Visualizations and Interaction Strategies for Hybridization Interfaces

Adrian Freed John MacCallum Andrew Schmeder David Wessel

Center for New Music and Audio Technologies (CNMAT)
Department of Music
University of California, Berkeley
{adrian,john,andy,wessel}@cnmat.berkeley.edu

ABSTRACT

We present two complementary approaches for the visualization and interaction of dimensionally reduced data sets using hybridization interfaces. Our implementations privilege syncretic systems allowing one to explore combinations (hybrids) of disparate elements of a data set through their placement in a 2-D space. The first approach allows for the placement of data points anywhere on the plane according to an anticipated performance strategy. The contribution (weight) of each data point varies according to a power function of the distance from the control cursor. The second approach uses constrained vertex colored triangulations of manifolds with labels placed at the vertices of triangular tiles. Weights are computed by barycentric projection of the control cursor position.

Keywords

Interpolation, dimension reduction, radial basis functions, triangular mesh

1. INTRODUCTION AND MOTIVATION

In this paper, we present two user interfaces for dimensionality reduction. The first allows data points represented by radial basis functions to be placed in a space and organized dynamically according to the goals of a given composition, musical performance, or pedagogical application. Similar data points can be organized together, or data points that will contribute to the same musical goal in a composition or improvisation can be placed near each other regardless of any judgement of their similarity.

The second method assigns the data points to the nodes of a triangular mesh in an arrangement that ensures that many transitions between combinations are available and can be explored efficiently.

Hybridization interfaces are powerful tools for managing the large and ever growing number of control variables available to the artist and performer working on image, motion, and sound synthesis. Using these interfaces, we navigate through a low-dimensional control space generating weights to combine multidimensional parameter sets from a small number of interesting data points. This approach is well known in image morphing [6], especially of faces, and has been used widely for the interpolation of musical timbre, meters, pitch spaces, and audio effects [12], [7] and [8]. We use the term *hybrid* here as a way to place emphasis on the new forms that arise as the user transitions from one data point to another, rather than on the data points themselves. We recognize, however, that a data point must be recoverable without the influence of any other data points in the space and ensure this is the case in both approaches that we present.

An important concern for designers of new musical instruments for the control of software synthesizers is to approach the level of nuanced control that a performer has when playing a traditional instrument. A violinist, for example, can create extremely subtle variations in spectral centroid, noisiness, and entropy by manipulating the bow position, pressure, and speed respectively. Slight timbral variations can also be produced by string choices and playing certain pitches as harmonics. Other modal acts such as the use of a mute (plastic, rubber, wood, leather, lead, etc.) and bow choice allow the performer to alter the sound of the instrument. With the computer, we are freed from the constraints of a physical instrument and are able to separate the interface from the sound source. A problem arises as we attempt to map an interface to the many parameters we wish to control. These mappings should occur in such a way that is intuitive, while allowing the performer to explore all possible combinations therein.

1.1 Interpolation Criteria

The majority of the interpolation properties presented in Section 2 of [5] are also applicable here. In particular, normalization is critical in audio applications where gain is a factor. We also typically need an exact interpolator that will allow us to visit a data point without the influence of any other data points in the space.

2. SOFTWARE INTERFACE

2.1 Radial Basis Functions

Data points may be placed in the space and moved with a pointer. Both the range of influence and the steepness with which the influence rolls off take the form of two concentric radii. In Figure 1, the steepness of the rolloff changes as the inner radius increases. In the fourth image of Figure 1 the inner radius has become larger than the outer radius, creating a null region where that data point has little influence surrounded by a region of great influence.

In Figure 3, we have 50 points arranged randomly in the space. By manipulating the values of the two radii, the space can take on wildly different characteristics while the data points remain at the same locations.

