GESTURAL STRATEGIES FOR SPECIFIC FILTERING PROCESSES

D. Arfib, J.M. Couturier, L. Kessous

LMA-CNRS, 31 Chemin Joseph Aiguier
13402 Marseille Cedex 20 France
`{arfib,couturier,kessous}@lma.cnrs-mrs.fr`

ABSTRACT

The gestural control of filters implies the definition of these filters and the way to activate them with gesture. We give here the example of several real “virtual instruments” which rely on this gestural control. This way we show that music making is different from algorithm producing and that a good gestural control may substitute to, or at least complement, a complex scheme using digital audio effects in real time implementations [1].

1. INTRODUCTION

The goal of controlling digital audio effects with gestural devices is to link the processing parameters of the effects to parameters issued from the physical measurement of gesture. The gestural control can this way make the digital effect a part of a real time instrument. Control has been addressed in a specific chapter of the DAFx book [2], and we will present here applications that can be explicitly linked, as practical examples, to what is write in this chapter. Filters are a very good example of simple and efficient digital audio effects. We will show here the importance of the gestural control [3] of such filters, which are included in the real-time instruments we have made using Max-Msp on Macintosh.

2. FIR FILTERS IN THE PHOTOSONIC INSTRUMENT

We will now describe the gestural control of FIR filters via a graphical tablet. This procedure comes from the emulation of an optical instrument, which provided the synthesis method and especially the gestural context [4], After giving a general description of these two versions: the optical instrument and its digital emulation, we will focus on moving a virtual filter according to two dimensions.

2.1. The optical and digital photosonic instruments

In the optical photosonic instrument, a rotating disk intercepts the rays of a light before it goes into a solar photocell (Fig. 1). This way the sound depends upon the position of the light and the waves inscribed on the photosonic disk. An "optical comb filter" can be placed on the trajectory of the light; this action crucially modifies the sound of the disk. The activity of such a filter and its possible emulation are described in this section.

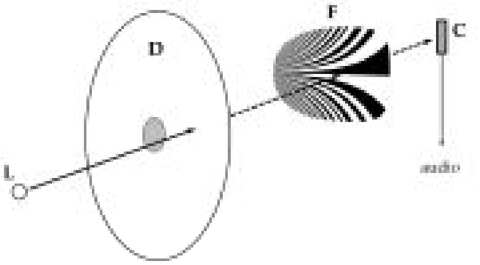

Figure 1. *The photosonic optical instrument.*

Horizontally moving the filter allows a dynamic filtering. In the case a comb filter is used, this generates a spectral scanning of the sound produced by the photosonic disc. With other optical filters, we obtain a succession of filterings depending upon the image. A vertical gesture produces a Doppler effect and vibrati can be created, as it is musically demonstrated in [1]. Moreover, the abrupt introduction of the filter gives a specific attack.

Figure 2. *The photosonic digital instrument control.*

The digital emulation of this instrument consists of a computer program written in the Max-Msp environment, linked to a digital tablet that allows a gestural control of the process. In this digital emulation, the sound source is produced by a mix of waves and is controlled by the left hand; the FIR filter is extracted from a digital version of the initial optical comb filter and is controlled by the right hand (Fig. 2).

2.2. The filtering process in the digital instrument

To emulate the optical instrument, a digital version of the optical filter has been implemented with the use of a Matlab program that computes the shape of the drawing and puts it into a matrix. Then a Max-Msp program extracts the FIR filter and applies it to the sound that it has calculated.

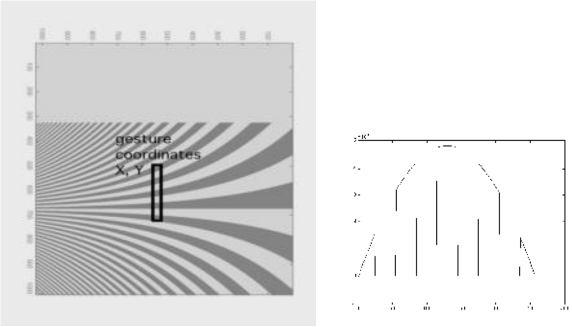

Figure 3. *Choice of the FIR filter.*

The FIR impulse response is the multiplication of a vector of 256 samples extracted from this matrix (Fig. 3, left), and a window that represents the shape of the photocell (Fig. 3, right).

