Creating Aesthetic Soundscapes with the use of Ambient Sounds

By Dinis Tavares

Table of Contents

Introduction 3

Aesthetics of sounds 3

The Patch 3

Recording, Looping and Time Stretching 4

Track Controls 4

- **Time flux** 4
- **Effects** 5
- **Filters and Recursive Delay** 5
- **Arpeggiator** 6

Mixing Ambient Sounds 6

Introduction

The idea for the patch called 'Ambien- cape' was influenced from an interview with John Cage, an American composer and Music Theorist. Cage thinks of music as somebody talking about their feelings and ideas of relationships, where he perceives every day sounds such as traffic, as if the sound were acting (jdavidm, 2007). Describing his satisfaction with these "acting" sounds without the need for them to "speak" to him, he mentions three key characteristics of ambient sounds; where they get louder and softer, higher and lower and longer and shorter, which eventually inspired the controls to manipulate ambient sounds in the patch. The thought of ambient sounds which do not 'talk' raised the Question; "How can these 'acting' sounds be used to 'talk' to the listener? and If they cannot 'talk', how can the listener perceive these sounds 'acting' in a new and different way?". An interpretation of having sounds such as music 'talk', could mean the sounds have aesthetic appeal to the listener, whereas sounds which 'act' could be defined as sounds that have no aesthetic influence on the listener. Resulting in a research question of "Is it possible to create aesthetic soundscapes with the use of everyday ambient sounds?".

Aesthetics of sounds

In order to determine if the possibility to turn non-aesthetic sounds into pleasing soundscapes, it will be valuable to briefly understand why people have different aesthetics. The idea of aesthetics being culture-bound or universal may affect whether or not a soundscape is considered as aesthetically pleasing to a listener, however it is thought that different cultures have their own aesthetics by an author of Anthropology and Aesthetics, Jeremy Coote (1993). Therefore, the patch will primarily aim towards users who have the desire to create a western aesthetic soundscape. A report evaluating a field study undertaken in Boston, Massachusetts, found that participants disliked the sound of loud vehicle sounds but had a positive evaluation of idling cars in the distance (Anderson et al. 1983), a variable which could suggest implementing the ability to create spacious and fluctuating effects into the patch. The report further acknowledges another study taken in 1980, where a sample of mountaineers and a sample of campers in a developed campground, who were more accepting to mechanical and unnatural sounds, where asked to rate sounds presented to them from 'pleasing' to 'annoying'. The evaluations showed that both groups found sounds of wind, water, fire, birds and insects to be pleasing sounds, whereas sounds of chainsaws, cars, trail bikes and other mechanical sounds to be annoying or unpleasant. In addition, a similar study was conducted on the sonic environment of cites, which showed that people preferred intense low to middle frequency sounds of the city and were delighted to hear sound which were novel, informative and responsive to personal actions (Southworth, 1969). These studies provide an idea of which sounds are perceived as 'pleasing' to western cultured people, resulting in creating a patch with effects that can transform and closely emulate 'pleasing' sounds with the use of various unpleasant sounds.

The Patch

The patch was created in Max 7 by Cycle'74, a visual programming language for music and multimedia. In-order to keep the patch as neat as possible, sub-patches with visible user interfaces called 'batchers' were embedded within each other. The first bpatcher created is called 'Buffers_Main' (Figure 1) and is used as individual tracks in the final patch.

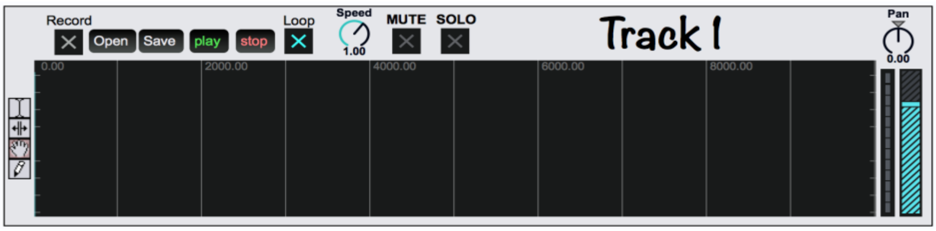


Figure 1, The buffer patch. Used to record, loop and manipulating playback time. See Appendix A for the build of the patch.

Recording, Looping and Time Stretching

The first step for the project required the ability to record live ambient sounds, which is done by using a buffer to temporality store the audio information. The 'buffer~' object requires a name and allows the options to specify the duration and channels, where each buffer of the patch has been set a duration of 10 000 milliseconds and stores the audio in mono. A 'record~' object with the same name as the buffer, is used to derive information from the default microphone of the hardware, using an audio to digital converter or an 'ezadc~' object and a toggle button to start and stop the recording. In addition, two messages called 'read' and 'write' are sent to the file operations of the buffer, allowing the user to save any audio recorded as well as load any previously recorded sounds.

