2018

Vocal Control of Sound Synthesis Personalized by Unsupervised Machine Listening and Learning

Stefano Fasciani
_University of Wollongong in Dubai, fasciani@uow.edu.au_

Lonce Wyse
_National University of Singapore_

Publication Details

Vocal control of sound synthesis personalized by unsupervised machine listening and learning.

Stefano Fasciani  
Faculty of Engineering and Information Sciences  
University of Wollongong in Dubai  
Knowledge Park, Block 15, P.O. Box 20183  
Dubai, UAE  
stefanofasciani@stefanofasciani.com  

Lonc Wyse  
Communication and New Media Department  
National University of Singapore  
11 Computing Drive, 117416  
Singapore  
lonc.wyse@nus.edu.sg

Abstract

This article describes a user-driven adaptive method for controlling the sonic response of digital musical instruments with information extracted from the timbre of the human voice. The mapping between heterogeneous attributes of the input and output timbres is determined from data collected via machine listening techniques and then processed by unsupervised machine learning algorithms. This approach is based on a minimum-loss mapping which hides any synthesizer-specific parameters, and maps the vocal interaction directly to perceptual characteristics of the generated sound. The mapping adapts to the dynamics detected in the voice, and maximizes the timbral space covered by the sound synthesizer. The strategies for mapping vocal control to perceptual timbral features, and for automating the customization of vocal interfaces for different users and synthesizers in general, are evaluated through a variety of qualitative and quantitative methods.

Introduction

The importance of mapping in the design of expressive interfaces for musical instruments has long been recognized (Poupyrev et al. 2001; Hunt, Wanderley, and Paradis 2003). Mapping techniques have been at the heart of the development of novel musical instruments (Wanderley and Depalle 2004) and of generic interactive sound controllers. Regardless of the input technology employed, most interfaces for digital musical instruments require physical motor interaction from users, captured with or without contact (Bongers 2000). A large proportion of interfaces are built for the hands,
evinced by a focus of many devices on tactile, haptic, and gestural input modalities. However human physical limits restrict the number of parameters or events that performers can control simultaneously using these interfaces. Few-to-many mappings or pre-programmed automations are common workarounds. With the exception of ‘augmented instruments’ that aim to saturate performers’ spare bandwidth (Cook 2001), this issue is rarely addressed in the design phase of instrument interfaces.

The aim of the framework described herein is to provide users with a simple system to implement personalized vocal control of sound generators, adapted to the sonic characteristics of both their voice and the output of the specific synthesis process of their choice. This enables the augmentation of control dimension through voice for controlling synthesis algorithms. The system is largely self-configurable, minimizing user effort in setting up the personalized interface. The balance between simplicity of user requirements and richness of musical expression (Stolterman 2008) is achieved in two stages. First, we establish a natural interaction between voice and sound synthesizers, implementing the mapping on the perceptually related timbre layer. Secondly, we develop and combine machine listening and unsupervised learning techniques to compute the mapping automatically from a minimal set of user-provided information. The framework we describe here addresses: (1) expansive and complex timbral capabilities of sound synthesis algorithms, which require high dimensional non-linear control spaces; (2) noisy characteristics of the vocal control signal, which challenge unsupervised mapping creation; (3) interaction with perceptually relevant characteristics of the sound rather than with the complicated synthesis parameters. The resulting mapping strategy maximizes the coverage of the perceptual sonic space of synthesizers given the span of the timbral domain of the controlling voice. It also minimizes the cognitive complexity of controlling sounds embedded in high-dimensional parameter spaces by providing a lower dimensional control with unchanged timbral potential. This paper summarizes the system integrating these
components, some of which have been described in detail in (Fasciani and Wyse 2013; Fasciani 2016), and then evaluates the effectiveness of automated vocal-to-synthesizer timbre mapping strategies through a variety of qualitative and quantitative user studies.

**Related Works**

Indirect gestural acquisition refers to strategies to extract control data from a sound source (Wanderley and Depalle 2004). Extraction of control intention from the sound of a clarinet has been proposed in (Puckette and Lippe 1994) by tracking the temporal dynamics of amplitude, pitch, and the principal components of timbre descriptors. Analysis of the timbre of plucked guitar based on MFCC, psychoacoustic parameters, time-frequency analysis, principal components analysis, and plucking point estimation has been used to implement musical interaction in (Orio 1999; Traube, Depalle, and Wanderley 2003). In contrast to synthesis algorithms, acoustic instruments generally allow only minor timbral nuances as the sound generated is constrained by their physical bodies, hence the data extracted from these sources is not optimally suited for control of complex synthesis engines. Generic methods to compute control data from audio signals have been proposed as well (Lazier and Cook 2003; Poepel and Dannenberg 2005; Schnell, Cifuentes, and Lambert 2010). These methods rely on the extraction of fixed features such as the Mel-Frequency Cepstrum Coefficients (MFCC), pitch, harmonic tracking, and timbre dynamics.

Several previous works proposed the voice as a gestural control source due to its wide timbral range. Oliver, Yu, and Metois (1997) extract ten dynamic parameters such as pitch, loudness, formants, cepstra, and their deviations which are used to interact with an ensemble of instruments. Janer (2008) developed a system to interact vocally with the synthetic emulation of acoustic instruments, singing voice synthesis, and audio mosaicing by explicitly mapping from detected vocal pitch, energy envelope, onset, and
syllable temporal segmentation. In (Hazan 2005), a set of spectral vocal features and an energy measure computed over the onset frame are used to query a decision tree to trigger sampled drum sounds, while a similar method has been applied to retrieving drum loops from a database (Kapur, Benning, and Tzanetakis 2004). In (Teglbjærg, Andersen, and Serafin 2015), the live voice signal replaces white noise as the excitation signal in the Karplus-Strong string physical model algorithm. Acoustic similarities between voice and sound segments are used in the interactive system of König (2006). Several approaches (Orio 1997; Ramakrishnan, Freeman, and Varnik 2004; Janer and De Boer 2008) use the principal components derived from Linear Predictive Coding (LPC) coefficients or MFCC representations of the voice as control signals. Loscos and Aussenac (2005) propose the use of the area of the weighted low-frequency spectrum computed from the voice signal to control a wah-wah effect, while Stowell (2010) computes a mapping based on the principal components of larger sets of vocal features to interact with the parameters of synthesis engines through an auto-associative regression tree. Low-level features extracted from vocal imitation of sounds have also been used for searching a collection of synthesis preset patches (Cartwright and Pardo 2014) with reinforcement learning, and to search a large database of pre-recorded sounds (Roma and Serra 2015). These works consider the voice as an immediate and natural means of expressing sound, and thereby enable fast ‘prototyping’ or ‘sketching’ of sounds (Cartwright and Pardo 2015; Delle Monache et al. 2015). The innate human ability to control the vocal timbre, rather than specific articulators (Lemaitre et al. 2013) is exploited, and it does not require the mastery new skills. Most of the existing methods are designed to address specific synthesis methods or applications.