$$h_i = M_{x,y+i} \cdot w_i \qquad i = 0,\ldots,255 \tag{1}$$

$$y_n = \sum_{i=0}^{255} h_i \cdot x_{n-i} \tag{2}$$

Here we have a classical situation of signal processing, and the Fourier Transform (FT) of this filter corresponds to the convolution of the FT of the 256-point vector extracted from the matrix and the FT of the window. In the case of a square comb filter (Fig. 4), we have only odd harmonic values, and a DC component due to the positive values in the matrix.

Taking different excerpts of the matrix, one obtains different FIR filters. The FIR is thus determined by the two coordinates in the matrix, which represent the placement of the optical filter in front of the cell in the original instrument.

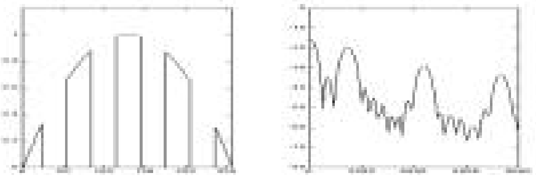

Figure 4. *FIR filter extracted from a square 2D filter and its Fourier Transform.*

The horizontal movement using the square comb matrix changes the shape of the filter, in our case the frequency of the square wave. The vertical movement will change the placement of this square wave inside the window. Moving the vertical coordinate is similar to move translatory the square shape inside this window. When a fixed frequency signal passes through this vertically moving filter, a Doppler effect is heard. This also means that an alternative vertical movement will provoke a vibrato at the haptic rate of the movement and with a depth proportional to the amplitude of the movement. This original feature of the optical filter is preserved in the emulator.

The coordinates of the starting point of the FIR filter inside the matrix have been linked to the physical coordinates of a stylus on a graphical tablet (fig. 3), thus allowing a movement similar to the one on the optical instrument.

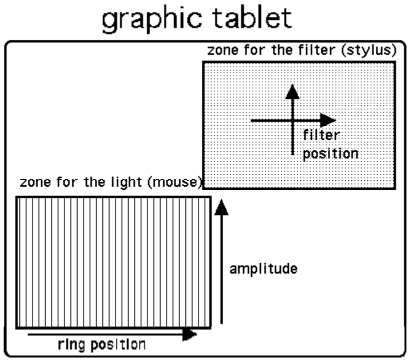

Figure 5. *Gestural control of the filter and the light.*

The complete digital instrument has a bimanual control: one hand governs the position of the light with a mouse, while the second one is devoted to the filter positions via a graphic stylet. It has been presented in a concert situation in Dublin (May 02) and from our experience the filtering process is an extremely important part of the play of this instrument [4].

Other filter matrixes have been constructed. The one on the left of Fig. 6 is a sweeping filter from one frequency to another (in this case the signal is bipolar). If we add three of them we can go from a configuration (fig. 6 right) of three sine waves to another configuration. The superimposing of the window shape (fig. 7) provides a formant structure in the FIR filter extracted form the matrix. As an example (fig. 6 on the right) one can construct a filter that sweeps from vowel “ee” to vowel “aah” by summing three sinusoidal filters that respectively sweep from 270 to 730 Hz, from 2290 to 1090 Hz and from 3010 to 2440Hz, according to values given in the Csound book [5].

;FORMANTS ;MALE
"EE", 270 2290 3010
"AAH", 730 1090 2440

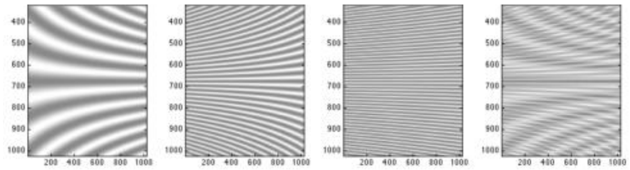

Figure 6. *Right: construction of a matrix (3-sine 2D filter) by summing 3 sinusoidal filters (left).*

The bandwidth of the formants corresponds to the FT of the window that is used (Fig. 7) and is in inverse proportion with the width of this window.