The 'groove~' object allows the patch to incorporate some of the controls initially inspired by John Cage, where sounds get faster and slower as well as higher and lower. The object is a variable-rate, loop and sample playback object which uses the same name as the buffer to refer to the audio information. A signal is sent to control the sample playback where a value of 1 will play the audio information at its initial speed and pitch which can be altered by simply changing the value of the signal. A decimal value below 1 will stretch the audio file, slowing down the playback time as well as lowering the pitch, using ideas from the key characteristics of 'acting' sound to control and manipulate the recorded sound files and creating a considerably novel and interesting sound for a soundscape. Moreover, using a lower signal to control the playback of an audio recording can create large spacious sounds. The loop feature of the 'groove~' object first needs to be activated by sending a "loop $1" message to the object, where the beginning and end of the loop are selected by sending values (in milliseconds) to the object. In order to make the patch more user friendly, the buffer is set to a 'waveform~' object, where the user can highlight and select the section of the audio file they would like to loop through, sending the beginning and end values to the 'groove~' object. Finally, the loop can be stopped by switching off the loop toggle, which allows the rest of the recording to play through and stops once it has reached the end. Alternatively, a 'stop' message can be sent, stopping the audio completely. The signal from 'groove~' object is then sent to an outlet of the first 'bpatcher', which will be sent to an inlet of the 'Controls_Main' patch (Figure 2).

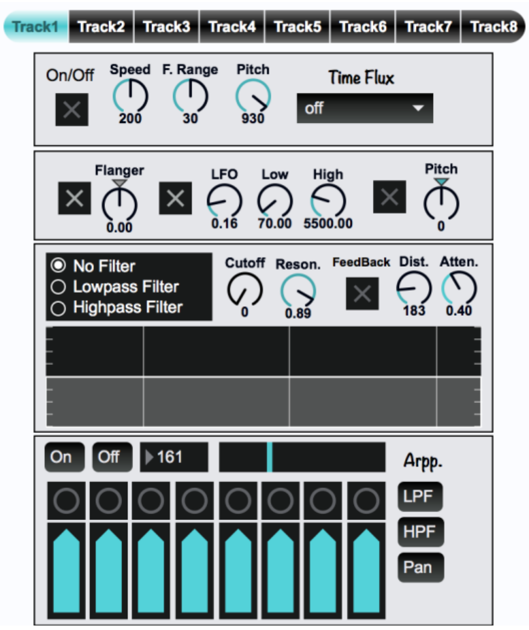


Figure 2 Controls patch, used to manipulate the recorded sounds in various ways. See Appendix 2 for patch build.

Track Controls

Time flux

The Controls patch further contains four embedded 'bpatchers' which are used to control and manipulate the recorded sounds. The first embedded patch is a time manipulating effect (Figure 3) and is used to automatically fluctuate the speed of the sample playback in the 'groove~' object. The sub-patch has 5 controls, an on/off switch, a speed control, fluctuation range, pitch control and playback dropdown menu. When the patch is switch on, a metronome is triggered and a counter is started, where the speed

Figure 3 Controls to manipulate the sample playback of the recorded sounds. See Appendix 3 for patch build.

of the counter is controlled by the speed of the metronome. Therefore, the first dial on the sub-patch is used to control the speed of the metronome and counter. A second dial is used to control the fluctuation range, where the value of the dial will set the maximum value of the counter and a 'select' object. The 'select' object is used to control whether the value of the counter will be increasing or decreasing, and once the value reaches the maximum or minimum value it will alternate. The third dial is used to control the value of the track speed and pitch, which is done by dividing the counter value by the 'Pitch' dial value. In addition, the pitch dial has been inverted to have the lower value at a lower pitch and speed and the higher value at a higher pitch and speed. The sub-patch further builds on using the idea of speed control in-order to create an intriguing ambient soundscape.

Effects

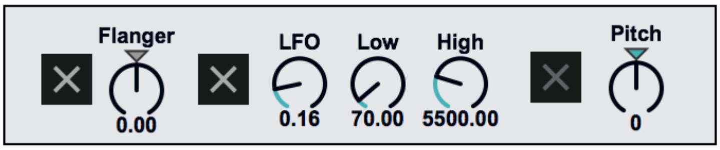


Figure 4 The Effects patch containing a flanger, LFO and pitch scaling effect. See Appendix 4 and 5 for patch builds.