NIME2010. 15-18th June 2010, Sydney, Australia

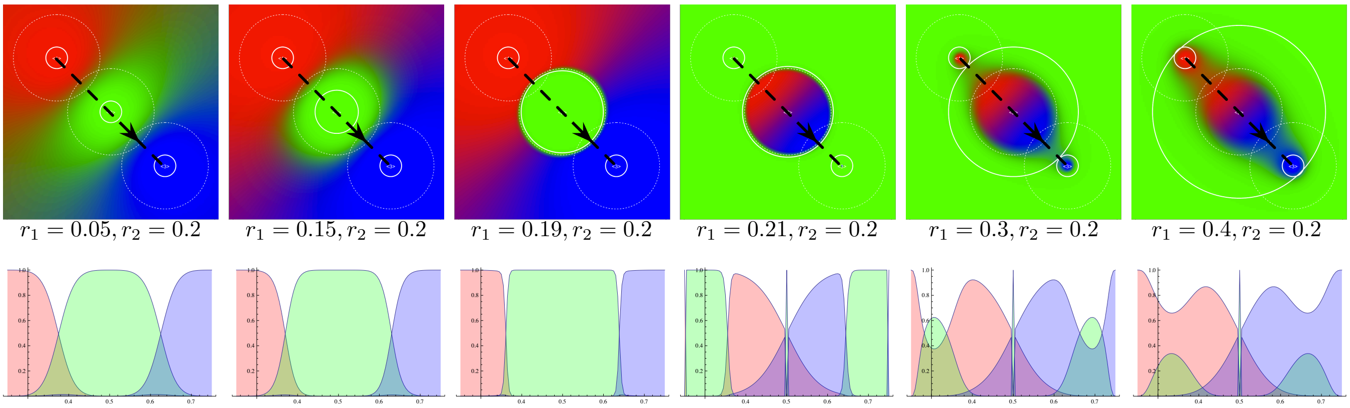

Figure 1: Three points in the space with different values for the inner radius (r_1) of the center (green) point. A straight path from the center of the red point to the blue point generates the weights plotted against time below.

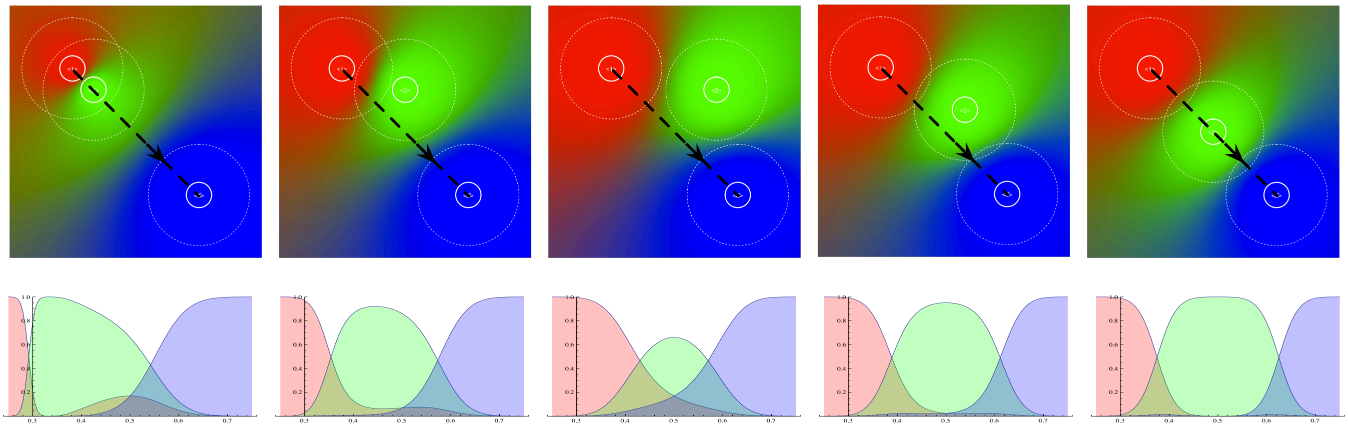

Figure 2: Three points in the space. By changing the position of the green point, different mixtures are produced by taking the same path through the space.

2.2 Triangular Mesh

The triangular mesh arrangement of 3 points on the plane seen in Figure 3.2 is both symmetric and periodic with respect to the way that the space is filled. This offers the user a more predictable layout in the space and ensures access to many of the possible combinations and transitions between them. This layout can be useful to explore combinatorially, the relationships between elements of an unfamiliar data set.

3. IMPLEMENTATION

3.1 Radial Basis Functions

Let L represent a layout given by a set of n data points

$$L = \{P_i, \mathbf{p}_i, r_{1_i}, r_{2_i}, c_i\} \text{ for } i = 1 \ldots n \qquad (1)$$

where $P_i \in \mathbb{R}^m$ is the data value, $\mathbf{p}_i \in \mathbb{R}^d$ is its location in d-dimensions; in 2-d, $p_i = (x_i, y_i)$. Typically, in the case of dimensionality reduction, $d \ll m$. r_{1_i} and r_{2_i} are the two radii that determine the values of α and β in equation 4, and c_i is the color of the data point in the space used for drawing the visual representation.

