Hand gesture to timbre converter

Miller Puckette
UCSD
msp@ucsd.edu

Kerry L. Hagan
University of Limerick
Kerry.Hagan@ul.ie

reprinted from Proceedings, ICMC 2020

ABSTRACT

In the design of a novel computer music instrument that uses the Leap Motion game controller to generate and play computer-generated sounds in real time, we consider the specific affordances that hand shapes bring to the control of time-varying timbres. The resulting instrument was used in a new piece, Who Was That Timbre I Saw You With?*, performed as a live duet. Central to our approach are: a geometric exploration of the shape of hands themselves; a consideration of accuracy and speed of limb motion; and an appropriately designed sound generator and control parameter space.*

1. INTRODUCTION

In this paper we describe a particular choice of hardware interface, parameter extraction technique, synthesis algorithm, and parameter mapping strategy we arrived at to realize a live electronic piece, named *Who Was That Timbre I Saw You With?*, that we perform as a duo. The exigencies of the production led us to develop some novel techniques. Our purpose in writing this is not to make any artistic claim, but rather to motivate and describe the techniques we developed in order to realize it.

Our starting point is an observation attributed to Max Mathews: the highest bit rates that humans can transmit, whatever the receiver might be, are emitted through two channels: the articulators of the vocal tract, and the fingertips. This claim is admittedly imprecise because the exact meaning of "bit rate" in this situation is hard to pin down. Nonetheless, we might intuitively guess that a good computer/human interface for real-time control over a computer music instrument should start at one or the other of these sites.

In our musical experiments over time, we have tried one and another approach to live electronic music performance, each adapted to a specific musical idea, and each taking advantage of the affordances of human output in one way or another. In the project described here we consider the idea of mapping the shape of the human hand to timbre (considered as a collection of entangled parameters). This has led us to a particular computer music instrument that we use in the piece.

In what follows we will consider, first, the affordances of human control and the requirements for controlling various aspects of musical sounds (section 2), the adaptation of a specific controller to our task (section 3), and the design of a suitable synthesis algorithm (4).

2. AFFORDANCES OF HAND GESTURES FOR CONTROLLING SOUND

In order to plan how to map hand and finger movement into sound we will first need a model of the space of sounds we are interested in. As a point of departure we adopt a crude model of sound as being a succession in time of instantaneous snapshots, each of which is characterized by zero, one, or more pitches and an overall spectral envelope that determines instantaneous loudness and timbre. This spectral envelope can be crudely described as an array of numbers giving the instantaneous signal power in each of 24 critical bands.

The 24 "timbre" numbers (positive real-valued functions of time) have bandwidths ranging from 150 Hz to a few kHz, whereas pitch, while it defies numerical characterization, probably typically changes much more slowly. On the other hand, pitch can be heard with much higher numerical accuracy than can loudness; we can hear on the order of a thousand distinct pitches but only between 50 and 100 gradations of loudness, and probably fewer than 50 of changes in power within a specific critical band in most circumstances. (These are just brute characterizations; actual numerical values would depend strongly on the choice of stimuli and listening environment.)

Meanwhile, the accuracy with which one can place a part of a finger, and the speed with which one can make changes, also depends strongly on the situation. If, for example, we wish to place a fingertip at a specific location in space, assuming we have access to some sort of feedback, we might have a range of a meter and an accuracy of a few millimeters, giving approximately a part in three hundred. This is roughly enough to control a musical pitch down to 3 cents over a range of an octave, so it is possible to consider this as a reasonable source of pitch control, as with a theremin.

The drawback here is that to get to a given location one must move an entire arm. This is not only slow (when compared to the speed at which a pianist or violinist can make pitch changes) but also makes for a distracting visual spectacle.

If on the other hand we restrict our range of motion to the motion of a fingertip (with a stationary palm) we lose perhaps a factor of ten in range (and perhaps improve the accuracy of placement by a factor of two or three, so the overall range-to-uncertainty only decreases by a factor somewhere between two and five, say), but in compensation we get a

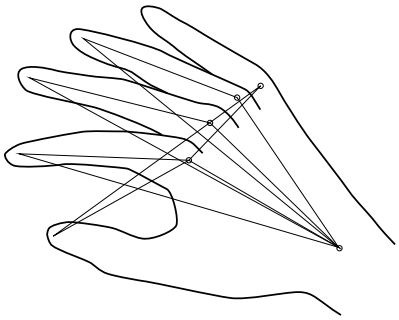

Figure 1. Characterizing hand shape as spanning four triangles, one through two knuckles and the tip of the thumb, and three others through wrist, knuckle, and fingertip.

big increase in speed, perhaps comparable to the loss in resolution. The positions of parts of hands (joints; fingertips) measured relative to each other have many degrees of freedom collectively—the 19 joints each offer one or two degrees of freedom. This, combined with the fact that resolutions are roughly compatible, suggests that hand shape might be well suited as a control over loudness and timbre.