The second sub-patch embedded into the Controls patch contains a flanger, a low frequency oscillator (LFO) and a pitch scaling effect (Figure 4). The flanger creates a comb filter by using a delay applied to the entire signal (Howell, 2006) and is done by using a short delay which is modulated by an LFO. This allows the user to further experiment with recorded ambient sounds to create an aesthetic soundscape. The second attribute of the effects patch is an LFO, which cycles through the frequencies of the signal using a 'reason~' object. The speed of the LFO is controlled by the dial labelled 'LFO', where the frequencies which the LFO cycles through are controlled by the dials 'Low', setting the minimum frequency and 'High' setting the maximum frequency. The final effect is a pitch shifter, used to pitch up or pitch down sounds without affecting the playback speed of the sound file and working similar to the Doppler effect. In order to create this effect, a 'tapin~' 'tapout~' delay and a phasor are used to pull the sound apart and decrease the frequency by increasing the amount the sounds are delayed overtime with a positive phasor value (dude837, 2011). In contrary, when the phasor value is negative, the delay of the sound is being decreased overtime resulting in the soundwaves being compressed together and outputting a higher pitch. Additionally, the phasor produces clicks and therefore the signal is processed, using a cosine ramp to silence the output of the sounds when approaching the clicks. Finally, the process is multiplied twice, where the signal will be shifted one third out of phase from the phasor and the second process will be shifted two thirds out of phase, to remove the phasing effect from the sound and producing only a pitch shifted sound.

Filters and Recursive Delay

Moving on to the third sub-patch of the controls, a high-pass filter (HPF), a low-pass filter (LPF) and a recursive delay are added (Figure 5). The high-pass and low-pass filters are both controlled by the 'filtergraph~' object or the 'Cutoff' dial and a 'radiogroup' object is used to select whether the signal is sent unfiltered, through the HPF or LPF, as well as change the display of the filter graph in respect to which filter is selected. The low-pass filter uses a 'lores~' object allowing two arguments, the first to set the cut off frequency and the second to set the resonance, controlling the sharpness of the filter (Cycle '74, n.d). Similarly, the high-pass filter uses the 'lores~' object, however instead, the lores signal is subtracted from the original signal to create a HPF. As high frequency content is one

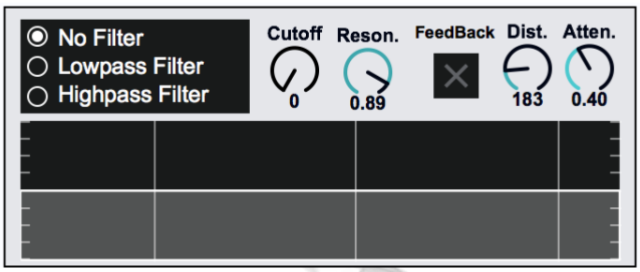


Figure 5 High-pass and low-pass filters, with a recursive delay. See Appendix 6 and 7 for patch builds.

of the most important cues to the human auditory system when measuring proximity (Schlette, 2013), a low-pass filter can be used to create the effect that one signal is further away than another unfiltered signal, creating spatial contrast between the two signals. The following effect is a recursive delay, using the 'tapin~' 'tapout~' objects which continuously feed the delayed signals through itself and is multiplied by decimals between 1 and 0 beforehand to attenuate the signal, and stop it from aggregating. The delay with a maximum of 1000 milliseconds can be controlled by the 'Dist.' dial, allowing a higher or lower delay time and the attenuation can be controlled using the 'Atten.' Dial.

Arpeggiator

The final sub-patch of the controls is an arpeggiator (Figure 6), which allows the option to control the HPF and LPF cutoff frequency as well as the panning attributes of each track. Three buttons allow the user to select what the arpeggiator will control, whilst also changing the values of the sliders respectively, which in turn change the values of the controlled attribute. However, the attribute value will only be changed once the 'on' button is clicked, starting a metronome and a counter to cycle through slider values. The arpeggiator further allows the user to be creative with the panning and filtering of each recording when creating soundscapes.

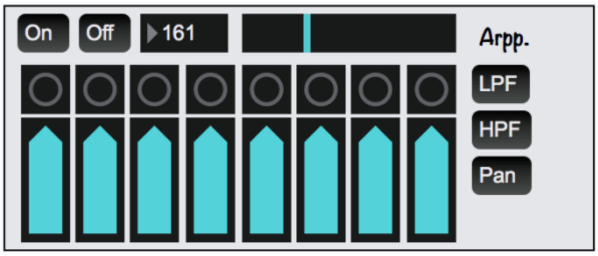


Figure 6 An arpeggiator used to control the HPF, LPF and panning of the track. See Appendix 8 for patch build.