The interpolation function is a map from $\mathbb{R}^d$ to $\mathbb{R}^m$ given by

$$S_L(\mathbf{q}) = \sum_{i=1}^{n} w_i P_i \qquad (2)$$

where w_i is the weight or contribution of each data point. An important property of our model is that

$$\sum_{i=1}^{n} w_i = 1 \qquad (3)$$

Equation 2 is written as a linear combination of P_i, but this hybridization can be generalized using any nonlinear mechanism that can be parameterized according to the definition of w.

We compute the contribution w_i of each data point P_i

$$w_i = \frac{1}{\sum_{i=1}^{n} w_i} \left(\sum_{i=1}^{n} \alpha_i d_i^{-\beta_i} \right) \qquad (4)$$

where

$$d_i = ||\mathbf{p}_i - \mathbf{q}|| \qquad (5)$$

and $||\cdot||$ is the norm in $\mathbb{R}^d$, e.g. the L_2 norm in $\mathbb{R}^2$ is

$$||\mathbf{p}_i - \mathbf{q}||_2 = \sqrt{(x_i - x)^2 + (y_i - y)^2}. \qquad (6)$$

The norm can be used to change the shape of the area of influence of a data point.

The values of α_i and β_i are derived from r_{1_i} and r_{2_i}

$$\alpha_i = \begin{cases} e, & \text{if } r_{1_i} = 1, r_{2_i} > 1 \\ 1/e, & \text{if } r_{2_i} = 1, r_{1_i} > 1 \end{cases} \qquad (7)$$