Machine listening and machine learning have been broadly used in creative applications such as sonic interactive systems and mapping for novel musical interfaces. Following the early work of (Wessel 1991) with Artificial Neural Networks (ANN), a variety of machine learning techniques have been integrated in such systems, including
classification, clustering, and regression algorithms (Caramiaux and Tanaka 2013). In particular these are used for mapping purposes and gesture recognition. Generally, musical interfaces and sonic interactive systems based on machine learning do not provide a ready-to-use mapping, but they support the implementation of personalized systems. Supervised machine learning for sonic interaction has been proposed in (Bevilacqua et al. 2010), where a Hidden Markov Model is used to follow the temporal unfolding of user gestures, in (Fiebrink 2011) and (Gillian, Knapp, and O’Modhrain 2011) which provides an ANN and classifiers integrated in a user-friendly generic mapping environment, as well as in (Scurto and Fiebrink 2016) to create mappings based only on user demonstrations of control gestures. Unsupervised machine learning has been proposed as well (Smith and Garnett 2012, 2011), including the Self-Organizing Maps (SOM) method, which has been used for automatic sonic data organization (Ness and Tzanetakis 2009). Instrument designers aiming for generative mappings can rely on toolkits for visual programming languages easing the integration of a wide spectrum of machine learning algorithms (Bullock and Momeni 2015).

In this article we describe and evaluate the integration of a user-driven generative mapping framework based on several techniques we introduced in (Fasciani 2012; Fasciani and Wyse 2013; Fasciani 2016). The method is independent of the specific synthesis method and it measures the perceptual timbre response of any deterministic sound synthesizer, providing low-dimensional and perceptually-based interaction independent of the type and number of synthesis parameter controlled. The strategy for indirect gestural acquisition from voice includes the automated selection of features to extract, and the adaptation to the users’ vocal interaction style, while providing multiple degrees of freedom. This approach depends upon machine listening and unsupervised learning techniques that work with minimal amounts of training data, and expose a manageable number of meaningful choices to users.
System Architecture

There are two major components comprising the integrated system: the vocal gestural controller (Fasciani and Wyse 2013), and the synthesis timbre space mapping (Fasciani 2016), as illustrated in Figure 1. The first is built upon robust and noise-free control signals extracted from the voice (Fasciani 2012), representative of control intention expressed by sub-verbal vocal gestures. The second component maps the control signals computed from the voice onto the synthesis timbre space, from which we can retrieve the parameters to control the synthesizer itself. Both components use machine listening, the first to analyze the user’s voice and the second to analyze synthesis output. The analysis data is then used to train the system that implements the voice-to-synthesis parameter mapping. We assume that the deterministic synthesis algorithm may generate any sound, while the sonic space of vocal articulation is very limited. The voice can be noisy and inaccurate because it is human-generated, whereas deterministic synthesis algorithms always produce the same timbre for a given set of parameter values. To provide tight coupling (i.e. low latency) between control action and system response we consider only the instantaneous characteristics of voice. Instead, timbre changes over longer periods, such as low-rate modulations, or attack and decay, are considered in the analysis of the synthesizer, especially because these may vary when changing parameters. Therefore the features we compute from voice and synthesis signals, and the processing methods to extract relevant information, significantly differ across the two components of the system. The system architecture is based on the concept of remapping across these heterogeneous domains. The spatial representation over which both the voice and the synthesis timbres move are considered as manifolds, each with an individual shape and distribution. The generative mapping finds the homomorphic (structure-preserving) transformation that maximizes their overlap. The system is constrained to only two or three intermediate dimensions across these spaces, to limit both computational and interaction cognitive
complexity. This also simplifies the display of user vocal interactions, an important feedback element in the interface for the user.

![Diagram](image)

**Figure 1**: The architecture of the vocal control of the sound synthesis system, showing the voice being transformed into synthesis parameters through two stages: a vocal gestural controller and the timbre space mapping.

**Computation of a control space from voice data**

Using the voice as a source of musical control offers two key benefits. First, it is often ‘spare bandwidth’ otherwise unused by performers engaged with musical interfaces. Second, it offers an immediate interaction modality with low cognitive complexity, which does not require mastering new skills. The voice, at the sub-verbal level, offers the kind of instantaneous, continuous, and multidimensional control essential for real-time interactive synthesis algorithms.

Togneri, Alder, and Attikiouzel (1992) showed that articulations in speech can be represented with as few as four-dimensions nonlinearly embedded in higher dimensional spaces of heterogeneous low-level features. The embedding depends on individual vocal tracts and on the specific articulations, and parameter extraction is hampered by intrinsic noise sources. The speech space is a subset of a wider vocal articulation space, but most individuals use the vocal apparatus primarily for the
purpose of speech. Micro-modulations are always present in human voice, even when producing steady timbres with invariant articulation (Quatieri 2008). Moreover, repeated instances of perceptually identical vocal sound can be acoustically quite different (Yang, Millar, and MacLeod 1996; Stevens 1971). Finally, linear variations of the vocal tract articulations determine non-linear dynamics in the acoustic features of the vocal sound. The model we learn from user-provided examples considers the inaccuracy that human-generated data can present at interaction time.

The system does not require any specific vocal sounds to operate, but adapts to the vocal control style of users. To provide natural and intuitive interaction, the control strategy relies on the following principle: the synthesis timbre changes if and only if the instantaneous acoustic characteristics of the voice change and vice versa. We define two categories of control sound, vocal gesture and vocal posture, which are sufficient to describe the interaction for the purpose of training. A vocal gesture is a sound with dynamic (i.e. variable) acoustic characteristics intended to determine a variation of the synthesis timbre. A vocal posture is a sound with steady acoustic characteristics that leaves the synthesis timbre unchanged. Silence is a special case of posture. Vocal postures are intrinsically restricted to ‘continuant’ articulations (those that can be sustained over time) such as vowels, nasal, liquid, and affricates. The training data includes both vocal postures and vocal gestures. In the training procedure the temporal unfolding of the vocal gestures is disregarded, thus vocal sounds can be presented in different orders across training instances.