Mixing Ambient Sounds

The main patch allows the user to record or load 8 different ambient sounds to use when creating the soundscape, where each track is provided with individual control panels (Figure 7). The global controls are simple and straight forward, where the user can stop all tracks,

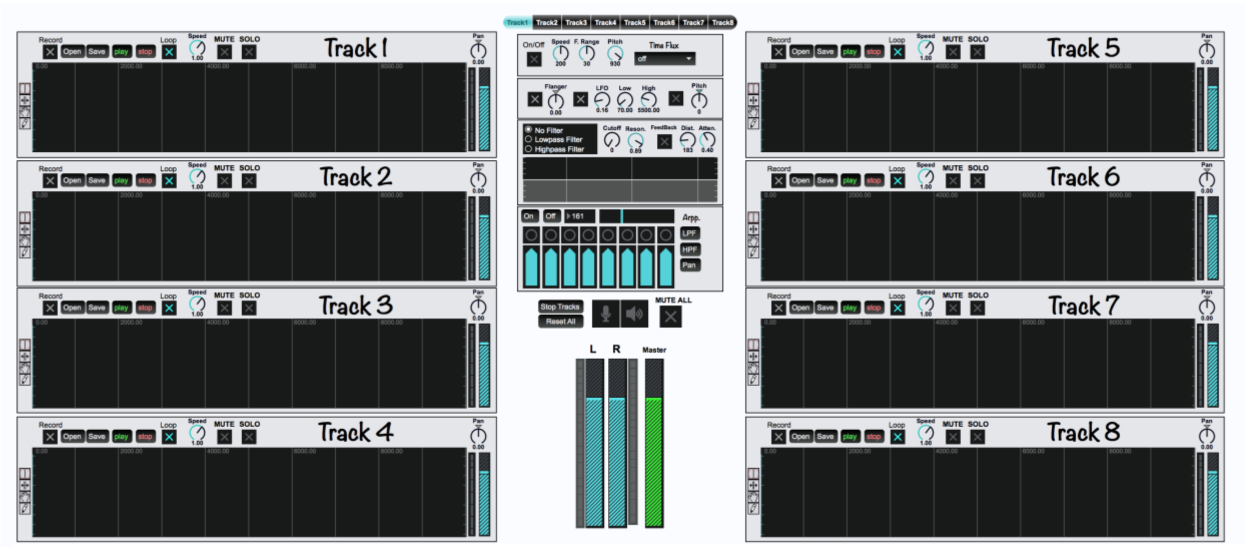
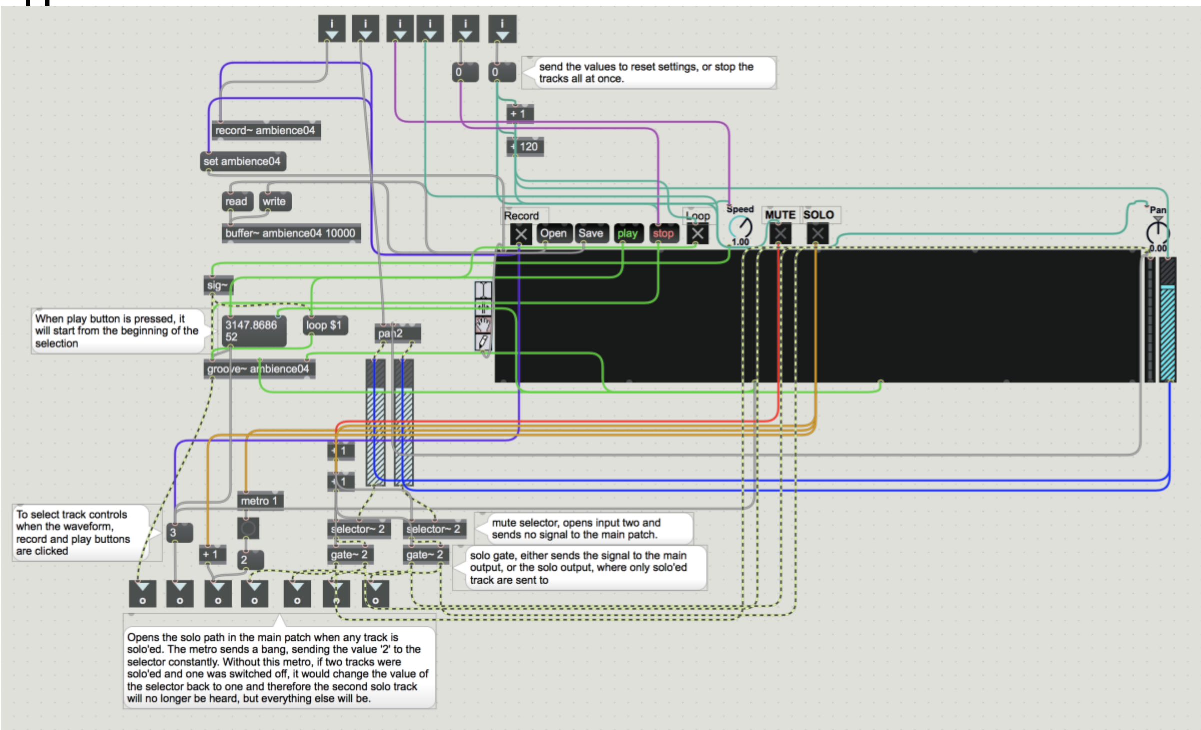