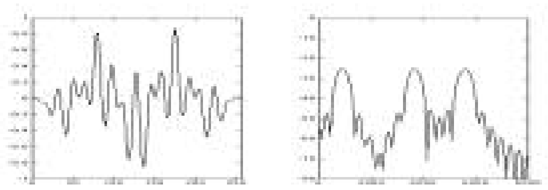

Figure 7. *FIR filter extracted from a 3-sine 2D filter.*

3. ALL POLE FILTERS IN THE VOICER INSTRUMENT

The “voicer” is a real-time electronic instrument in which a source is filtered by a cascade of three filters. The source and the filters’ algorithms are controlled by a graphic tablet and a joystick [6] (fig. 8).

Figure 8. *The voicer in concert situation.*

The source is usually a sawtooth wave, and in some configurations can be a sum of three sawtooth waves in order to give a choral effect. The filter is a cascade of three two-pole filters, which resonant frequencies and Q values imitate the formants structure of the natural voice (fig. 9), giving a vowel character to the instrument. However, fixed values produce a mechanical voice. For this reason, the gestural control of the formants is the essential part of this instrument.

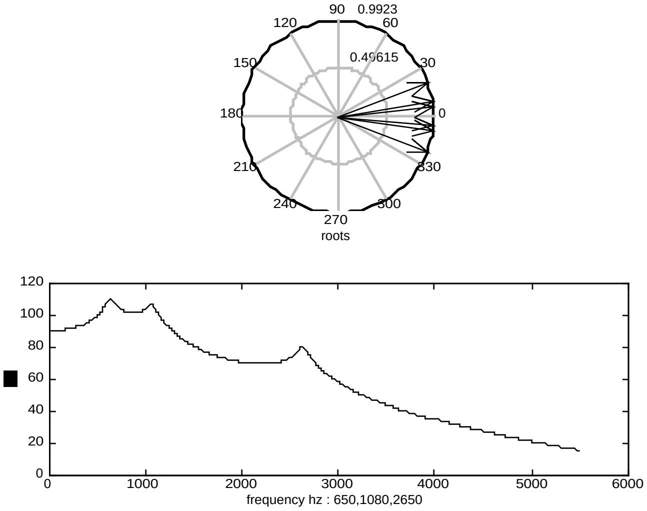

Figure 9. *A voicer filter in the z plane and its fft frequency response.*

To make a peak at $\omega = \omega_0$, we place a pole at the complex location: $p = R.e^{j\omega_0}$ where the pole magnitude is 0<R<1. Together with the conjugate pole $p^* = R.e^{-j\omega_0}$, we obtain the transfer function:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{G}{(1 - R.e^{j\omega_0}.z^{-1}).(1 - R.e^{-j\omega_0}.z^{-1})} = \frac{G}{1 - 2.R.\cos(\omega_0).z^{-1} + R^2.z^{-2}} \quad (3)$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{G}{1 + a1.z^{-1} + a2.z^{-2}} with: a_1 = -2.R.\cos(\omega_0), a_2 = R^2$$

$$and: G = \frac{(1-R)}{\sqrt{1 - 2.R.\cos(2.\omega_0) + R^2}} \quad (4)$$

$$Y(z) = G.X(z) - a_1.z^{-1}.Y(z) - a_2.z^{-2}.Y(z) \quad (5)$$

$$y(n) = G.x(n) - a_1.y(n-1) - a_2.y(n-2) \quad (6)$$

Voice synthesis has given rise since the early time when value tables to imitate the vowels appeared. One of the best and simplest tables comes from the beginning of the synthesis research and we use it to produce the key synthetic vowels [7]. Psychoacoustic researches have allowed placing all the vowels on a bi-dimensional map using a proximity measurement [8]. Therefore, the implementation is based upon the definition of a set of 4 key vowels at each corner defined by the three pole values of the filters (fig. 10).