It would be a fallacy to suggest that one could simply map joint angles directly to bark spectrum amplitudes, for two reasons. First, this would raise the very difficult question of what type of sound is to be shaped in this way (FM? granular synthesis?) and, *a fortiori*, what further control space must then be opened for input from somewhere else. Second, any coordinatization of hand shape would have many dependencies between the coordinates, so that the great majority of tuples would be unreachable, and some combinations would be much more quickly and accurately accessible than others. Just to start with, we note that the lengths of the 19 bones in play do not normally change during a musical performance.

Another limitation is speed. While the hand can move quickly compared to other parts of the body, its motions are at least an order of magnitude too slow to capture the fastest perceptible changes in loudness or timbre. Many physical instruments are able to make much faster changes in timbre than the human player is required to move, for instance by a gathering of energy for quick release (a plosive attack for a wind instrument, or an impact of two physical objects that have previously been put in motion.) It is hard to see how to make similar quick changes controllably in the absence of an opposing physical object.

These considerations lead to soft guides (not hard constraints) on what aspects of audio production we might want to tie the measured quantities to.

3. MEASURING HAND SHAPE

Figure 1 shows an idealization of the geometry of a hand in space, reduced to the shapes of four triangles. Three of them span the wrist, a knuckle (of the second, third, or fourth finger), and the corresponding fingertip. These capture most of the possible motion of those three fingers. The fifth finger is not included since it and the fourth finger are hard to move independently.

The fourth triangle similarly tries to capture the motion

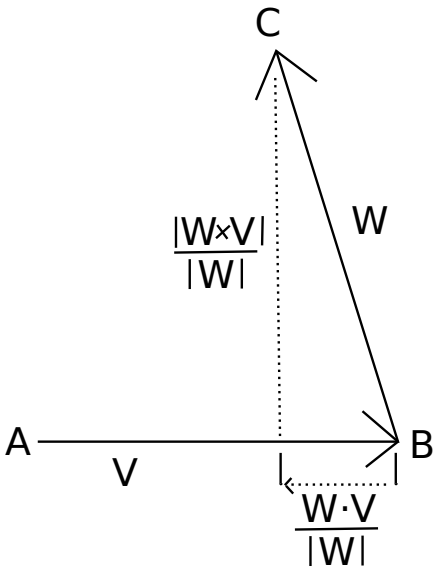

Figure 2. Characterizing a triangle whose base has known length.

of the thumb. Here we use as a frame of reference the segment between the first knuckles of the second and fifth finger.

We consider only the intrinsic shapes of these four triangles, and not their positions or orientations in space. Although in general a triangle requires three parameters to describe it (base length and horizontal and vertical position of apex), the four triangles we consider only have two degrees of freedom since one side (which we take to be the base) has fixed, or only slightly variable, length. The three finger triangles have as one segment the length of the bone from wrist to knuckle, which in healthy adults does not change over the time span of a musical performance. The fourth triangle, including the thumb, uses as a base the segment between two knuckles, which, although not fixed, only varies slightly in practice. (One could also have taken the bone from the wrist to the first thumb knuckle as a base, but the triangle so formed is less easy to control than is the one we finally chose.)

For each triangle between points A, B, C where the base AB has fixed length, we measure the two remaining degrees of freedom as the signed dot product:

$$h = W \cdot V$$

and the absolute value of the cross product:

$$v = |W \times V|$$

where V and W are vector displacements from A to B and from B to C, respectively. The values h and v are equal to the length of the base AB times, respectively, the signed projection of the apex C onto the base (measured from the point B, and the (unsigned) altitude of the triangle. They therefore determine the coordinates of the point C within the plane of the triangle in a coordinate system whose origin is B and whose horizontal axis runs along AB.

We consider that the shapes of these four triangles, while not completely determining the shape made by the hand,

leave it little additional freedom. Moreover, the eight numbers so derived are almost independently controllable; and, while their combined range doesn't form an 8-cube (as truly independent uniform parameters would), the 8-dimensional blob that they do form is usable as a control space. We suppose also that the time resolution, or speed, with which the eight parameters may be controlled are roughly equal.