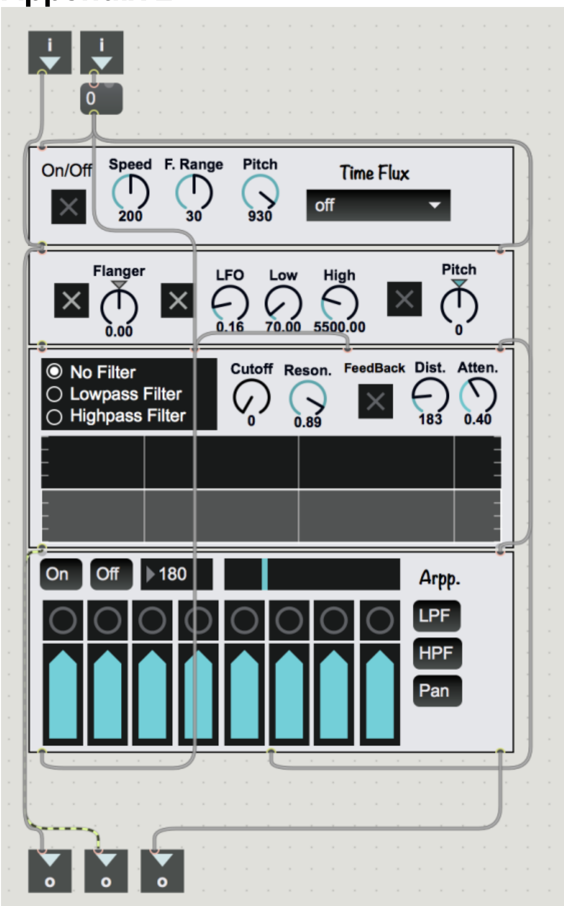
Figure 7 The main patch user interface.

mute all tracks and reset all of the settings. Additionally, the user can use the left, right or main gain slider to control the volume of all tracks. Ultimately, the patch can be used to effectively create aesthetic soundscapes to the western ear, which can manipulate sounds such as traffic, busy coffee shops, loud planes, trains and many more 'acting' sounds into novel and interesting sounds. Further aspirations for the patch include applying noise with effects, to emulate natural sounds such as waterfalls and rainfall, as well as including a multi-band EQ rather than only using pass filters. Another desire for the patch would be including a vocal control, where the user can use their voice to control effects of the patch, creating direct responses to personal actions.