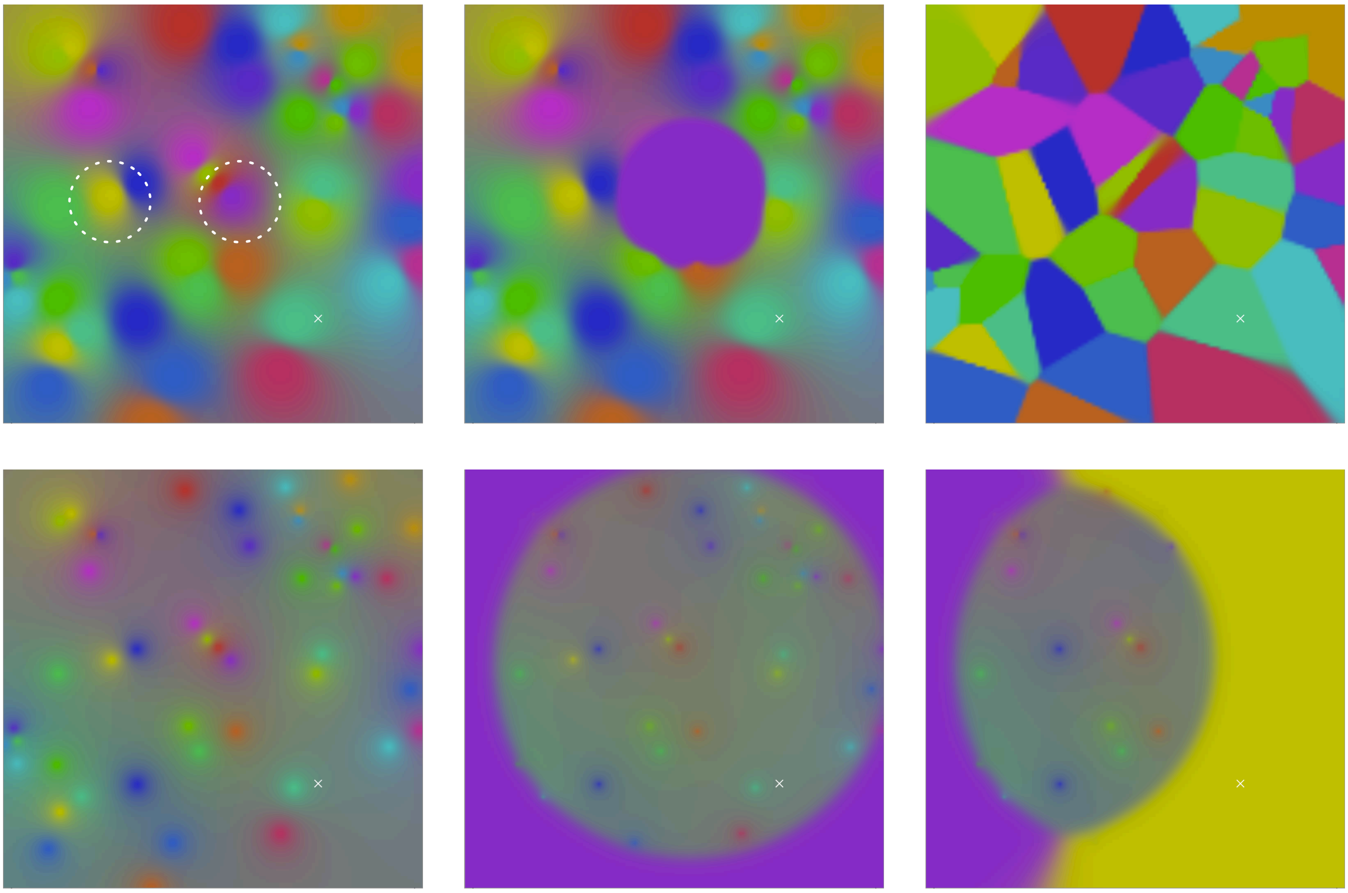

Figure 3: 50 points randomly positioned in the space with different settings of the inner and outer radii. From top left to bottom right: 1. $r_1 = 0.05, r_2 = 0.2$ for all points. 2. $r_1 = 0.19$ for the purple point circled in the first image. 3. $r_1 = 0.19, r_2 = 0.2$ for all points. 4. $r_1 = 0.0005, r_2 = 0.01$ for all points. 5. The purple point circled in the first image has been "inverted" ($r_1 = 0.474399, r_2 = 0.469678$). 6. Both points circled in the first image have been "inverted" (yellow: $r_1 = 0.429808, r_2 = 0.423348$).

$$\beta_i = \begin{cases} \frac{2}{\log r_{2_i}}, & \text{if } r_{1_i} = 1, r_{2_i} > 1 \\ -\frac{2}{\log r_{1_i}}, & \text{if } r_{2_i} = 1, r_{1_i} > 1 \end{cases} \tag{8}$$

In the case where $r_{1_i} < r_{2_i}$, the contribution of a data point approaches infinity as the distance decreases

$$\lim_{d_i \to 0} w_i = \infty. \tag{9}$$

When $r_{1_i} > r_{2_i}$, the contribution of a data point approaches infinity as the distance from it increases thus changing the limit behavior of the model. This can be seen in Figure 1 where r_1 of the central (green) point has become larger than r_2 in the three last images.

3.2 Triangular Mesh

Let M represent a triangular mesh layout given by a set of n data points

$$M = \{P_i, \mathbf{p}_i, c_i\} \text{ for } i = 1 \ldots n \tag{10}$$

where the $\mathbf{p}_i$ are constrained to lie on the points of a triangular mesh in two dimensions.

The interpolation function at a point $\mathbf{q}$ is a combination of the three nearest $\mathbf{p}_i$, $\{\mathbf{p}_{t_1}, \mathbf{p}_{t_2}, \mathbf{p}_{t_3}\}$.

$$S_M(\mathbf{q}) = \sum_{i=1}^{3} \lambda_i P_{t_i} \tag{11}$$

where

$$\lambda_1 + \lambda_2 + \lambda_3 = 1 \tag{12}$$

and λ_i are the barycentric coordinates of $\mathbf{q}$

$$\lambda_1 = \frac{(y_{t_2} - y_{t_3})(x - x_{t_3}) - (x_{t_2} - x_{t_3})(y - y_{t_3})}{\det(\mathbf{T})} \tag{13}$$

$$\lambda_2 = \frac{-(y_{t_1} - y_{t_3})(x - x_{t_3}) + (x_{t_1} - x_{t_3})(y - y_{t_3})}{\det(\mathbf{T})} \tag{14}$$

$$\lambda_3 = 1 - \lambda_1 - \lambda_2 \tag{15}$$

where $\mathbf{T}$ is a matrix

$$\mathbf{T} = \begin{pmatrix} x_{t_1} - x_{t_3} & x_{t_2} - x_{t_3} \\ y_{t_1} - y_{t_3} & y_{t_2} - y_{t_3} \end{pmatrix} \tag{16}$$

We can tell if a point is inside a triangle or on an edge if $0 < \lambda_i$ or $0 \leq \lambda_i \leq 1 \;\forall\; i$ in 1, 2, 3 respectively.