**Selection of optimal low-level acoustic features**

The next task is to identify which vocal features will be most effective for a given user and the particular gestures they plan to use to define the control they desire. The overall strategy is to start with a large number of features, including LPC coefficients, MFCCs, and Perceptual Linear Predictive (PLP) coefficients, under a variety of different
computational conditions (i.e. sampling rate, window sizes, overlap, number of coefficients), and discover which of these features is optimal. ‘Optimal’ means meeting multiple constraints, the two most important being insensitivity to noise, and the ability to capture the majority of the variation in the signal. Interface usability depends on the stability of the computed features. Vocal postures, intended to be static, contain short-time noise-like variations in the low-level features which might be propagated to the synthesizer. Also, features computed from vocal gestures and used for mapping purposes can be redundant which could undermine the browsability of the timbre space. Therefore, instead of choosing a fixed set of low-level features, as in most related works, we observe the statistical properties of a large set of features across the training data, and identify those features and their computational conditions which are best at representing the vocal data.

To determine the feature set, for each vocal posture we compute the Relative Mean Difference (RMD) for each low-level feature. The RMD is a dimensionless measurement of statistical dispersion equal to the absolute mean difference divided by the arithmetic mean, which when averaged across the whole set of training vocal postures, provides a good estimate of the sensitivity of features to noise. Only features with an average RMD below a threshold are selected, while the rest are rejected because of their significant variation when computed on postures, which by definition, are perceptually invariant vocal sounds. To measure the amount of non-redundant information embedded in the set of vocal gestures we measure the intrinsic dimensionality of the matrix containing the selected features computed on all instances of vocal gestures in the training set. We define a quality measure based on a combination of the intrinsic dimensionality and the robustness computed over a large set of computational conditions, and we choose the set of features and parameterizations that maximize the quality measure. We have observed that similar training data from different users determine different features and parameters, as does different training data from the
same user. Thus, the optimal set of low-level features is case-specific and must be adaptive to both the user and specific control style. For mathematical details see (Fasciani 2012).

**The vocal gestural controller**

Generally, the embedded dimensionality of the training data, which includes the selected low-level features computed on the provided examples of vocal gestures, is significantly lower than the number of features selected from the previous stage. The aim is to represent the vocal data with the smallest number of dimensions, and then to linearize the dynamics relating vocal gesture to low-level features. For simplicity our target space is an N-Dimensional (N-D) hypercube (e.g. a square in 2D, a cube in 3D) that in the next stage is remapped onto the specific timbre space of the synthesizer.

The exclusive use of dimensionality reduction techniques can produce non-accessible sub-regions of the original space, nonlinear response characteristics, and discontinuities between gesture and control signals, which may be detrimental for the synthesis control purpose. To address these issues we train an N-dimensional (’N-D’) Self Organizing Map (SOM) with the gestural training data reduced by Isomap (Tenenbaum, Silva, and Langford 2000), and then we use the SOM as a control structure generating the N independent control signals. These are computed by interpolating the lattice indexes of the SOM nodes closest to the vector of low-level instantaneous features of the voice. This approach provides effective control only if the trained SOM preserves the structure of the manifold embedded in the training data without distortions such as lattice twisting and edge curling. This behavior is not guaranteed by the standard SOM training algorithm. Moreover, the SOM requires choosing parameters such as the number of nodes, number of training iterations, and learning and attraction rates, which can generate distortions if not carefully chosen. In our training procedure, detailed in (Fasciani and Wyse 2013) and briefly illustrated in
Figure 2, these parameters are adapted to the size of training data. The modified training algorithm and the prior stage of dimensionality reduction pulls the vertices of the lattice towards the gestural extrema detected in the training data, avoiding folding or twisting, providing a distortion-free lattice adapted to the local distribution in the training vocal data.

![Figure 2: 2D and 3D examples of SOM lattice weights represented as Isomap components at initialization (left) and after the training (right). At initialization, the vertices of the lattice are collocated with the detected gestural extrema, and then pulled towards these points during the training to prevent folding, which could cause sonic discontinuities when mapping gestures during performance.](image)

In summary, the computation of the vocal gestural controller output from the live vocal input is performed as follows: (1) compute the low-level features from the voice signal and reject the noisy ones; (2) apply the Isomap dimensionality reduction derived from the training data; (3) compare the resulting N-D vector with the SOM lattice nodes; (4) spatially interpolate the relative indexes of the nodes using the Inverse Distance Weighting (IDW) metric and normalize to unit range. To provide smoother transitions and discontinuity-free control signals, we limit the SOM search to the $3^N$ immediate neighbors of the closest node in the previous iteration. When the input signal is absent or different from those used for training, the controller does not generate any output.
Computing and interacting with synthesis timbre spaces

At this point, the control signals computed from the voice could simply be mapped onto a few synthesis parameters. However, to provide a richer sonic interaction we propose a strategy to drive a larger number of parameters by mapping control to the synthesis timbre space, from which we derive the synthesizer parameters. Methods to interact with timbre instead of synthesis parameters have been proposed from the pioneering work of Wessel (1979). Such methods embed the perceptual characteristic of the instrument’s timbre in the control-to-parameters mapping. Arfib et al. (2002) formalized this approach in mappings that include intermediate perceptual layers, which enhances the sensitivity and efficiency of the interface, but a generic method to compute and map to synthesizer-specific perceptual spaces has not yet been proposed.

Assuming no prior knowledge of the synthesis algorithm, but assuming a deterministic response, we estimate the model relating parameters to sound by observing inputs and then ‘listening by machine’ to the sonic output. The analysis procedure includes computing the set of unique synthesis parameters vectors, which are sent one at a time to the sound generator while computing a set of perceptually related audio descriptors on the sound output. Parameter vectors and associated timbre descriptor vectors are collected in two databases, which together model the synthesis timbral response. Analyzing all possible parameter combinations with a fine resolution requires excessive time and memory. The model we compute considers only those parameters that the users intend to vary, leaving the other fixed. Moreover, we use a coarse parameter resolution in the analysis stage, which is later compensated with temporal and spatial interpolation. To provide a comprehensive study of the timbre, for each parameter combination we compute descriptors over multiple overlapping windows of the synthesized signal. These are further analyzed and processed using three different user-selected methods, detailed in (Fasciani 2016). In particular we provide techniques to analyze steady, dynamic, and decaying timbres. The analysis and
the following mapping strategy are independent of the specific descriptors used in the system. However, the current prototype uses the loudness of the 24 Bark critical bands.

**Organizing and mapping the timbre space**

After the analysis stage we have two high dimensional spaces, relating synthesis parameter vectors to sound descriptors, and we aim at using spatial coordinates of the timbre space to drive the synthesizer. The dimensionality of the timbre space is high while the output of the vocal gestural controller is at most three-dimensional. Hence we reduce the timbre space dimensionality to a number of components $N$ identical to that of the vocal gestural controller. The reduction is performed with Isomap to preserves the geodesic distance in the reduced space, providing greater accuracy in the low dimensional representation of the timbre than linear techniques (Burgoyne and McAdams 2007). Over a large set of experiments with 4 to 8 variable parameters we observed that 2 to 4 dimensions are generally sufficient to cover a significant degree of the total timbre variance. In the left column of Figure 3 there are two examples of reduced timbre spaces from two different synthesizers subject to the variation of 8 synthesis parameters. It is evident that each synthesis case presents a unique shape and distribution of data points. There are regions with either extremely low or high densities. A linear mapping of the gestural controller output onto the reduced timbre space would have strong limitations, such as spanning empty sub-regions of the timbre space which have no computable response. The spatial interpolation in high density regions increases the chance of driving the synthesizers to inaccurate timbres due to the not bijective relationship between parameters and sound. To handle such cases, we use a nonlinear mapping that project the gestural controller output onto the timbre space.