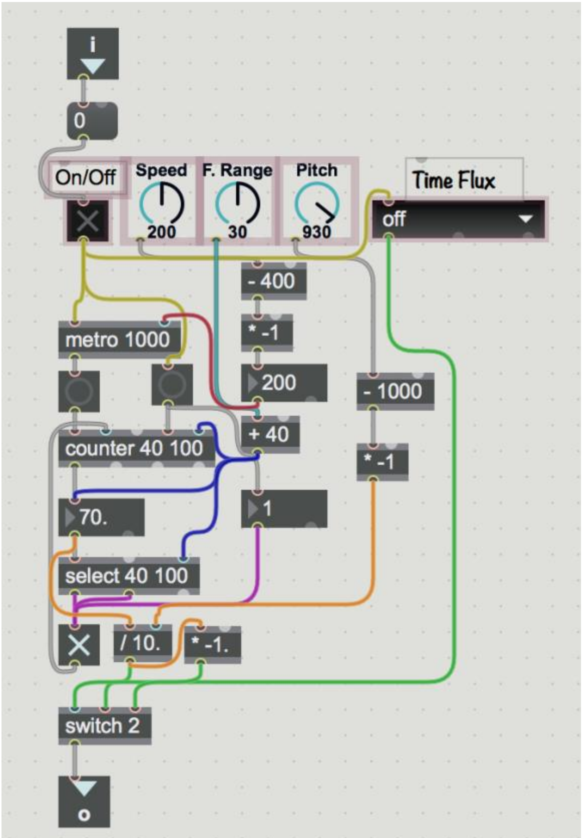
Appendix 1



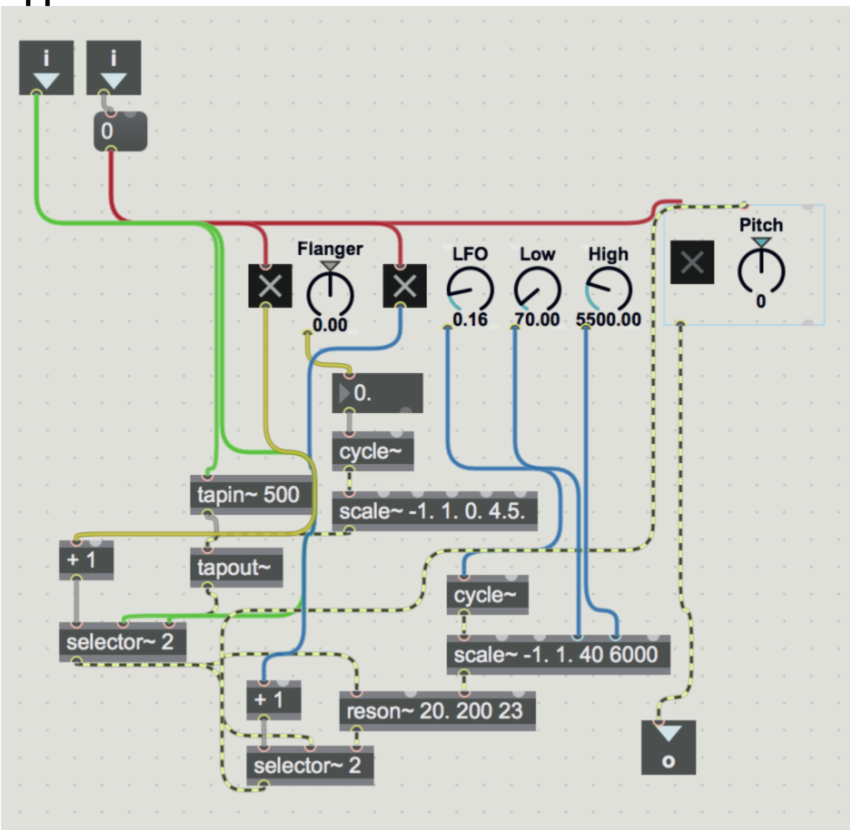
Appendix 2



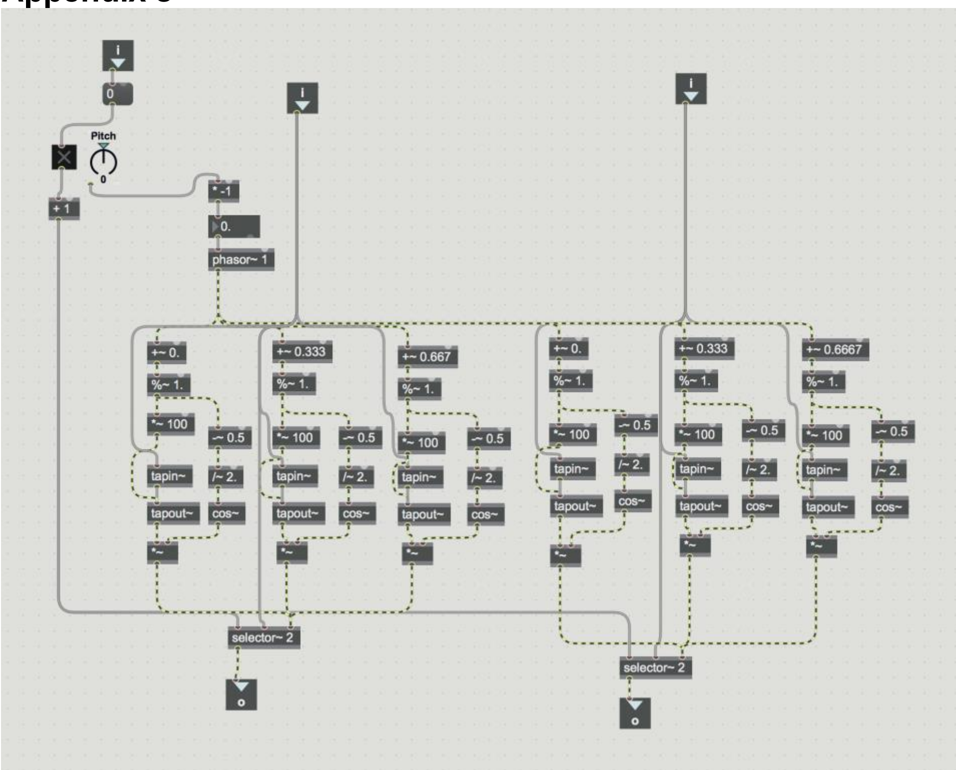
Appendix 3



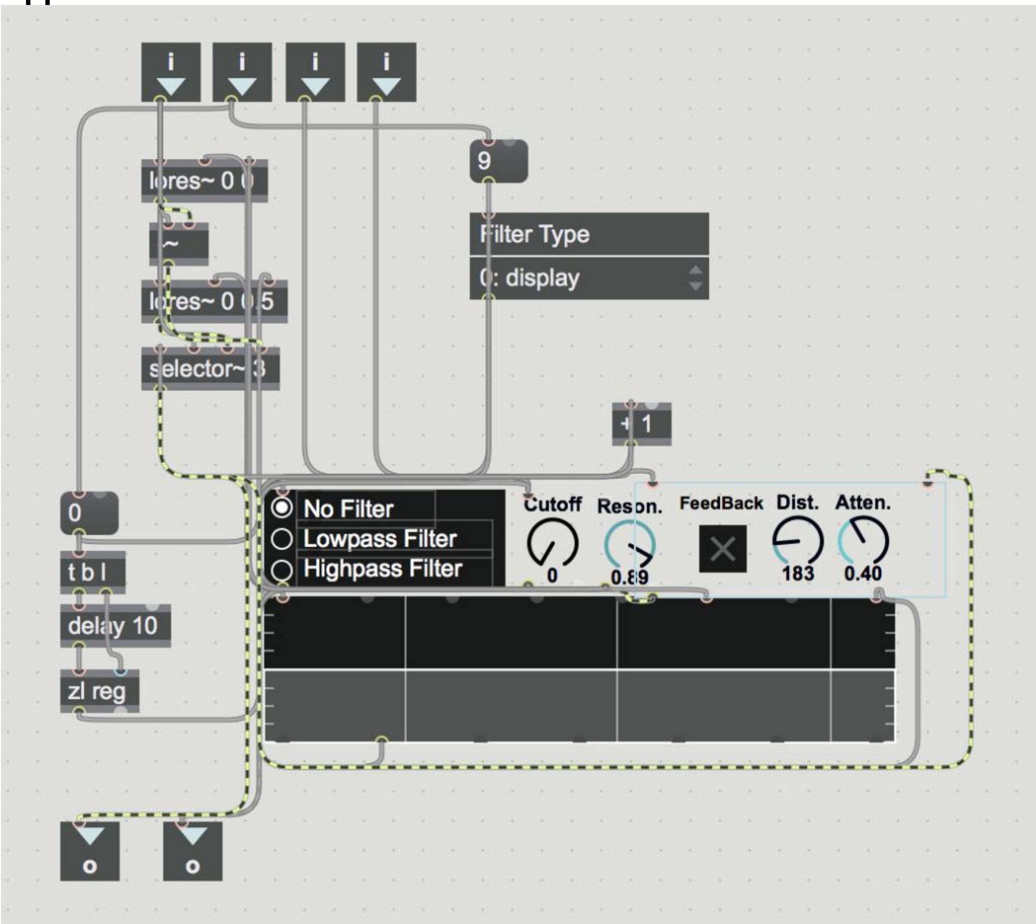
Appendix 4



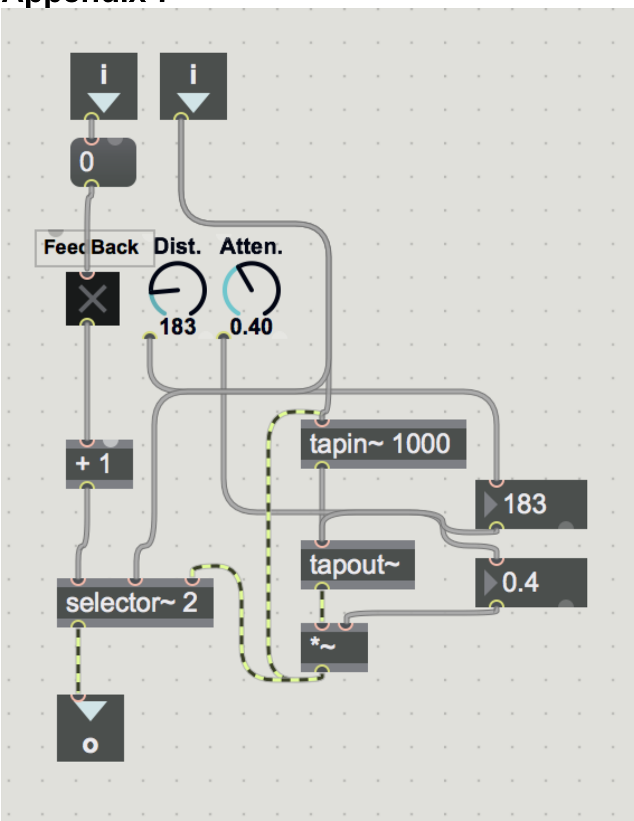
Appendix 5



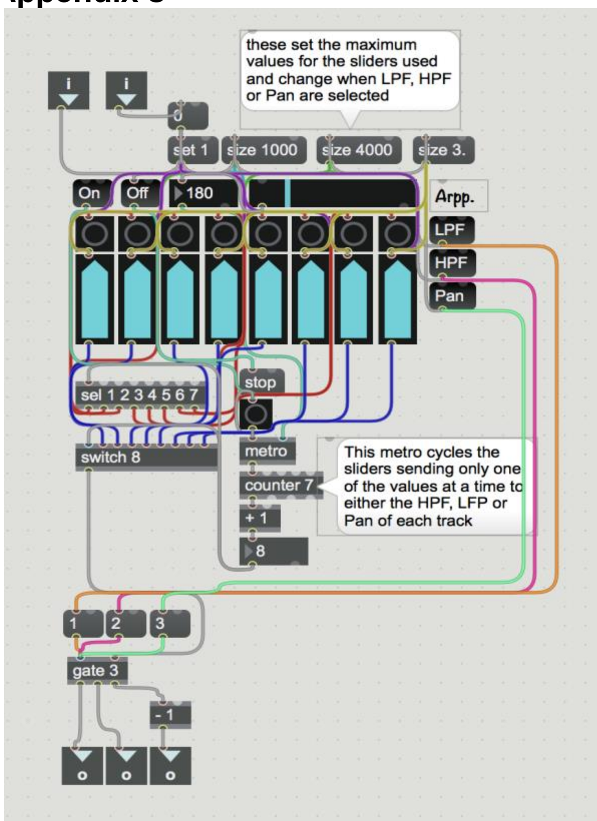
Appendix 6



Appendix 7



Appendix 8



References

- Anderson, L. M., Mulligan, B. E., Goodman, L. S., Regen, H. Z., 1983. *Effects of Sounds on Preferences for Outdoor Settings.* Environment and Behaviour. Vol 15, Issue 5, pp. 539 - 566
- Coote, J. 1992. *Marvels of Everyday Vision: The Anthropology of Aesthetics and the Cattle-Keeping Nilotes.* In J. Coote & A. Shelton Eds., Anthropology, Art, and Aesthetics. Oxford: Clarendon Press.
- Cycle ’74, n.d., *lores~ Resonant lowpass filter.* [online] Available at: <https://docs.cycling74.com/max5/refpages/msp-ref/lores~.html> [Accessed on 8 January 2018].
- Cycle ’74, 2011. *Did You Know #5: High Pass Filters.* [video online] Available at: <https://youtu.be/W7iVrC-zDWo> [Accessed 4 January 2018].
- dude837, 2011. *Delicious Max/MSP Tutorial 7: Pitch shifting*. [video online] Available at: <https://www.youtube.com/watch?v=uyzY_ZP54pA&t=456s> [Accessed on 6 January 2018].
