, t1, and parameters used to produce the best sound textures of duration, t2.

The texturing of Y_e 6 - crowd chatter - was very interesting. The noisy crowd ambience was easily reproduced in the synthesis, and w could be quite small due to its fairly stochastic nature. During experimentation, any tiling of the training example was detectable through the pattern of a vendor shouting "nuts" and a man talking and laughing over the noise. Our best texture somewhat randomises these "event-on-noise" patterns, although a slight "whirring" is present if the shouted word "nuts" emerges more often than in the original sample. However, it is suspected that this artifact - and some tiling - would go happily unnoticed if this sound texture were to be used as low-volume background ambience.

Y_e 7 - piano phrases - can be textured in a number of ways. With large w, the two phrases emerge in their original order, as can be seen in Fig. 3 (left). Use of the parameters listed in Table 3 results in a random ordering of the phrases, as can be seen in Fig. 3 (c. left). At DT-CWT levels $K > 10$ and with short w, these phrases can be decomposed into much smaller units. Fig. 3 (c. right) demonstrates the breakdown of the original phrases to almost *single notes* with $K = 15$, $w = 3$, and $\varepsilon = 0.01$. This texture sounds like short, unpredictable bursts of piano played erratically! Further decomposition to level $K = 17$ and $w = 3$ results in the emergence of the original phrases once again. This is intuitive, since the neighbourhood size at this level is equivalent to $(3 \times 2^{17})/22kHz = 17.83s$, which represents about 80% of the original training example. However, there is unpredictable inter-phrase spacing, as can be seen in Fig. 3 (right).

Y_e 8 and 9 - German speech and English speech with background music, respectively - produced exciting textures! Table 3 lists the parameters used to texture German speech with good variation and minimal clicking. During experimentation, a breakdown

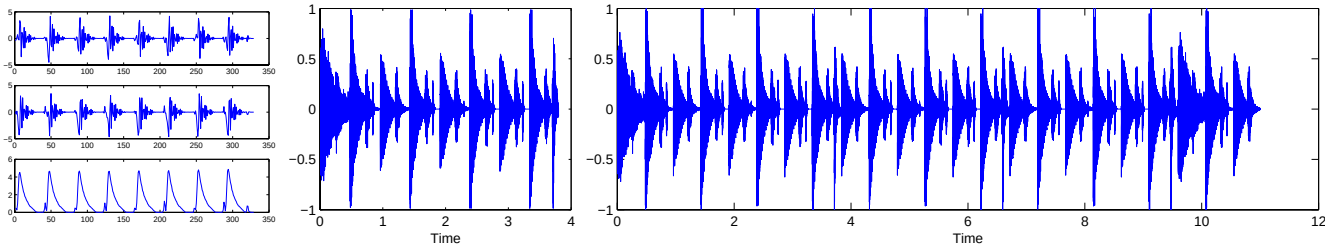

Figure 1: Synthesised drum texture: The real, imaginary, and absolute (t-b) values of the detail coefficients at $K = 8$ of the DT-CWT of the training example (left), the $3s$ training example (centre), and $11s$ of the sound texture with $1.5s$ seed at the start, $K = 8$, $w = 41$ and $\varepsilon = 0.1$ (right).