3.3 Generalization of the Triangular Mesh to Higher Dimensions

Goudeseune [3] (Section 4.2.2) points out that as the triangle in $\mathbb{R}^2$ generalizes to a *d-simplex* in $\mathbb{R}^d$, this triangluar mesh generalizes to a *simplical d-complex*. In $\mathbb{R}^d$ where $d > 2$, the irregular polyhedra required to tile the space impose an asymmetric scaling in the mapping between the gestures driving the cursor and the interpolation-weight space. Devices such as the array of touch pads discussed below in Section 4.1 possess such an asymmetry and thus are well-suited to this type of extension. For devices such as the GameTrak [2], however, a space tiled with nearly regular

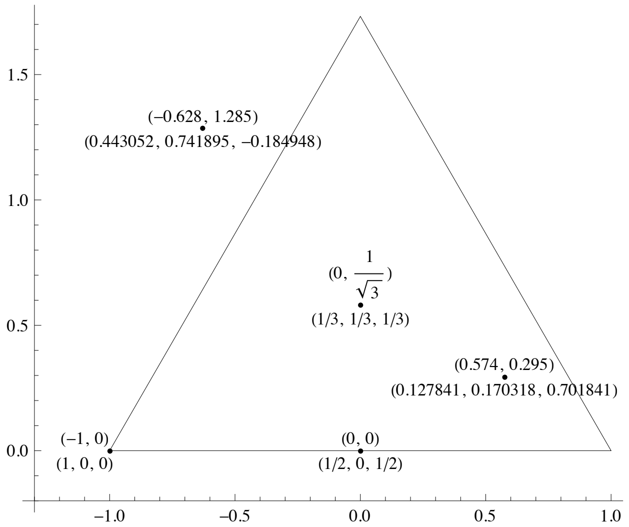

Figure 4: Triangle with Cartesian coordinates above each point and barycentric points, which represent the amount of influence of each vertex, below.

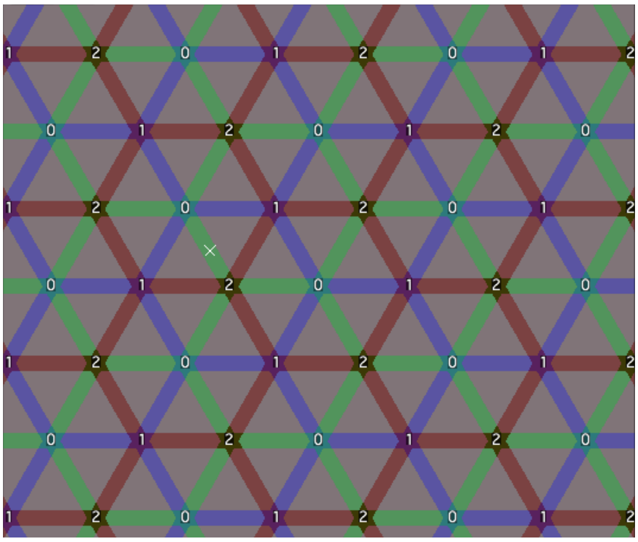

Figure 5: A layout of 3 data points on a trianglular mesh.

polyhedra as proposed by Eppstein et al. [1] may be more appropriate.

The flat 2-D surfaces presented above can also be wrapped around various solids such as a sphere or a torus which can allow the user to revisit an area of the space without a change of direction. One could imagine a monotonic process that, once set in motion, would produce a periodic path through the space. Such a process could be useful for modelling rhythmic processes.

4. APPLICATIONS

4.1 Implementation on an Array of Touch Pads

CNMAT has developed controllers based on pressure sensitive 2-D touch pads [10], [11], and [13]. One such controller uses 32 touch pads each of which can be used in a straight forward manner as the pointing device for the interpolation algorithms presented here. In elementary implementations pressure is used to control the intensity and the results have proven to be musically expressive.

In an effort to make more expressive use of pressure, we developed a multilayered approach wherein each 2-D interpolation controller has some small number of spaces, usually two or three. We then select from those layered spaces using pressure: a space containing low intensity material will be selected with a light touch, and increasing pressure will cause spaces with increasingly intense material to be selected. This method allowed us to use the weights generated by the stacked spaces to create a smooth interpolation associated with dynamics.