4. SYNTHESIS ALGORITHM

We know of no interesting synthesis algorithm that takes eight parameters and yields a sound that depends in even a roughly Cartesian way on them. A two-operator FM generator, for example, requires three parameters (if we exclude output gain), but the modulating frequency has either a large or a negligibly small effect on the output sound depending on the index, and the perceived pitch depends in a complicated way on the two frequencies and the index.

In general we have been at a loss to find any synthesis algorithm that can simultaneously offer a wide range of interesting sounds, predictability of sonic output, and independence of parameters. However, if we are willing to trade off predictablilty, for instance by choosing a method whose sonic output might not change as a smooth function of its input parameters, we can achieve much better parity and independence of the effect of parameter changes along several simultaneous dimensions and the perceived sound.

Our point of departure is the design by Don Buchla in his 200-series synthesizer of a pair of coupled oscillators, in which one provides a synchronization signal that can affect the internal phase of the other. Coupled oscillators have also previously been used in digital synthesis [1].

Our design uses two oscillators, each affecting the other's phase. The phase space is two-dimensional, coordinatized by the instantaneous phases of both oscillators. (In an analog circuit each oscillator would require at least two state variables but digital oscillators only require one apiece.)

Two un-coupled oscillators may be considered as the motion of a ball on an ideal pool table (see figure 3). The phase space is four times the surface of the pool table, since the ball may be traveling up or down and left or right; if we unwrap these four possibilities at any given point we arrive at a toroidal space of four times the area of the pool table itself. If the pool table acts ideally, the velocity of the ball is a constant 2-vector everywhere in the toroidal phase space. If we output the horizontal and vertical coordinates of the moving ball we get two sawtooth waves with independently controllable frequencies.

Our twist on this setup is as follows: on each of the four quadrants, we move at an independently chosen (x, y) velocity, always moving in the positive direction in both coordinates (so that we never get stuck). The result varies reasonably as a function of the eight velocity components, because increasing any of them increases any perceived frequencies present in the sound by decreasing the average time between bounces off the edges, but the effect on the sound timbre is in general hard to predict, and the path may change from periodic to aperiodic or vice versa as a result of small changes in the eight input parameters. Two example paths, one periodic and the other not, are shown in figure 4.

Our control strategy is simply to map the eight measured

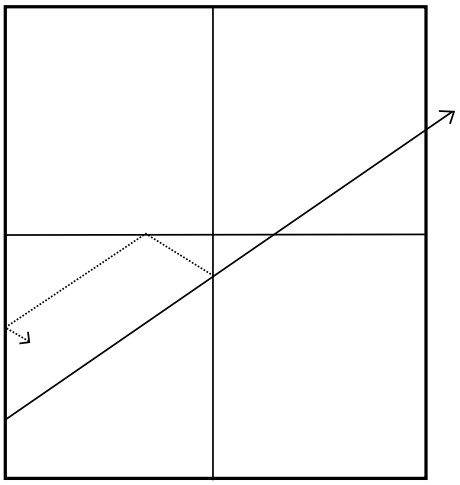

Figure 3. Pool table travel considered as two uncoupled oscillators.

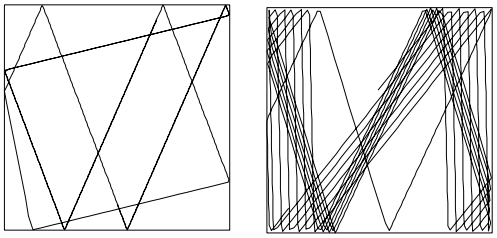

Figure 4. Pool table travel with different velocities for the four phase space quadrants.

dimensions of the four triangle shapes into a positive range (obtained by measuring the range of the raw values and rescaling.)

4.1 Output mapping

We may use any collection of real-valued functions of the state space to generate audio output from the time-varying state. Each output function corresponds to one channel of audio output.

For simplicity, in practice we use only the folded-down position as shown in figure 4, although if desired we could make the audio output depend also on which of the four quadrants the state lies in. If continuous waveforms with continuous derivatives are desired, candidate functions could take the form

$$\cos(\pi m x)\cos(\pi n y)$$

where m, n are nonnegative integers and (x, y) denotes the position with $0 \leq x, y \leq 1$. The simplest example would then be a half cycle of the cosine function aligned along x, ignoring y altogether.

The output function(s) may be time-varying and may depend on other measurements from the controller. In particular the overall amplitude of output (in each output channel) may be considered as a scaling factor in the corresponding output function.