- jdavidm, 2007*. John Cage about silence.* [video online] Available at: <https://www.youtube.com/watch?v=pcHnL7aS64Y> [Accessed 4 December 2017].
- Schlette, R. 2013. *The Complete Guide to Audio Pass Filters.* Pro Audio Files, [online] Available at: <https://theproaudiofiles.com/audio-pass-filters/> [Accessed 6 January 2018].
- Southworth, M., 1969. *The Sonic Environment of Cities.* Environment and Behaviour. Vol 1, Issue 1, pp. 49 – 70.
- Howell, S., 2006. *What's the difference between phasing and flanging?.* Sound on Sound, [online] Available at: <https://www.soundonsound.com/sound-advice/q-whats-difference-between-phasing-and-flanging> [Accessed on 6 January 2018].

Bibliography

- MUST1002, 2017. *Max 7 Tutorials: 8.1c Effects: Delay—recursive (feedback) delay* [video online] Available at: <https://www.youtube.com/watch?v=rt13YUkBoJc&feature=youtu.be> [Accessed 5 December 2017].
- MUST1002, 2017. *Max 7 Tutorials: 8.1a Effects: Delay—introducing [tapin~] and [tapout~].* [video online] Available at: <https://www.youtube.com/watch?v=hOJ8o7g8ARc&feature=youtu.be> [Accessed 5 December 2017].
- MUST1002, 2017. *Max 7 Tutorials: 8.1b Effects: Delay—multitap delay.* [video online] Available at: <https://www.youtube.com/watch?v=17QBxJXhTQk&feature=youtu.be> [Accessed 5 December 2017].
