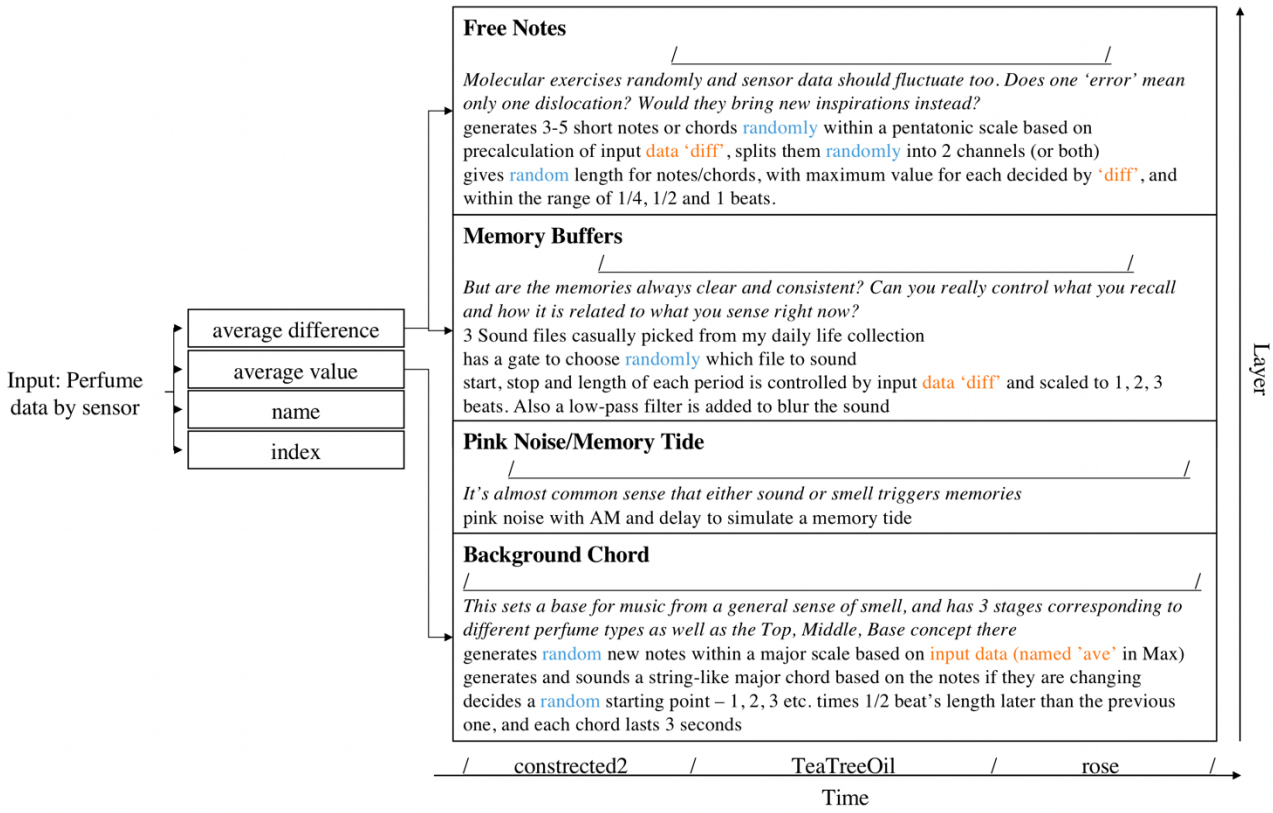
Creative Coding for Sound Assignment 1
Author: Yining Xie

Synaesthesia

This work involves both sonification of scent and random music generation. This idea comes from a not-rarely noticed phenomenon, synaesthesia, while less mysterious and in view of sound and smell, there have been studies and practices on the crossmodal effect of either the influence of music on olfactory perception[1] or the use of smell in music performance[2]. Another critical element involved and more frequently talked about is the memories they bring about. Therefore, what I would like to explore here are:

Contrary to the more common knowledge of 'smell sound'[3],can scent be sounded alternatively? Can it be represented initiatively and imaginatively if there is no such natural synaesthesia? If so, what would it be like? I need much correspondence between both for this.
People have been good at making and controlling music, but is smell controllable too? Can we really predict how it would evolve precisely, and if not, what shall we expect?
Molecular exercises randomly and interacts actively with the environment. Hence what would a single disturbance result in, is it single or a series? Would that bring disaster or inspiration?
I am reminded of the Proustian Effect. Are the memories brought back always clear and consistent? Can you really control what you recall and how it is related to you sense right now?
How does time play its role here? Would it be disturbed by either smell or sound too? Do we really feel the same each time smelling the same scent or hearing the same sound?