For this timbre space mapping we take a different approach than for the vocal gestural controller because the synthesis algorithm generating the sound is strictly deterministic, and, as opposed to the voice, the mapping will not be required to respond
to new data not present during training. First, we redistribute the entries in the timbre space to create a uniform distribution using an iterative technique based on the Voronoi tessellation (Nguyen et al. 2009). The tessellation provides a rigid structure that we progressively deform but do not modify, preserving the neighborhood relations of the data points (i.e. homomorphic transformation). The inverse of this redistribution process is the transformation that maps a generic control space (N-D hypercube) to the specific synthesis timbre space maximizing their relative overlap. Therefore, we use a feed forward Artificial Neural Network (ANN) to learn the inverse of the redistribution which represents the gestural controller output to timbre space mapping function. The size of the single layer ANN is gradually increased until the mapping is sufficiently accurate, measured by combining the mean squared error of the nonlinear regression and the percentages of non-accessible timbre points in a simulated use. In the right column of Figure 3 we show mapping of the uniform control space through the trained ANN, showing similar shape and distribution with the original timbre space.

The ANN maps the output of the gestural controller onto the reduced timbre space, and the synthesizer is driven with the parameter vector associated with the closest timbre data point. To cope the with the limited parameter resolution used in the analysis stage, we apply IDW interpolation. To further smooth the synthesis control, we also use linear interpolation between two consecutive parameter vectors. This synthesis control method has an intrinsic drawback in the case where multiple synthesizer parameter sets generate the same or similar sounds. Mapping similar sounds to widely varying parameter values can cause sound audible glitches during synthesis. As discussed before, this issue is also aggravated by spatial and temporal interpolation. To address this problem we restrict transitions in the timbre space to those points that minimize parameter discontinuities, dynamically locating these points that map to neighboring parameter vectors. We also allow users to define the maximum
instantaneous leap in the parameter space so that they are in control of the tradeoff between parameter continuity and the browsability of the timbre space.

![Figure 3: 2D (top row) and 3D (bottom row) examples of Isomap reduced timbre spaces related to synthesizers subject to the variation of multiple parameters (left column), with entries uniformly redistributed (center column), and the trained ANN mapping to the timbre space (right column).]

**Functional open-source prototype**

The entire system is implemented as an open-source prototype, which has been optimized to run in real-time on general-purpose personal computers, and has been used for live performances. The prototype and demonstrations are available at http://stefanofasciani.com/vci4dmi.html. The prototype is implemented in Cycling ’74 Max, with a backend engine running in MATLAB. For the runtime interface, a Max patch with a GUI provides users with options to customize the synthesis analysis, the voice preprocessing, and the runtime mapping. A separate Max patch implements the analysis of the synthesizer, and the mapping computation is performed by two offline MATLAB scripts, one for the vocal gestural controller and another for the synthesis timbre space. These generate two data structures including the mapping for the vocal control and for the sound synthesizer, which can be loaded independently to the Max
patch at runtime. This separation of functionality allows same vocal gestural controller to be used with different synthesizer and vice versa.

The procedure for setting up the system does not require technical expertise and the mapping computation is completely automated. We also provide utilities to establish communication between the system and any synthesizer hosted in digital audio workstations such as Virtual Studio Technology (VST) plug-ins using the Open Sound Control (OSC) protocol. Users are required to identify the specific synthesizer parameters they intend to control, with their respective ranges and analysis resolutions. The vocal training data is provided with audio files. There is no minimum requirement on the size of the training data, but with larger sets a more accurate model can be trained. Usually, a usable vocal gestural controller can be produced with as little as one minute of training data, which includes an equal share of vocal postures and vocal gestures. Since the system is trained using machine listening and unsupervised learning techniques, the user is freed from the burden of providing additional data that relates explicitly to the mapping. This implies that users are required to familiarize themselves with the generated mapping and we provide visual feedback of vocal and timbre spaces to support this task. Moreover, we provide options to further tune the voice-to-synthesis control at run-time without the need to repeat the training. We allow users to invert the orientation of the axes of the intermediate space mapping the output of the vocal gestural controller to the low-dimensional synthesis timbre space. In the mapping computation, there is no explicit criterion we use to set the reciprocal axis orientation. This choice determines a significant change in the response, such as changing the association between voice and synthesis timbre, without adding computational overhead. Other options include the IDW interpolation coefficient, the maximum instantaneous leap in the vocal and synthesis parameter spaces, and the possibility to directly map the output of the gestural controller to the uniformly redistributed timbre space, or to the original timbre space, bypassing the ANN mapping. These enable users
to gradually tune the system between two radically different types of vocal interactions: an explicit voice to synthesis timbre remapping, or else a mapping of voice variations to directions and depth to browse the perceptual timbre space of the sound synthesizer. In Figure 4 there are two examples of the first vocal interaction type, which is more intuitive due to the high correlation between variations of voice and synthesis timbre.

Other optional utilities we provide to enhance the interaction with the sound synthesizer include: system activation based on voice energy or on similar to training voice, pitch-tracking based note generation, and settling to a user-defined state when inactive (i.e. no input voice detected). These user-defined customizations extend the capabilities of the system, are all intuitive and done through listening, and require no understanding of the mathematics of the mappings, and no retraining computation.

![Image](image_url)

Figure 4: Two examples of vocal control of sound synthesis, with the synthesizer sound spectrogram on top and the driving vocal gesture spectrogram on bottom.

The computational performance was evaluated on a 2.4 GHz quad-core Intel i7 with 1.3 GHz bus speed, 16 GB of memory, and OSX operating system. Training time depends on the amount of vocal data and synthesis analysis settings provided by users, because the size of the SOM and ANN scale with the amount of training data. A typical case that provides sufficiently accurate and stable control uses one minute of vocal training data and about 2k synthesis parameter combinations. The resulting analysis and training time is about 45 minutes for a 2D gestural controller and timbre space, while it doubles for the 3D case.
The interface latency has two components: the time to acquire one segment of a voice signal, and the mapping computation time. The first component depends on the length of the voice segment we analyze to compute feature vectors, which in turn depends on the sampling rate and window size that are automatically selected to be their case-specific optimal values. Length values range between 10 ms and 42 ms, with 21 ms being the most commonly selected by our algorithm. The computation time, from the generation of low-level features to the transmission of the OSC message including synthesis parameters, depends on the system complexity, determined by the amount of data provided for training. On typical cases, the computational latency is approximately 1 ms for the 2D case and 2 ms for the 3D case. This enables a large window overlap and a throughput of up to 1000 synthesis parameters update per second. We measured a latency of 10 ms in a worst case 3D scenario we tested with 1728 nodes in the gestural controller, 18 and 6 neurons in the two layers of the ANN, a neighborhood graph with 3.5M entries for the nonlinear dimensionality reduction, and 15k data points in the timbre space. Memory requirements have never exceeded 3 GB in total.