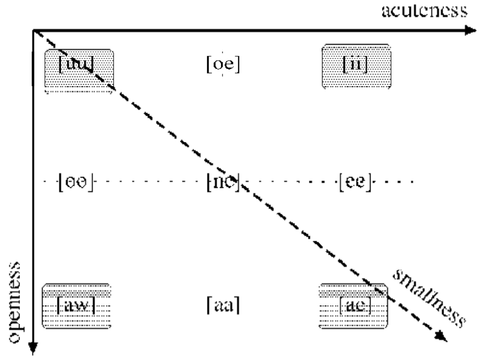

Figure 10. *The vowel space.*

The in-between vowels are defined by the interpolation of the resonant frequencies and the radius values of the filters. The formulation we give in Fig. 7 represents the interpolation between two vowels and has been generalized to interpolate between four key vowels. Moreover, in our implementation the radius value is constant: the interpolation of the angle values is sufficient.

$$\omega_0 = \alpha \,.\, \omega_{0k} + (1-\alpha) \,.\, \omega_{0l} \qquad k,l = 1,2,3 \quad (7)$$

(α: factor of the interpolation between two vowels)

Perceptually, this interpolation works well and the morphing of one vowel towards another is very natural. The instrument itself uses a 2D gestural controller (a joystick for example) to explore the vowel plane. In association with another gestural controller (a graphic tablet) for the pitch of the source wave, our device becomes a bimanual instrument with many musical possibilities [6]. This interpolation technique is derived from the early one implemented in the Syter [9] system. The design of Real-time interpolation tools is a key of musical use of filtering processes [10] [11] [12] [13].

In a musical context, we may use a real sound instead of a sawtooth wave for the source of the filter. This effect can be classified in the effect family of Talkbox, Wah-Wah, or

Vocoder. The Talkbox is well known by guitarist but also by other instrument players. The analog version consists of a tube that propagates a sound source (from a guitar for example) into the mouth of the performer, then a microphone records and an audio system amplifies the sound colored by the vocal resonance. The analog system is musically very interesting but imply some technical (and hygienic) difficulties that make it presently less used by musicians. Some interesting researches have been made to use the mouth shape to control filter related musical processes [14] [15]. The Wah-Wah uses a classical guitar player configuration with a 1-D continuous foot pedal to modify parameters effects by using one-to-one or one-to-many mapping. We have improved this configuration by adding degrees of freedom to a pedal and then controlling more expressive possibilities at the same time.

In this Talkbox-like configuration a different implementation is necessary to have a good perceptual behavior. In fact, a sawtooth signal in the cascade filter gives a spectrum that is related to glottal vibration. More precise and complicated calculation permits to adjust the cascade filter. Pre-filtering the real sound can be another solution. A more simple and efficient implementation consists in using parallel band-pass filter (one zero and two poles). In that case, five resonances give an interesting color to the sound. Musical perception is better if frequency bandwidths are small and all the resonance gains of each filter are equal, which seems to indicate a stressed articulation when one plays with a Talkbox.

4. REPRODUCING MUSICAL FILTERING GESTURE OF DIDJERIDU BY THE USE OF CEPSTRAL METHOD AND DATA BASE

The didjeridu [16] (sometimes spelt didgeridoo, and known as yidaki in the language of the Yolngu, one of the people of Northern Australia, where the instrument is from) is a remarkable instrument because of the wide variety of tones that it produces. Furthermore, this instrument is the most evident example of the interaction between a wind instrument and the vocal tract of the player. The sound of the didjeridu is given by the motions of the player's lips, the "filtering process" that is produced in the player's vocal tract, and the sound propagation into the instrument. Playing the instrument requires circular breathing that implies changes in sound and gives life to it.

To make an emulation of didjeridu playing we have separated the different musical gestures in the sound. Cepstral's method permits to separate the source s(t) which contains a lot of information about the circular breath musical (and motion of player's lips too) and formant structure h(t) that is more related to vocal tract filtering process.

$$F(\omega)=f[f(t)=s(t)*h(t)]= S(\omega).H(\omega) \qquad (8)$$

$$\log|F(\omega)|^2 = \log[|S(\omega)|^2]+ \log[|H(\omega)|^2] \qquad (9)$$

$$f[\log|F(\omega)|^2]=f[\log|S(\omega)|^2]+f[\log|H(\omega)|^2] \qquad (10)$$

(f[…] is the Fourier transform).