4.2 Research Applications

Wessel et al. developed a version of the radial basis function software as part of a project to optimize hearing aids for music [14]. Subjects were presented with musical stimuli and asked to adjust the compression and equalization. The technical nature of the parameters made their adjustment by subjects lacking a background in audio engineering impractical. By placing data points representing different configurations of the compression and equalization parameters into a 2-D space, subjects were able to easily and intuitively explore the different settings.

5. CONCLUSIONS AND FUTURE WORK

We have presented two methods of dimensionality reduction for dealing with large sets of control parameters. One approach offers the user flexibility with respect to spatial layout, while the other imposes a regular tiling of the space but offers the user a symmetrical and periodic arrangement of the data points.

Dimensionality reduction implies that the set of all mixtures of parameters reachable through the interpolation scheme is a d-dimensional manifold in $\mathbb{R}^n$. This means that not all combinations of data points are reachable. The triangular mesh scheme presented above ensures that all combinations of up to three data points are reachable. The radial basis function method does not ensure this for any number of presets greater than one, although one could employ the tiling strategy of the triangular mesh to approximately reach all combinations of three data points.

In practice, certain data points may be judged to be similar to others in the space. In order to maximize the variation of reachable mixtures through interpolation, the spacing between similar presets in the d-dimensional space should be inversely proportional to similarity. For small numbers of data points (e.g. fewer than 10), it is possible to create layouts with this property manually, but for larger data sets, optimization methods such as those suggested by [4] and [9], can be used to lay out the space. Future work will include the implementation of such optimized layout strategies.

As mentioned above, both of the methods we present generalize to higher dimensions which makes them particularly well-suited for use with >2-D controllers such as the GameTrak and the Wii Remote. We are planning new higher-dimension implementations of the techniques presented here. Since manual layout of a >2-D space can be challenging, we will have to rely on optimization techniques such as those mentioned above.

6. ACKNOWLEDGEMENTS

The authors wish to express their gratitude to the those at Meyer Sound, and Sha Xin Wei for their invaluable feedback during the development of this work.

7. REFERENCES

[1] D. Eppstein, J. M. Sullivan, and A. Üngör. Tiling space and slabs with acute tetrahedra. *Computational Geometry*, 27(3):237–255, 2004.

[2] A. Freed, D. McCutchen, A. Schmeder, A.-M. S. Hansen, D. Overholt, W. Burleson, C. N. Jensen, and A. Mesker. Musical applications and design techniques for the gametrak tethered spatial position controller. In *SMC 2009*, 2009.

[3] C. Goudeseune. Interpolated mappings for musical instruments. *Organized Sound*, 7(2):85–96, August 2002.

[4] P. Kassakian and D. Wessel. Optimal positioning in low-dimensional control spaces using convex optimization. In *Proceedings of the International Computer Music Conference*, pages 379–382, 2005.

[5] M. Lombardi. Interpolation and smoothing. *Astronomy & Astrophysics*, 395:733, 2002.

[6] W. Matusik, M. Zwicker, and F. Durand. Texture design using a simplical complex of morphable textures. *ACM Trans. Graph.*, 24(3):787–794, 2005.

[7] A. Momeni and C. Henry. Dynamic independent mapping layers for concurrent control of audio and video synthesis. *Computer Music Journal*, 30(1):49–66, Spring 2006.

[8] A. Momeni and D. Wessel. Characterizing and controlling musical material intuitively with geometric models. In *Proceedings of the International Computer Music Conference*, 2003.

[9] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[10] A. Schmeder and A. Freed. A low-level embedded service architecture for rapid diy design of real-time musical instruments. In *Proceedings of the 9th International Conference on New Interfaces for Musical Expression*, 04/06/2009 2009. real-time musical interface, DIY design, embedded web services, rapid prototyping, reconfigurable firmware.

[11] A. Schmeder and A. Freed. Support vector machine learning for gestural signal estimation with a piezo-resistive fabric touch surface. In *Proceedings of the 10th International Conference on New Interfaces for Musical Expression*, 2010.

[12] D. Wessel. Timbre space as a musical control structure. *Computer Music Journal*, 3(2):45–52, 1979.

[13] D. Wessel, R. Avizienis, A. Freed, and M. Wright. A force sensitive multi-touch array supporting multiple 2-d musical control structures. In *Proceedings of the 7th International Conference on New Interfaces for Musical Expression*, 2007.

[14] D. Wessel, K. Fitz, E. Battenberg, A. Schmeder, and B. Edwards. Optimizing hearing aids for music listening. In *Proceedings of the 19th International Congress on Acoustics*, 2007.