**User study and evaluation**

It is not possible to define a generic strategy for the evaluation of sonic controllers or musical interfaces (Fels 2004). Methods used in this field present conflicts, discrepancies and limited scope (Marquez-Borbon et al. 2011; El-Shimy and Cooperstock 2016). The evaluations often are informal, idiosyncratic, or not performed at all (J. Barbosa et al. 2015). Traditional Human Computer Interaction (HCI) evaluation methods are only effective to compare similar systems or to assess performances on specific tasks (Orio, Schnell, and Wanderley 2001). Limiting the evaluation to a set of pre-defined tasks may restrict the view on original and creative use, obscuring the different concepts of controllability and sonic expressivity (Dobrian and Koppelman 2006). Perception of benefits and limitations may depend and aims of individual evaluators. In evaluations
centered on the creative and performative use of an interactive system, the recruitment of subjects can be an issue if the available population is small, especially when the design addresses users with specific backgrounds and expertise (Wanderley and Orio 2002).

Evaluation strategies for small sample sizes should be focused on the identification of trends and patterns, since statistical measurement or generalizations can be misleading or inconsistent. Moreover, short evaluation sessions are sufficient to assess only basic usability factors, while estimating creative and exploratory affordances requires observations over longer periods (Gelineck and Serafin 2009). The design of mapping techniques is the rational outcome of an engineering process, often driven by creative design. The final results can be subjective and criticized, but a scientific methodology to verify the initial engineering principles is still important (Wanderley and Depalle 2004). The user-centered evaluations are essential to determine how a novel interface is received and whether it enables creativity and expressivity, how it imposes or suggests new modes of thinking, interacting, and organizing time or texture in music. The adaptability of a generative mapping and its expressivity are hard to measure objectively. These strongly depends on users' intention, which may vary significantly across subjects. Expressivity depends on, and can be associated with, the range of available choices rather than with the number of controllable synthesis parameters (Clarke 1988), and it is also related to the dimensionality of the control space (Pressing 1990). Furthermore, subjective evaluation can also be strongly influenced by the specific sound synthesis selections when attempting to evaluate interactive mapping only (Gelineck and Serafin 2009). Formal user-centered evaluations do not guarantee effective interaction design, though it can certainly identify poor ones, or eventual drawbacks and limitations.

Based on these considerations the approach we take for evaluation includes both qualitative and quantitative methods. We sample a small group of participants with
specific expertise in the field. We carry out a long evaluation session, which includes free exploration of the system, use cases, and in open-ended in-depth interviews. These are analyzed to identify trends supporting or modifying our original claims, to point out limitations and possible improvements to our current system, and to aid in developing general principles for applications beyond our immediate aims.

**Method and experimental setup**

The mapping between voice and sound synthesis provided by our system depends on the mapping computed from training data, which is case specific and generated by users. Open-ended systems and frameworks are challenging to validate due to the lack of obvious evaluative criteria (Marquez-Borbon et al. 2011). Results depend on how participants interpret prototypes, which can be different than expected (Gaver et al. 2009). Our evaluation is not limited to measurements and observation on use cases, but it includes free exploration of the system, from which we attempt to identify user experience, embodied interaction, and value-sensitive design (Harrison, Tatar, and Sengers 2007) by analyzing the audio-video recordings of the sessions and activity logs. A comprehensive objective evaluation is not possible, but the design and usability principles can be verified in user studies. The qualitative evaluation is based on the analysis of individual open-ended in-depth interviews with participants we invited to use our voice-to-synthesis control system. We interview each subject before and after using the system for determining individual perspectives, and identify patterns across subjects. In both the pre- and post-usage interviews we use the same discovery-oriented strategy (Kvale 1996; Boyce and Neale 2006), asking open-ended questions that allow the interviewer to probe participant perspectives and experiences with the novel interface, and to encourage the respondents to freely answer questions using their own words. The semi-structured and conversational format of the interviews allows for unexpected digressions to divert the planned question sequence and follow the
participant’s interest or knowledge. For the quantitative evaluation, we analyze system signals and activity logs over a set of use cases performed by the participants.

To evaluate the system, we recruited musicians and performers, who are familiar with sound synthesis and experienced with interacting with these devices in creative or performing contexts. We chose an initial sample size of ten subjects, considered adequate for phenomenological research of this kind (Thomas and Pollio 2004), and had an additional six available if needed for achieving thematic redundancy in interview results (Guest, Bunce, and Johnson 2006). Participants were purposely selected to have rich knowledge about the study questions (Gubrium 2012), and come from a musically diverse range of stylistic backgrounds to provide multiple perspectives while being representative of the general potential user population (Highhouse and Gillespie 2009).

In user-studies with a non-uniformly sampled respondents, each participant may use a different ‘frame of reference’ and ‘standard of comparison’ when interpreting questions and making judgments. Therefore scaling responses to questions using Likert scales may results in striking contradictions when compared to open-ended answers (Ogden and Lo 2012). The relatively low sample size favors the ability to probe participants’ deeper and protracted use of the system, and increases the pertinence of interview data with the interface experience. A summary of participant profiles is provided in Table 1, where ‘Live Electronics’ in the ‘Instruments’ category indicates the broad category of electronic and digital musical instruments controlled by touch-based interfaces other than the piano-like keyboard.
Table 1: Profiles of participants recruited for user studies, including: age, nationality, years of experience as musician or performer, instruments played, previous experience in building musical instruments or interfaces, active contribution in sound and music computing research field. All participants were males.