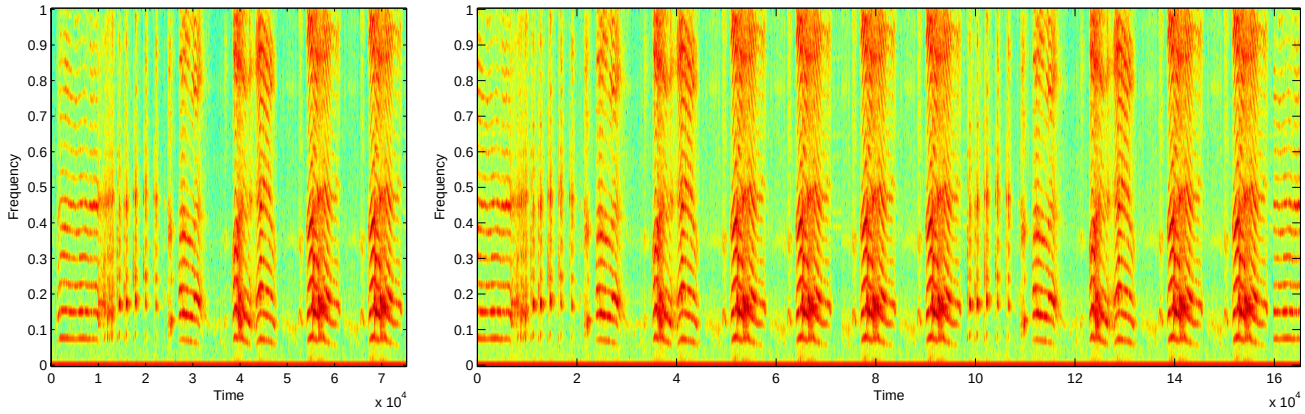

Figure 2: Synthesised baby crying texture: Spectrograms of the $13s$ training example (left), and $30s$ of the sound texture with $0.4s$ seed at the start, $K = 6$, $w = 25$ and $\varepsilon = 0.1$ (right).

of the structured language phrases seemed to occur at greater levels of the DT-CWT (e.g. $K = 15$) with a small neighbourhood size (e.g. $w = 3$). It is interesting to note that the value of parameter w used to texture English speech at level $K = 8$ is roughly equivalent in duration to that used for German at level $K = 6$ according to the dyadic DT-CWT. Perhaps this suggests that the units of German and English spoken at this relaxed tempo are similar in duration.

Fig. 4 (centre) shows the waveform and spectrogram of the first $5s$ of our best English speech sound texture. The high frequency inconsistencies seen in the spectrogram are transitional clicks that are barely audible on most sound systems. The presence of background music may be contributing to this effect. Perhaps a smoothing constraint on the coefficient values could be introduced to fix this problem. During experimentation, English speech could be decomposed to almost *phoneme level* with $K > 12$ and $w = 3$. Fig. 4 (right) shows the first $5s$ of this effect with $K = 13$, $w = 3$ and $\varepsilon = 0$. This particular texture contains short periods of distortion, but it is still interesting due to the seeming presence of words that were not spoken in the original training example!

5. CONCLUSION

We have described the success of our STS on a variety of real-world training examples including natural, mechanical, human and musical sounds. We have demonstrated the application of the Efros and Leung Image Texture Synthesis algorithm [6] to audio, and have reduced complexity by employing the Dual-Tree Complex Wavelet Transform for multi-resolution analysis and synthesis. The acoustic products of our research - along with a list of parameter values - are available online at `http://www.deirdreoregan.com/STS_EUSIPCO.html`. We conclude that our results compare favourably to other state-of-the-art STS [2], and AT [4, 5] algorithms whose results are available for acoustic comparison.

Further work will focus on a strategy for automatically choosing parameters, as we hope to render our algorithm truly non-parametric. The value of K for the DT-CWT, and the dyadically-related MRF neighbourhood size, w, should be chosen to balance computational efficiency with error minimisation. Inspired by the use of Entropy in (particularly wavelet-based) signal compression, we hope to use a similar technique to choose the optimal value of K for the DT-CWT.

Different classes of training example seem to respond to particular values of w, however. Our results suggest a link between w and the tempo of musical sound samples, for example. Perhaps Beat Detection [11] could be used to match the value of w to the tempo of rhythmic training examples. This would be a low-complexity operation on the coarsest scale of the DT-CWT for a large K. A classification technique could be used to choose a small w for noisy, stochastic signals, or set w to a multiple of the average length of a phoneme for speech signals. We could even detect the onset of phonemes as we synthesise speech signals and vary w to accommodate the length of the particular phoneme being synthesised.