I think I have my own answer while exploring them (though partly because of the too-small database too) directs me to consider all the following aspects with the diagram of structure:



Correspondence
all notes generated are based on or guided by the real smell data. The 'free notes' part, with a shorter duration and greater density, representing any inconsistency or disturbance, holds a rather complicated relationship with the data, and is only triggered when it sees a change. The 'background chord' part, instead, works directly around the value of data we have, generating longer notes no more than 12 semitones away from the base one, which in this 3 minutes has only three values that are mapped to C4, G3, G5. Then it makes major chords, just how 'average data' feels should sound. Horizontally the 3 fundamental notes (and eventually chords) also correspond to the 'top, middle, base' concept in scents.
Memory buffers are cut to different parts and lengths too as is the case in real experience.

Generation
new notes/chords are generated in both 'background chord' and 'free notes' parts, respectively in the major scale and pentatonic scale, to put attention to their different nature but also their ability to self-develop. Generation for chords carries the richness of smell and that for free notes gives the answer to disturbance – each brings a series of changes.

Randomness
as above, the new notes are generated randomly (though within a range). The 'presets' patch even makes it possible to choose between a single note or chord randomly. Starting points for background chords, the gate to open for memory files, length for free notes, and which channel to sound are all random, corresponding to the uncontrollability of scents.

Time
In this project, I do not stick to a strict rhythm, while still, all lengths/start/end points involved are multiples to the ¼ beats – hence they shall be able to be written on a score anyway. The tempo is set to be 60 as I would like it to develop slowly. The one with particular consideration is the 'free note part', within the range of sixteenth, eighth and quarter notes in 4/4 as I'd like it to be relevantly more musical. As also mentioned above, the length of memory buffers is decided by real data which also works with midi values. Those for free notes and the starting points for each background chord are picked randomly. They together forms a mysterious relationship and interaction between time, perception, smell and sound.
The length for each background chord though, is set to the same long value for the stability aspect of both smell and sound.
One other thing to notice is free notes only start to generate when input data is changed while buffers start when input is unchanged. Also duration of sounding buffers are always shorter than their original length, meaning memories are cut, inconsistent, (and also unclear with low pass added).

Instrumentation
I would love to make the chords sound like string (violin etc) and free notes like piano or music box, but have not yet found a satisfactory way to include their vsts together (and too lazy to output to DAW). In this assignment, they are simulated by triangle oscillator and sine oscillator with amplitude modulation.

Noise
added pink noise with AM and delay to simulate a memory tide.

Besides the above, the 4 layers also follow an order to come and go as shown in the graph. The randomness also means that each time the patch is played, it sounds different. I designed it to run the

whole process by itself with AM while they can also be replaced by buttons for user to control (or they can even just put new data in to make completely different music).

Data is pre-processed – added average and calculated difference from average, and did some observation to multiply 10 for the 2nd part. Both processed and original data are submitted. The only reference of code (except read data from Learn) is how to output the opposite of 'change'[4].

Reference:
1 Velasco C, Balboa D, Marmolejo-Ramos F and Spence C (2014) Crossmodal effect of music and odor pleasantness on olfactory quality perception. Front. Psychol. 5:1352. doi: 10.3389/fpsyg.2014.01352
2 Scent of Memory: smell and classical music prove an intoxicating combination, Clarissa Sebag-Montefiore
3 Smelling sounds: olfactory–auditory sensory convergence in the olfactory tubercle. J. Neurosci. 30, 3013–3021. doi:
4 Code 'opposite of change': https://cycling74.com/forums/opposite-of-change
5 Original data source: https://archive.ics.uci.edu/ml/datasets/Perfume+Data#