resyn2 vst
spectral resynthesis by xoxos

synthesis parameters

shift
Continuous adjustment of spectral contour, spectral or "formant shift". See spectrum (list) parameter for modes.

read
Position of read pointer in file in read mode, or speed of playback in play mode.

smear
Adds random value to phase increment between frames, creating varied effects depending on window length.

shift2, read2, mix
Synthesis parameters for second file, when cross synthesis is selected.

analysis parameters

PURPLE parameters perform file analysis when they are changed
BLUE parameters are only used during analysis

analyse
File analysis is performed automatically with some actions (eg. loading a new file). Otherwise, use this button to apply parameter selections.

lifter
Specifies separation of spectral contour data. Spectral resolution increases with higher values, but pitch colouration occurs (usually abruptly) above a certain point. Low values suffice. Leave it about a 1/4 way up if you don't know.

free range
When selected, analysis time is unlimited, affecting quality of resynthesis.

src
A sneaky trick param. If selected during analysis, the magnitude information for the original file, which is normally used in "spectrum off" mode, is replaced with the "source" information from the cepstral analysis (see detailed description).

frames
Number of frames used for analysis. The maximum is 1024 analysis frames for mono files, or 512 for stereo. More frames means better results, but eg. 1024 frames will use over 100mb of ram.

file
Loads wav file for analysis. Second button loads wav file for cross synthesis.

start, len
Allows selection of the wav file region to analyse.

list parameters

spectrum

sp.shift off	bypasses spectral shift calculation for efficiency and precision
sp.shift fix	retains constant spectral contour when repitched
sp.shift on	spectral shift active, no correction for pitch
sp.cross fix	cross synthesis with constant spectral contour
sp.cross on	cross synthesis with no spectral correction for pitch

overlap

The standard phase vocoder calls for 4 overlaps as a minimum to reduce sidebands. At 2x overlap, sidebands are as loud as the generated frequencies, creating dense timbres. The amplitude of sidebands drops as overlaps increase, dramatically from 2-5, then subsequent increases drop off a sideband here and there, then stop well before the maximum.

window

Length of the analysis and resynthesis buffers (eg. 256 to 4096). Longer windows provide better frequency resolution but less temporal resolution during resynthesis. I find 2048 offers the best balance, with 4096 occasionally for slowly evolving timbres. The resynthesis step of phase vocoders introduces more distortion than normal IFFT resynthesis due to the time step difference between windows, shorter window lengths distort lower frequencies. Values lower than 2048 are left in for creative purposes.

practical information - analysis

Operations requiring file analysis or reading take time. Loading wave files can take several seconds and will occur when changing patches. If you are not using cross synthesis, make sure to remove any filename.

technical description

The phase vocoder and cepstral process are classics of audio dsp and we already have an idea of what they do and how we want to use them. I will try to convey how these processes are used here for the best results.

Phase vocoders analyse the phase increment between successive fft analysis frames to estimate the pitch of each frequency band, or bin. Less distance between frames improves the accuracy of analysis but increases data. Bin aliasing, or sidebands, occur when frequencies are between central frequencies of the analysis window bins.

The other fundamental issue with the phase vocoder is that windowing resynthesis frames modulates the signal, producing sidebands equally as loud as the signal. 4x overlapping cancels them to a level canonically considered to be tolerable, I think around -24 or -30dB.

Classic phase vocoders are known for sounding reverby. A number of schemes attempt to improve this impression in various ways.

Frequency resolution of the fourier transform is determined by samplerate divided by window length (eg. 44100/2048 = 21.533 Hz).

The classic phase vocoder performs time stretching by modifying the phase information. Pitch shifting is then achieved by changing the playback speed of the resynthesised file. The computational expense depends on the pitch/speed of playback. Other schemes achieve a fixed cost by shifting the bin data. This method achieves and retains good quality with extreme modification.

If the only application is pitch or time stretching, turning the spectral process off will save cpu.

Spectral or "formant" shifting is achieved with cepstral separation of source and filter. The cepstrum is derived by taking a spectrum of a spectrum's magnitude bins (after they are log10'ed). It sort of separates "slow moving parts of the spectrum" or resonances and places them in the lower bins. Dividing the cepstrum into a lower and upper part produces a "filter" equivalent to the spectral contour, and a "source". These two elements can be modified and recombined by the principle of spectral convolution. Ideal liftering of a pitched signal locates the sharp transition where the resynthesis is affected by the pitched component, and setting the cutoff below it. Precise separation of the source and filter will have better results for the source resynthesis.

The best way to understand this process is to observe it with a spectral analyser. Using "sp.shift off" allows you to observe the original magnitude/spectral contour, then switch to another function to see how closely the reconstructed contour fits and sounds. You will see, even low liftering achieves a strong match for the contour.

Similarly, the resynthesis procedure can be appreciated by observing how increasing overlaps picks off sidebands. There may be no benefit to using higher than 5x overlapping with some files. Freeze playback to one frame to observe how partials react to parameterisation.

Resyn was updated for higher analysis specification. This platform will allow you to generate over 100mb of data from a 10k rimshot wav, and use 80% of your cpu resynthesising it, or 8%, depending on your quality requirement. Stereo cross resynthesis with 16x overlapping is all yours if you want to use it.

The primary variable is how many frames to use. Resyn2 uses 512 frames for stereo analysis and can use all of them (1024) for a mono file, for over 8 seconds of high quality analysis. The center of the frames dial is set at 64 and steps through powers of two. A high framecount is used for smooth resynthesis. Very low framecounts may sound more periodic due to linear interpolation but have creative uses. There is no readout because I am attached to the awesome looking gui.

When the free range button is inactive, hop size (distance between analysis frames) is limited to windowlength/5, which gives reasonable reconstruction of frequencies. Maximising audio quality requires using the parameters to select only as much of the file to analyse as you will use. Push the analysis and examine the best quality results this method can achieve. When hop size is too large, resynthesized tones will have inconstant amplitude, and sound warbly or pulsy.

Filters adapted from source by Robin Schmidt (SV forms) and Neotec.

also

Cross resynthesis can get loud. Perhaps use a limiter.

SynthEdit knobs have a "standard increment" of 100 places, eg. a knob from 0 to 10 will step through 0.1, 0.2 when dragged.

Fine tuning of parameters is achieved by holding [ctrl] while dragging.

You may wish to reset some parameters (eg. shift) to an exact central value. I don't have an easy way to implement this. The best way is to drag the knob to the end of the range to zero out any "fine tuning", then try and drop it back on the center. It will sound right if it is right. The shift knobs have an 8 octave range..