The effect of the parameter ε has not been fully explored in this work. We note the tendency of our algorithm to pick up on low-energy troughs (including periods of silence) in the amplitudes of our training signals as coincidental "edit-points" that are more likely to be followed by variety in the synthesised signal. This tendency echoes the explicit objectives of transition-point location in many segmentation-based AT algorithms [3, 4, 5]. Perhaps ε could be tuned on the fly as amplitude troughs emerge or dissipate in the synthesis, or vary with the tempo of a percussive signal to amplify its effect at certain points in the cycle. We could also decrease ε on early detection of clicks or distortion in the sound texture to curb the propagation of errors.

We may also explore the possibility of applying SR-STS (see Sec. 2) to all levels of the DT-CWT simultaneously, or introduce multi-level coefficient predecessor searching to refine the existing method of simple parent-child "copying". This would liken our ap-

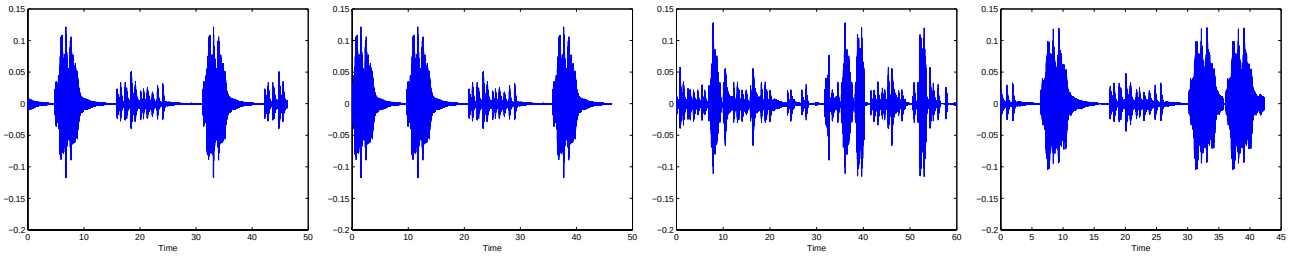

Figure 3: Synthesised piano texture: Tiling of the original phrase ordering (left), random ordering of fully-preserved phrases (c. left), structural breakdown and the formation of novel phrases (c. right), random phrase ordering *and* spacing (right)