By analysis of a recording of the typical vowel-related circular filtering gesture known as "OU-I", we have extracted spectral envelopes with the cepstral method and store them in a file. With cross-correlation algorithms, we have localized where are the two more similar envelopes to know where is the beginning and the end of the "OU-I".

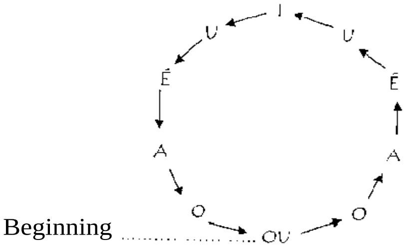

Figure 11: *Circular filtering gesture known as "OU-I".*

Then, spectral envelopes can be applied one by one (by multiplication in spectral domain using phase vocoder technique) in real-time with gesture control by using a circular gesture with the stylus of a graphic tablet.

5. EQUALIZER CONTROLLED BY SCANNED SYNTHESIS

We now present how to control an equalizer with the dynamical system of a Scanned Synthesis algorithm. Scanned Synthesis is a generic term for a synthesis method where a shape is generated at haptic rates (which means below 20Hz), while this shape serves as a waveform for a sound production [17]. We have implemented a Scanned Synthesis algorithm in Max-Msp, which uses a circular string model to create the shape. We have then developed complex mappings to create real time musical instruments [18] (fig. 12).

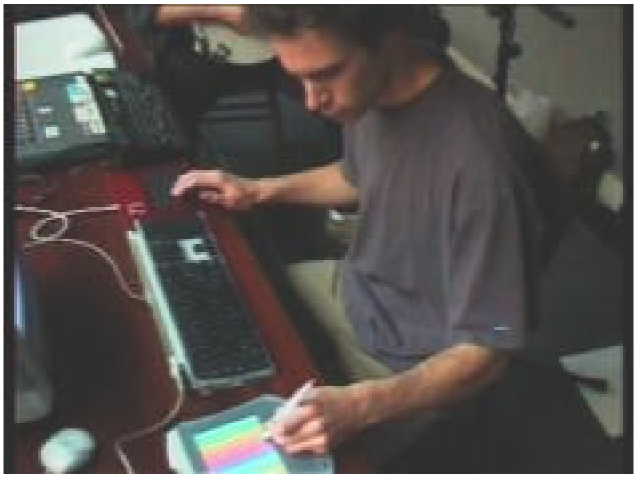

Figure 12. *A performance with the Scanned Synthesis instrument.*

As Boulanger suggests [19], the dynamical system can be used for the control of other sound processors, like additive synthesis or filters. In other way, filter parameters can be controlled with algorithms, including physical models that are coherent with the perception [2]. Here we propose an implementation in which the string model of the Scanned Synthesis algorithm drives an equalizer. The implementation combines two imbricate controls: the equalizer is algorithmically controlled by the string, and the string is controlled by gestures.

5.1. Algorithmic control of the filters

The control process of the equalizer filters' gain is represented in the following figure:

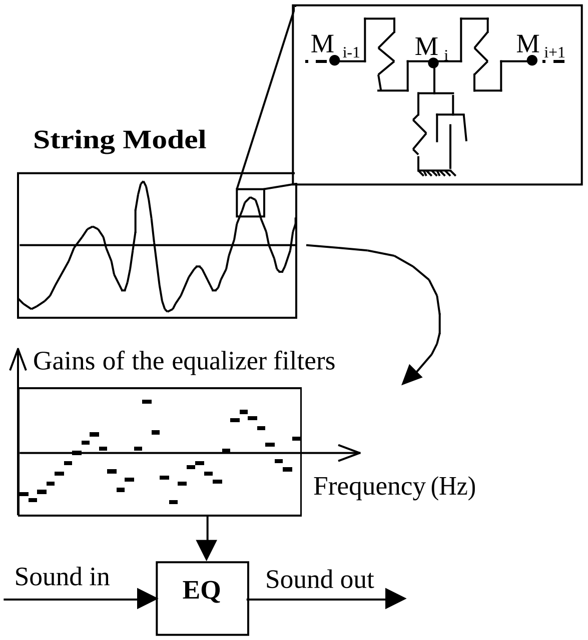

Figure 13: *The dynamic system is a slow-moving string modeled in finite differencies. Each element is composed of a mass, a spring and a damper that are connected to earth, and a spring that connect the element with the next one. The gains of the filters are mapped to the string shape.*