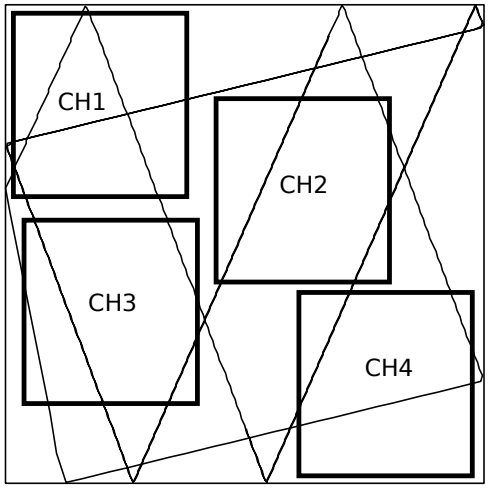

Figure 5. Spatialization and amplitude control using square regions of influence.

In our piece, we took advantage of the fact that the triangle shapes do not depend on the three-dimensional orientation of the hand, using those additional degrees of freedom to modify the output functions. This grants us both amplitude control and a spatialization strategy that is coherent with the sound world of the piece.

Each of the two performers generates four channels of output, placed mostly frontally, with each performer's outputs panned toward their side of the performance space (so that the audience can easily associate each sound stream with the performer making it.) A virtual-source-based surround spatialization approach would not fit well into this scheme. Instead, since the algorithm itself can be thought of as spatial, we model the output channels as windows into an exterior space, in a way somewhat like the SPACE algorithm[2]. Four squares are inset into the state space of the algorithm (actually the four-to-one collapsed state space), as shown in figure 5. The squares can controllably shrink almost to a point or grow to encompass the entire state space.

The four output functions are the product of $\cos(\pi x)$, $\cos(\pi y)$, and the indicator functions of each of the four squares (1 if the state is inside the square and 0 otherwise). If the square grows to fill the entire space the the sound is a bit like a waveshaped sinusoid. If the squares are small, and if the state path is moving slowly, one hears the moving location in state space as it becomes momentarily visible in each of four virtual windows, and the signal resembles width-modulated pulses. At higher speeds (controlled by hand geometry) the paths become too fast to follow audibly, offering a smooth transition between spatial motion and timbral evolution.

Two controls are thus needed, overall amplitude and the size of the square "window". this is supplied by the roll and pitch of the hand as measured by the vector normal to the palm. The roll, which can physically be changed quickly, is mapped to amplitude so that a level hand is full blast and a vertical one (thumb up or down) is silence; a horizontal pitch maps to maximum window size and vertical (fingers pointed up or down) gives a size near zero (in which case the sound is quiet, regardless of roll, because the pulses are narrow.) So although roll and pitch are not independently controllable, the result feels and looks natural. Yaw is not used; intuitively at least, it would seem less freely variable than roll or pitch.

Figure 6. Still image from performance of *Who Was That Timbre I Saw You With?* at NIME 2018, Virginia Tech.

4.2 Other ideas

The possibilities for embedding hand shapes into musical parameter spaces are probably limitless. One direction we've looked at but haven't yet implemented is to somehow map the convex hull of the hand to a synthetic vocal tract, measuring cross sections up and down the length of the hand and then using the classic lattice filter model of a tube with varying cross section [3], which might be perceived as an inarticulately talking hand.

5. CONCLUSION

The piece, *Who Was That Timbre I Saw You With?*, has received three performances so far, both in itself and as a lecture-demonstration. It is up to the audience to assess the musical value of the work. In any event it seemed to be better received in music-oriented settings than scientific or scholarly ones, which perhaps gives an indication. One reviewer from a scientific conference though the performance evoked a long-married couple fighting over the TV remote. Figure 6 shows a still image from that performance.

In this paper we have described our design strategies and realization in hopes of either inspiring, instructing, or perhaps dissuading whomever may now wish to pursue similar ideas. There is, as always, plenty of room for further exploration.

6. REFERENCES

[1] C. Heiland-Allen, "Lyapunov Space of Coupled FM Oscillators," in *Linux Audio Conference 2013*. Citeseer, 2013, p. 119.

[2] S. Yadegari, "Inner room extension of a general model for spatial processing of sounds," in *Proceedings of the International Computer Music Conference*, 2005.

[3] L. R. Rabiner and R. W. Schafer, *Theory and applications of digital speech processing*. Pearson Upper Saddle River, NJ, 2011, vol. 64.