<table>
<thead>
<tr>
<th>User ID</th>
<th>Age</th>
<th>Nationality</th>
<th>Years of Experience</th>
<th>Instruments</th>
<th>Instr. Builder</th>
<th>Research</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>21-29</td>
<td>India</td>
<td>6</td>
<td>Acoustic Percussion</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>21-29</td>
<td>Singapore</td>
<td>4</td>
<td>Bass / Live Electronics</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>21-29</td>
<td>China</td>
<td>12</td>
<td>Guitar with Effects</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>4</td>
<td>30-39</td>
<td>Singapore</td>
<td>5</td>
<td>Live Electronics</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>5</td>
<td>21-29</td>
<td>Singapore</td>
<td>10</td>
<td>Piano / Sax</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>6</td>
<td>50-65</td>
<td>Scotland</td>
<td>40</td>
<td>Synthesizer Keyboard</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>7</td>
<td>30-39</td>
<td>Malaysia</td>
<td>4</td>
<td>Live Electronics</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>8</td>
<td>40-49</td>
<td>Switzerland</td>
<td>8</td>
<td>Live Electronics / VJ</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>9</td>
<td>40-49</td>
<td>Canada</td>
<td>25</td>
<td>Guitar / Live Electronics</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>10</td>
<td>30-39</td>
<td>Spain</td>
<td>18</td>
<td>Live Electronics / DJ</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

The user studies were conducted in a sound-isolated room with continuous audio-video recording. To facilitate the exploration, the interaction options and tuning settings were mapped on to a labeled hardware controller with LED feedback, as shown in Figure 5. A large screen was used to display the interactive visual representation of the control and timbre spaces. We used a head-worn microphone with a hyper-cardioid polar pattern for vocal control that minimized the microphone pick up of synthesizer sound diffused from the loudspeakers and any other sound sources. This configuration was the same as we use for live performances and allows users to perform other control task with their hands. We included a keyboard and a controller in the setup for playing the synthesizers in as in a typical performance environment.

The automatic processes of synthesis sound analysis and the timbre space mapping computation are too lengthy to be included in evaluation sessions. Instead of allowing participants to choose and train the system for a single sound synthesizer, we provided a set of eight synthesizers trained in advance, including 2D and 3D timbre spaces, that the participants selected just by pressing a button on the interface on Figure 5. This allowed presenting all participants with an identical experimental condition and identical set of choices, which they freely explored and customized based on their
personal preferences. The set of synthesizers covered a broad range of sonic timbres and with various control parameters including: FM synthesis, wavetable-based synthesis, granular synthesis, virtual analog modular synthesis, and physical modeling. Moreover we also included three cases in which the parameters were controlling respectively a low pass filter, a guitar amplifier model, or a delay with reverberator applied to the output of a synthesizer with fixed parameters. The number of variable parameters ranged from 2 to 6, and the data points in the timbre spaces ranged from 388 to 12,664.

![Image](image.png)

**Figure 5:** User study experimental setup and details of the hardware controller (left) on which we mapped the options for tuning the voice-to-synthesis mapping at runtime.

**Protocol and interview guidelines**

The first interview gathered information about the profile of the participants including their relevant experience with sound synthesizers, and their interaction practices in live performance. We also asked question about limitations of their current performance setup, how they propose to overcome such existing limitations, their previous experience with voice driven systems, and how they imagine a useful vocal interaction with musical instruments. This was followed by the explanation of the principles of voice-to-synthesis timbre interaction and by a demo of the full system.
Moreover we demonstrated the individual system components by visualizing the output of the vocal gestural controller in a 2D or 3D space, and the audio-visual interaction with the timbre space through a touchpad interface.

In the next step, we collected voice samples and used them to train the personalized vocal gestural controller. A critical issue for the appropriate training and interaction is the understanding of the working principles of the vocal gesture and posture interface. We explained that postures are ‘continuant’ vocal articulations, while gestures may include any articulation including simple gliding between postures. From each user, we collected 8 postures with a length of 5 seconds each and several instances of vocal gestures with a total duration of 30 seconds. For gesture production, we guided the users to span the whole space of vocal articulations they intended to use for control.

Training participants were initially given some assistance to get familiar with the system and the options provided to tune the two components of the mapping. Then we let them freely use and explore the system and the set of pre-trained synthesizers. We engaged with participants only if requested or if major issues were evident, such as an unresponsive mapping due to a poor choice of settings. Then we asked participants to select their preferred synthesizer and mapping options to perform the three following use case tasks: (1) use 4 different vocal postures targeting 4 unique synthesis timbres, each with a duration of about 5 seconds, and repeat these in 6 non-consecutive attempts; (2) perform vocal gestures producing identical dynamic transformation of the synthesis timbre, and repeat these in 6 consecutive attempts for 2 different cases; (3) generate as many different synthesis timbres as possible by vocal interaction within 60 seconds.

A follow-up interview concluded the evaluation session. This second interview, which represents the core of the qualitative evaluation, aimed to highlight understanding, experience, and perceptions of advantages and drawbacks related to personalized voice-to-synthesis mapping. In particular we encouraged participants to
first describe and then criticize the system in the light of their expertise with other sonic interaction devices. We asked about possible improvements, modifications and applications relevant to their own performance contexts. Participants were also encouraged to discuss their impression of the control abstraction (timbre versus parameters), their progress in mastering control skills during the practice session, and the benefit of the visual feedback. We limited the total duration of each session to a maximum of 90 minutes, dedicating approximately 15 to 20 minutes for an interview before the trial and similar time or another interview afterwards, and at least 30 minutes for the free exploration of the system.

**Interview analysis and qualitative evaluation**

The qualitative evaluation of this study is based on the analysis of the recorded participant interviews. We started with a time-stamped transcription of the audio-video recordings. Then the transcription and video was further annotated, normalized into non-colloquial text, labeled by relevant topics, and organized at higher hierarchical levels in order to synthesize participant answers to the key questions (Kvale and Brinkmann 2008; Bryman 2012), while maintaining time alignment between analysis and raw data. For the analysis we worked in two directions: we look for answers to specific questions on interaction and usability, but the open-ended questions stimulated conversations in which participants freely expressed their thoughts across different topics. In order to validate and then generalize the respondents’ information, we used triangulation (Rothbauer 2008) where specific opinions and evaluation of the system were verified by multiple participants with substantially different profiles.

Although there is a clear diversity in the answers, we can identify recurrent trends from the first batch of interviews focusing on prior experience. All participants were used to control synthesizers from interfaces where individual parameters are mapped on elements, such as push buttons, faders, dials, encoders, or their equivalents.
implemented on touchscreens. Six of them used pre-programmed automations to morph synthesis timbres in live performances because of their precision and accuracy in tuning the parameters, but also because manual execution is often impractical due to the high dimensionality of the parameters space, and because of the lack of spare bandwidth for interacting with it. With only one exception, participants had not regularly used voice-driven systems to address their needs, although seven had tried at least hands-free verbal control, such as speech recognition. Their main concern expressed was the reliability of the system in noisy environments. Finally, we also noticed that participants' expertise in music technologies was strongly correlated with their willingness to use the voice for interaction with digital musical instruments and with the sophistication of their ideas to overcome existing synthesis control limitations. The less experienced participants proposed speech recognition for driving instruments with voice commands (e.g. “set parameters X to Y”). The others suggested using non-verbal characteristics of the voice, tracking and mapping low-level features onto continuous valued synthesis parameters. However, high latency and loudspeaker feedback were among the main concerns.