- MUST1002, 2017. *Max 7 Tutorials: 8.1c Effects: Delay—recursive (feedback) delay.* [video online] Available at: <https://www.youtube.com/watch?v=rt13YUkBoJc&feature=youtu.be> [Accessed 5 December 2017].
- MUST1002, 2017. *Max 7 Tutorials: 8.2 Effects: Comb filtering & Flange.* [video online] Available at: <https://www.youtube.com/watch?v=O8wz7FvKvcc&t=182s> [Accessed 5 December 2017].
- MUST1002, 2017. *Max 7 Tutorials: 8.3 Effects: Resonant filters ([reson~] and [fffb~].* [video online] Available at: <https://www.youtube.com/watch?v=HEpai_5E0tg&t=233s> [Accessed 5 December 2017].
- MUST1002, 2010. *Max MSP Tutorial 04b7: Playback using [buffer~] & [groove~] [1].* [video online] Available at: <https://www.youtube.com/watch?v=cy9yY5V6Y6s> [Accessed 5 December 2017].
- MUST1002, 2010. *Max MSP Tutorial 04b7: Playback using [buffer~] & [groove~] [2].* [video online] Available at: <https://www.youtube.com/watch?v=zuFH8c5DfMg> [Accessed 5 December 2017].
- MUST1002, 2010. *Max MSP Tutorial 04b7: Playback using [buffer~] & [groove~] [3].* [video online] Available at: <https://www.youtube.com/watch?v=8d_8pLyvydc> [Accessed 5 December 2017].