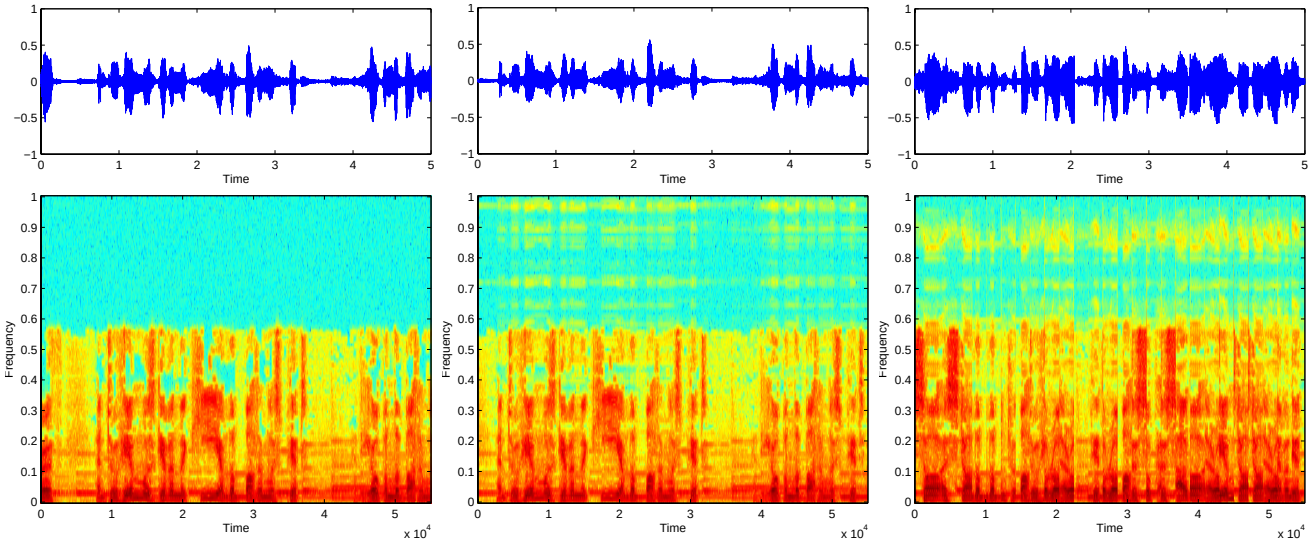

Figure 4: Synthesised English speech: $5s$ of the original training example (left), variation yet structural preservation with $K = 8$, $w = 201$, $\varepsilon = 0.1$ and $1.7s$ seed (centre), almost phoneme-level decomposition with $K = 13$, $w = 3$, $\varepsilon = 0$ and $0.9s$ starting seed (right)

proach to that of Dubnov et al in terms of spectral coherency, but some complexity would be gained. We have demonstrated that our algorithm produces equally plausible sound textures, so this refinement is not a priority. It would be interesting, however, to treat the non-decimated levels of the DT-CWT of our sound sample in its 2-D form as an image texture training example, and attempt to extend it spatially via ITS. To the best of our knowledge this has not yet been attempted for the synthesis of sound textures.

REFERENCES

[1] G. Strobl, G. Eckel and D. Rocchesso,"Sound Texture Modelling: A Survey," *Proc. of Sound and Music Computing (SMC)*, pp. 61-65, 2006.

[2] S. Dubnov, Z. Bar-Joseph, R. El-Yaniv, D. Lischinski and M. Werman, "Synthesizing Sound Textures through Wavelet Tree Learning," *IEEE Journal of Computer Graphics and Applications*, pp. 3848, 2002. `http://www.cs.huji.ac.il/labs/cglab/papers/texsyn/sound/`

[3] L. Lu, L. Wenyin and H-J. Zhang, "Audio Textures: Theory and Applications," *IEEE Trans. on Speech and Audio Processing*, vol. 12, num. 2, pp. 156-167, 2004.

[4] R. Hoskinson and D. K. Pai, "Synthetic Soundscapes with Natural Grains" *Presence*, vol. 16, num. 1, pp. 8499, 2007. `http://www.cs.ubc.ca/~reynald/naturalgrains.html`

[5] T. Jehan,"Event-Synchronous Music Analysis/Synthesis," *Proc. 7th Intl. Conf. on Digital Audio Effects (DAFx04)*, 2004. `http://web.media.mit.edu/~tristan/Blog/Music_Stretching.html`

[6] A. A. Efros and T. K. Leung,"Texture Synthesis by Non-Parametric Sampling," *Proc. Intl. Conf. on Computer Vision (ICCV)*, pp. 1033-1038, 1999.

[7] Y-L. Wei and M. Levoy,"Fast Texture Synthesis Using Tree-Structured Vector Quantization," *Proc. ACM Siggraph*, pp. 479488, 2000.

[8] C. Gallagher and A. Kokaram,"Nonparametric Wavelet Based Texture Synthesis," *Proc. Intl. Conf. on Image Processing (ICIP)*, 2005.

[9] P. E. Kudumakis and M. B. Sander,"Synthesis of Audio Signals Using the Wavelet Transform," *Proc. IEE Colloquium on 'Audio DSP - Circuits and Systems'*, Digest No: 1993/219, 1993.

[10] N. Kingsbury, "Complex Wavelets for Shift Invariant Analysis and Filtering of Signals," *Journal of Applied and Computational Harmonic Analysis*, num. 3, pp. 234-253, 2001.

[11] E. Scheirer, "Tempo and Beat Analysis of Acoustic Musical Signals," *Journal of Acoustic Society of America*, vol. 103, num. 1, pp. 588-601, 1998.