From the second batch of interviews we observed that all participants had a clear understanding of the mapping principles and purposes, with eight subjects also highlighting the key concept of training for adaptation. Beneficial aspects mentioned included high expressivity, intuitive and natural interaction, extra control layer, wide dynamic timbre range of the synthetic sound, smooth response, and generalizability across specific applications or contexts. Respondents considered the use of such a system as an extension to traditional musical interfaces rather than an alternative, and eight participants considered it beneficial for their specific performance context.

When discussing drawbacks and limitations, we observed diversified answers, sometime contrasting, but all suggesting usability improvements. Some problems were due to users selecting inconsistent system options, which could have been overcome by
using default settings or selecting an appropriate tuning. This suggests, as three respondents stated explicitly, that the options exposed to tune the system require simplification, especially for novice users. Fewer settings with more meaningful identifiers are necessary. Four participants expressed the need for cross-user and ready-to-use simple mappings, because controlling only a couple of extra parameters by voice already constitutes an improvement over current practices. Finally, three participants argued that adding an adaptive mapping modality that directly associates voice and synthetic timbre by their similarities would further simplify usability and learnability.

All participants described the interactive graphical representation of the vocal, control, and synthesis timbre spaces, as essential for proper learning of the customized mapping. Nine argued that once familiar with the interaction, the visual feedback was not necessary for performing, or it could be even distracting, unless presented in a minimalistic form. Nine respondents found the system easy and intuitive at first use, but seven of them were concerned about the possibility of achieving the same level of detailed control they get with traditional controllers. However, all users experienced improvements after a half an hour of free practicing, and agreed that mastering skills and becoming familiar with the different interaction modalities supported by the system are possible, and these will improve control accuracy and intimacy. Interestingly, eight subjects considered the simultaneous multi-parametric control more important than absolute control precision for performance expressivity.

One of the most significant emergent themes for vocal control of synthesizer timbre in general concerned the user experience, with seven subjects expressing the feeling of interacting directly with a sonic object, so that the experience was one of driving a synthetic timbre via their voice timbre without thinking about the synthesis algorithm involved in the process. The remaining three subjects tried to explicitly control the synthesis parameters (rather than the resulting timbre), which resulted in challenges due to the weak bijective parameter-to-timbre relationship. However, they argued that
training and using the system with the synthesizers they are familiar with would allow focusing on timbral results directly rather than parameters.

Examining the recordings of the participants freely exploring the vocal control system, we identified several recurring patterns. Nine participants started exploring the responsiveness of the system and then passed to a more systematic phase, in which they used the visual feedback to learn the generated mappings. In particular they used the control space first and then the timbre space visualizations to understand the interaction provided by each component. Four users experienced difficulties in understanding the tuning options, and did not explore all available choices. The two options more frequently used in tuning the generative mapping were the maximum instantaneous leap in the parameter space, also called interface ‘sensitivity’ by participants, and the inversion of axes orientation in the intermediate space mapping the output of the vocal gestural controller to the low-dimensional synthesis timbre space. We did not observe any correlation between the number of synthesis parameters simultaneously controlled by voice and ease or complexity of use. This suggests that the parameter space is totally transparent and hidden from users. Finally, we noticed that when the output sound of the synthesizer had a clear pitch, participants often tuned their voice to it. In some cases, this resulted in the use of vocal signals significantly different from the ones they generated for the training set, which in turn impaired the equal access to sub-regions of the control space. A design supporting multiple vocal gestural controllers, each associated with a different pitch range, could address this tendency to match voice and synthesizer pitch, as would including a preprocessing stage that tracks and converts the runtime vocal pitch to match training data.

**Quantitative evaluation**

The quantitative evaluation is based on the analysis of the configurations participants chose for vocal control of sound synthesis and on the performance metric
computed over the three use cases described in the protocol. In Table 2, we summarize the preferred synthesizer and the key mapping options that participants selected to perform the use case tasks.

Table 2: Synthesizer and key mapping options participants selected for the use cases, including, the timbre space mapping mode, parameters maximum leap, synthesis timbre analysis mode, number of controlled parameters, and number of points in the timbre space.

<table>
<thead>
<tr>
<th>User ID</th>
<th>Mapping Mode</th>
<th>Max Param. Space Leap</th>
<th>Sinth Type</th>
<th>Timbre Analysis</th>
<th>Sinth Parameters</th>
<th>Timbre Space Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>To Unif.</td>
<td>minimum</td>
<td>FM</td>
<td>steady</td>
<td>6</td>
<td>4.8K</td>
</tr>
<tr>
<td>2</td>
<td>ANN</td>
<td>0.1</td>
<td>virt. analog</td>
<td>steady</td>
<td>4</td>
<td>1.3K</td>
</tr>
<tr>
<td>3</td>
<td>ANN</td>
<td>0.2</td>
<td>guit. amp.</td>
<td>steady</td>
<td>5</td>
<td>7.2K</td>
</tr>
<tr>
<td>4</td>
<td>ANN</td>
<td>minimum</td>
<td>low pass</td>
<td>steady</td>
<td>2</td>
<td>478</td>
</tr>
<tr>
<td>5</td>
<td>To Unif.</td>
<td>minimum</td>
<td>wavetable</td>
<td>decaying</td>
<td>4</td>
<td>5.3K</td>
</tr>
<tr>
<td>6</td>
<td>ANN</td>
<td>0.5</td>
<td>virt. analog</td>
<td>steady</td>
<td>4</td>
<td>1.3K</td>
</tr>
<tr>
<td>7</td>
<td>ANN</td>
<td>0.7</td>
<td>virt. analog</td>
<td>steady</td>
<td>4</td>
<td>1.3K</td>
</tr>
<tr>
<td>8</td>
<td>ANN</td>
<td>minimum</td>
<td>virt. analog</td>
<td>steady</td>
<td>4</td>
<td>1.3K</td>
</tr>
<tr>
<td>9</td>
<td>ANN</td>
<td>minimum</td>
<td>delay &amp; rev.</td>
<td>dynamic</td>
<td>3</td>
<td>388</td>
</tr>
<tr>
<td>10</td>
<td>ANN</td>
<td>0.4</td>
<td>wavetable</td>
<td>decaying</td>
<td>4</td>
<td>11.6K</td>
</tr>
</tbody>
</table>

All participants selected a 2D gestural controller and timbre space, which is easier for first time users and provides a more effective dimensionality reduction of the synthesis control space. This also suggests that the slightly higher cognitive complexity required to interact with the 3D mapping is not worth the increase timbral control details. Eight subjects selected the ANN for mapping from the gestural controller to the timbre space, and the remaining 2 selected the alternative mapping onto the uniformly redistributed timbre space, which is slightly less accurate in term of the linearity of timbre variation, but less computationally complex and therefore has a lower response latency. As expected, no participant selected to map the gestural controller to the original timbre space, just scaled to match the ranges, due to the poor resulting usability. This is significant because it suggests that the usability of the system draws a great benefit from the generative mapping implementing the homomorphic transformations across voice and timbre heterogeneous spaces.
The values limiting the instantaneous leap of synthesis parameter space were different and spanned from the minimum to 0.7. The minimum, which is case specific and depends on the parameter resolution used in the analysis stage, only allows navigation to timbre space points which are also immediate neighbors in synthesis parameters space, resulting in lower sensitivity, but higher control accuracy.