This implementation can be represented like a graphical filter whose sliders are attached to a slowly-moving string. The simplest way to do the mapping between the string shape and the filters gains is to choose a string model whose number of masses is the same than the number of filters. In this case, the cursors' position of the gain sliders corresponds to the position of the masses in the string model. The algorithm calculates the position of masses and simple expressions give the gains from the masses positions. Here we only control the amplitude of the equalizer filters, so any equalizer can be used; the relation between gain and string position have to be adapted depending on the chosen equalizer.

Our experiments were realized in the Max-Msp environment, with two equalizers: the *fffb~* object (fast fixed filter bank) a bank of IIR filter, and an equalizer using a FFT process implemented in a patch called *Forbidden Planet* [20]. The first one is composed of 32 filters, and the second one is controlled by 253 values (linear scale). We also have good results when the gain is controlled by the absolute value of the string; in that case, there is no sound out if the string is in its middle position, and sound comes out from the filter only when the string is excited.

In the Scanned Synthesis string model, the string is usually circular (the first and the last mass are linked with a spring) to avoid discontinuities in the scanning of the shape. Here the left side of the string corresponds to the low frequencies and the right side to the high frequencies, so connecting the extremity of the string is not necessary.

Experiments show that controlling filters with a haptic string is interesting because the filtering process is controlled by a system that has a well-known physical behavior. The gestural manipulation of the string will allow controlling the filters in the best way.

5.2. Gestural control of the string model

We have developed a research on the gestural control of the circular string parameters for the scanned synthesis algorithm [18] [21]; with few modifications the same devices can be used to control the equalizer filters gain.

We have experimented different configurations with different sensors. The major part of device can be the same as Scanned Synthesis control, but the way and the gesture process to control the system is strongly different. For example, with scanned synthesis, when one “freezes” the string, the sound seems poor (or no sound if the string shape is a constant) because the tone is not changing. With the equalizer control, the tone changes still follows the input sound. In the two cases, we control the same dynamic system, but the audio feedback pushes the user to have specific controlling process.

We now expose one example of control with a Tactex [22] touch surface (figure 14) that we have demonstrated at NIME-02 conference [18].

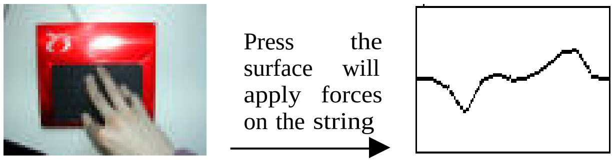

Figure 14: *The Tactex sensor is a rectangular touch surface composed of 6*12 pressure sensors. We use this sensor to apply forces on the string. Positive forces are applied when we put pressure on the highest part of the touch surface, and negative forces on the lowest part.*

The shape of the string can be designed through forces by the position of the fingers on the sensor. If the user puts his fingers on the surface and stops moving them, the string is motionless and the string shape will correspond to the forces profile. When the user changes the forces applied on the string by moving the fingers, the string follows the fingers' positions through its own dynamic behavior. The controlling board can be completed by foot pedals or other devices to control the intrinsic parameters of the string, such as tension and damping. This allows more control on the dynamic of the string.

The link between the shape of a haptic string and the gains of an equalizer represents an original way to filter sounds. Associated with a gestural control of the string, this implementation provides a complete filtering system, which has a huge effect on the sound dynamic.

6. CONCLUSION

Musical implications of digital audio effects are not always easy to set up. Though a lot of research has been done on the algorithms themselves (see the DAFx conferences) very few has been done and published on the musical use of these effects. Aside from the compositional use of these effects, the real time gestural control is still a field to develop in creative ways such as the ones this paper has addressed.

7. REFERENCES

[1] M. Wanderley & M. Battier, Ed, CD-ROM *Trends in Gestural control of music*, Ircam, 2000.

[2] T. Todoroff, “Control of digital audio effects”, *DAFX digital audio effects*, U. Zolzer Ed., J. Wiley & Sons, 2002, pp. 465-497.

[3] D. Arfib, L. Kessous, “Gestural control of sound synthesis and processing algorithms”, *Gesture workshop 2001*, ed. Ipke Wachsmuth & Timo Sowa, Springer-Verlag, Lecture Notes in Computer Science, LNAI 2298.

[4] D.Arfib, J. Dudon : “A digital emulator of the photosonic instrument”, *Proceedings of the 2002 Conference on New Instruments for Musical Expression (NIME-02)*, Dublin, Ireland, May 24-26, 2002, proceedings also on line at http://www.nime.org

[5] H. Mikelson, “Modeling a Multieffects Processor in Csound”, *The Csound Book*, R. Boulanger Ed., MIT Press, 2000, pp. 575-594.

[6] L.Kessous, “A two-handed controller with angular fundamental frequency control and sound color navigation”, *Proceedings of the 2002 Conference on New Instruments for Musical Expression (NIME-02)*, Dublin, Ireland, May 24-26, 2002.

[7] J.L. Flanagan, Ed., *Speech analysis, synthesis and perception*, Springer-Verlag, 2nd edition, 1972.

[8] W. Slawson, *Sound Color*, Berkeley University of California Press, 1985.

[9] Syter: http://www.genesis.fr/english/pro1.htm

[10] Tristan Jehan, Adrian Freed, and Richard Dudas, “Musical Applications of New Filter Extensions to Max/MSP”, *Proceedings International Computer Music Conference*, Beijing, China, 1999. http://cnmat.cnmat.berkeley.edu/ICMC99/papers/MSP-filters/filticmc.pdf

[11] GRM Tools: http://www.ina.fr/grm/outils_dev/grmtools/index.en.html

[12] Adrien Lefèvre, *Vect, package of Max's externals*, http://www.adlef.com/

[13] Daniel Terrugi, “L’interpolateur, une interface de contrôle multi-paramètrique”, *proceedings of the 8th Journées d’Informatique Musicale (JIM 01)*, Bourges, France, 2001.

[14] Michael J. Lyons & Nobuji Tetsutani, “Facing the Music: A Facial Action Controlled Musical Interface”, *Proceedings of CHI 2001 Conference on Human Factors in Computing Systems*. http://www.mis.atr.co.jp/~mlyons/mouthesizer.html

[15] N. Orio, “A Gesture Interface Controlled by the Oral Cavity”, *Proceedings of International Computer Music Conference*, 1997, Thessaloniki, pp. 141-144.

[16] Didjeridu: http://www.phys.unsw.edu.au/~jw/didjeridu.html

[17] B.Verplank, M. Mathews, R. Shaw, “Scanned Synthesis”, *Proceedings of the 2000 International Computer Music Conference*, pp. 368-371, Berlin, Zannos editor, ICMA, 2000.

[18] J.M. Couturier, “A scanned synthesis virtual instrument”, *Proceedings of the 2002 Conference on New Instruments for Musical Expression (NIME-02)*, Dublin, Ireland, May 24-26, 2002.

[19] R. Boulanger, P. Smaragdis, J. Ffitch, “Scanned Synthesis : An introduction and demonstration of a new synthesis and signal processing technique”, *Proceedings of the 2000 International Computer Music Conference*, pp. 372-375, Berlin Zannos editor, ICMA, 2000.

[20] Z. Settel, C. Lippe, “audio-rate control of FFT-based processing using few parameters”, *Proceedings of the 2nd COST G-6 Workshop on Digital Audio Effects (DAFx99)*, NTNU, Trondheim, December 9-11, 1999

[21] J.M. Couturier, “La synthèse par balayage et son contrôle gestuel”, *Proceedings of the 9th Journées d’Informatique Musicale* (JIM 02), Marseille, France, May 29-31, 2002.

[22] Tactex, touch surfaces, http://www.tactex.com/.

[23]