The timbre spaces including approximately 1k to 5k data points from the synthesis analysis stage were preferred and more usable than those outside this range. Spaces with a higher number of points were selected by one participant only. As explained earlier, these may determine a more challenging interaction, especially when a small value of the maximum parameter leap is set, without providing any significant benefit to the timbre accuracy. However the participant that selected a timbre space with 11.6k data points also set a relatively relaxed limit on the instantaneous parameter leap, which is the selection we would expect since it determines a better interface response. Participants 4 and 9 selected a timbre space with fewer points. However the use of IDW interpolation hid the coarse resolution of the timbre space, which was unnoticed by participants. Seven participants selected synthesizers generating a steady timbre, for which the specific analysis mode provides the tightest coupling between vocal articulation and synthesis sound, implementing an instantaneous remapping across the sonic domains. Finally, the synthesizers presenting the widest timbre range was the most popular and it was selected by participants 2, 6, 7 and 8.

The quantitative evaluation computed over the three use case tasks aims at estimating reliability, usability, and the repeatability of the vocal control of sound synthesis. Results are illustrated in the four diagrams in Figure 6. While executing the task, participants had no feedback on accuracy, precision, or completion. In the first use case we evaluate the stability of control over vocal postures, measuring the standard deviation of the output of the gestural controller mapped onto the synthesis timbre space, and the standard deviation of the resulting synthesis parameter. Results show
small standard deviation values, also averaged over the multiple dimensions of the gestural controller output and synthesis parameter vector. Low-level noisy features were already rejected from entering the system, and the SOM-based approach contributed to improve control stability. The standard deviation values can slightly increase in the parameter space depending on specific tuning options. However, in most cases the values are still below 0.1, meaning that we successfully rejected the intrinsic noises in voice signals, and therefore vocal postures effectively determine invariant synthesis parameters.

We evaluated the repeatability of static control measuring the difference between vocal postures that were aimed at generating an identical synthesis timbre. The difference is estimated with the Euclidean distance in the timbre space, which is equal to the squared root of two in the worst case, and with the Euclidean distance in the synthesis parameter space, which is equal to the square root of the number of parameters in the worst case. Results show large values for some subjects, especially for the distance between the parameter vectors. We verified that for participant 3 the large distances were due to poor skills in repeating similar vocal postures. For participants 5 and 8, the distance was due to strict limits on the instantaneous parameter leap. Indeed, the average distances in the control space are generally lower, which suggests that the static control of timbre is more repeatable and accurate than the detailed control of parameter values, thereby providing justification for mapping control to perceptual timbre rather than synthesis parameters. Moreover, since different sets of parameters can generate similar or identical timbres, larger distances in the parameter space does not necessarily imply timbre inaccuracy.

In the second use case, we evaluate the repeatability of dynamic control, measuring the differences across instances of identical vocal gestures aiming at identical timbre variations. Dynamic time warping was used to equalize the length of the sequences generated by participants across instances. As with postural control, the differences in
the timbre space, directly controlled by the vocal gestural controller, are lower than those in the synthesis parameter space. When analyzing starting coordinates, ending coordinates, and overall shape of the timbre control path, 8 participants managed to repeat similar patterns across different trials. The trajectories are not identical but show that repeating control space trajectories is possible. As observed and explicitly stated by participants, accuracy of dynamic and static control can be mastered practicing with the same configuration for longer periods.

Finally, for the third use case we evaluate the ability to access all nodes of the SOM, which map onto different sub-regions of the timbre space. We also measured the percentage of unique parameter combinations obtained, which are computed by IDW interpolation. Results show that all participants were able to cover at least 70% of the control space. The percentages of unique parameter combinations tended to be lower and are correlated with the total number of data points and with the maximum parameter leap set by the user. The coverage of the spaces that we detected across the whole free practicing sessions was close to 100% for all participants. However, these measurements are not derived from a systematic and structured test. The diversity of the results across users is evident, and for participants 3, 4 and 8 this can be partially ascribed to insufficient familiarization with the system we observed from the session recordings. For this study, we intentionally did not recruit any performer with specific prior vocal training, such as singers, in order to demonstrate that the system can be used by those with average untrained vocal articulation skills.
Figure 6: Quantitative evaluations results: stability and repeatability for vocal postures for use case 1 (average standard deviation – top left, distance – top right); repeatability of interaction for vocal gestures for use case 2 (average difference – bottom left), space coverage of vocal gesture for use case 3 (percentage – bottom right).

Conclusion

Using the voice for controlling digital musical instruments is an emerging trend, which has interesting practical and creative applications. It also presents many challenges due to the high-dimensionality of timbral spaces and the demands for continuity, expressivity, and customizability that we find in musical contexts. We developed a generic framework for the unsupervised training of a personalized mapping between voice and synthetic sound generation, adapted to the sonic characteristics of the voice and of the synthesized sound. Unsupervised machine listening and machine learning were integrated to fully automate the generation of
mappings, while still providing instrument designers or users with the capability to further personalize the interaction. The adaptive mapping strategy maximizes the explorable sonic space of a synthesizer given a vocal control timbral domain. The resulting interaction minimizes the cognitive complexity for controlling high dimensional synthesis parameter spaces and, at the same time, minimizes possible timbre losses due to potentially low dimensional control spaces. The nonlinear and poor perceptual relevance of traditional synthesis controllers was addressed by coupling control data with sonic features rather than with synthesizer parameters themselves.

We evaluated various aspects of the vocal-to-synthesizer timbral mapping system, as well as the overall effectiveness of vocal control of synthesis with experienced performing musicians in a series of qualitative and quantitative user studies. The system was able to provide reliable and repeatable sonic control that users were able master after a short period of initial exploration and practice. The musicians achieved a satisfactory level of musical accuracy using only their voice for control. Participants responded favorably to the possibilities and benefits provided by such a control layer in performance contexts when other control channels (the hands) are fully occupied. The mapping techniques we developed would also have utility in other high-dimensional interaction contexts beyond the vocal control of synthesizers.

References


