Cárthach Ó Nuanáin, Perfecto Herrera, and Sergi Jordà
Music Technology Group
Communications Campus–Poblenou
Universitat Pompeu Fabra
Carrer Roc Boronat, 138, 08018
Barcelona, Spain
{carthach.onuanain, perfecto.herrera, sergi.jorda}@upf.edu

Rhythmic Concatenative Synthesis for Electronic Music: Techniques, Implementation, and Evaluation

Abstract: In this article, we summarize recent research examining concatenative synthesis and its application and relevance in the composition and production of styles of electronic dance music. We introduce the conceptual underpinnings of concatenative synthesis and describe key works and systematic approaches in the literature. Our system, RhythmCAT, is proposed as a user-friendly system for generating rhythmic loops that model the timbre and rhythm of an initial target loop. The architecture of the system is explained, and an extensive evaluation of the system's performance and user response is discussed based on our results.

Historically, reusing existing material for the purposes of creating new works has been a widely practiced technique in all branches of creative arts. The manifestations of these expressions can be wholly original and compelling, or they may be derivative, uninspiring, and potentially infringe on copyright (depending on myriad factors including the domain of the work, the scale of the reuse, and cultural context).

In the visual arts, reusing or adapting existing material is most immediately understood in the use of collage, where existing works or parts thereof are assembled to create new artworks. Cubist artists such as Georges Braque and Pablo Picasso extensively referenced, appropriated, and reinterpreted their own works and the works of others, as well as common found objects from their surroundings (Greenberg 1971). Collage would later serve as direct inspiration for bricolage, reflecting wider postmodernist trends towards deconstructionism, self-referentiality, and revisionism that include the practice of parody and pastiche (Lochhead and Auner 2002).

In music and the sonic arts, the natural corollary of collage came in the form of *musique concrète* (Holmes 2008), a movement of composition stemming from the experiments of Pierre Schaeffer and, later, Pierre Henry at the studios of Radiodiffusion-Télévision Française in Paris during the 1940s and 1950s (Battier 2007). In contrast to the artificially and electronically generated *elektronische Musik* spearheaded by Karlheinz Stockhausen at the West German Radio studios in Cologne, the French composers sought to conceive their works from existing recorded sound, including environmental sources like trains and speech. Seemingly unrelated and nonmusical sounds are organized in such a way that the listener discovered the latent musical qualities and structure they inherently carry.

It is important to note that in music composition general appropriation of work predates these electronic advancements of technology. In Western art music, for example, composers like Béla Bartók—himself a musicologist—have often turned to folk music for its melodies and dance music styles (Bartók 1993), and others (e.g., Claude Debussy, cf. Tamagawa 1988) became enchanted by music from other cultures, such as Javanese gamelan, studying its form and incorporating the ideas into new pieces. Quotations, or direct lifting of melodies from other composers' works, are commonplace rudiments in jazz music. Charlie Parker, for example, was known to pepper his solos with reference to Stravinsky's *Rite of Spring* (Mangani, Baldizzone, and Nobile 2006). David Metzer has compiled a good reference on appropriation and quotation music (Metzer 2003).

The modern notion of sampling stems from the advent of the digital sampler and its eventual explosion of adaptation in hip-hop and electronic music. Artists such as Public Enemy and the Beastie Boys painstakingly assembled bewildering permutations of musical samples, sound bites, and other miscellaneous recorded materials that sought to supplant

Computer Music Journal, 41:2, pp. 21–37, Summer 2017
doi:10.1162/COMJ_a_00412

the many cultural references that permeated their lyrics (Sewell 2014). Later, the influence of hip-hop production would inform the sample-heavy arrangements of jungle and drum and bass, in particular with its exhaustive rerendering of the infamous "Amen Break." John Oswald, an artist who directly challenged copyright for artistic gain, dubbed his approach "plunderphonics" and set out his intentions in a suitably subtitled essay "Plunderphonics, or Audio Piracy as a Compositional Prerogative" (Oswald 1985). Using tape-splicing techniques, he created deliberately recognizable montages of pop music, such as that by Michael Jackson, in a style that became later known as "mashups." Nowadays, artists such as Girltalk create extremely complex and multireferential mashups of popular music, harnessing the powerful beat-matching and synchronization capabilities of the modern digital audio workstation (Humphrey, Turnbull, and Collins 2013).

Although the question of originality and authorship is not in the realm of this discussion, this interesting and pertinent topic is under the scrutiny of researchers in musicology and critical studies. We encourage the reader to consult work by Tara Rodgers (2003), Paul Miller (2008), and Kembrew McLeod (2009) for a more focused discourse.

Associated research efforts in computer music, signal processing, and music information retrieval (MIR) afford us the opportunity to develop automated and intelligent systems that apply the aesthetic of sampling and artistic reuse. The term *concatenative synthesis* has been extensively used to describe musical systems that create new sound by automatically recycling existing sounds according to some well-defined set of criteria and algorithmic procedures. Concatenative synthesis can be considered the natural heir of granular synthesis (Roads 2004), a widely examined approach to sound synthesis using tiny snippets ("grains") of around 20–200 msec of sound, which traces its history back to Iannis Xenakis's theories in *Formalized Music* (Xenakis 1971). With concatenative synthesis, the grains become "units" and are more related to musical scales of length, such as notes and phrases. Most importantly, information is attached to these units of sound: crucial descriptors that allow spectral and temporal characteristics of the sound to determine the sequencing of final output.

In the following sections, we will present a thorough, critical overview of many of the key works in the area of concatenative synthesis, based on our observation that there has not been such a broad survey of the state of the art in other publications in recent years. We will compare and contrast characteristics, techniques, and the challenges of algorithmic design that repeatedly arise. For the past three years, we have been working on the European-led initiative GiantSteps (Knees et al. 2016). The broad goal of the project is the research and development of expert agents for supporting and assisting music makers, with a particular focus on producers of electronic dance music (EDM). Consequently, one of the focuses of the project has been on user analysis: thinking about their needs, desires, and skills; investigating their processes and mental representations of tasks and tools; and evaluating their responses to prototypes.

Modern EDM production is characterized by densely layered and complex arrangements of tracks making liberal use of synthesis and sampling, exploiting potentially unlimited capacity and processing in modern computer audio systems. One of our main lines of research in this context has been the investigation of concatenative synthesis for the purposes of assisting music producers to generate rhythmic patterns by means of automatic and intelligent sampling.

In this article, we present the RhythmCAT system, a digital instrument that creates new loops emulating the rhythmic pattern and timbral qualities of a target loop using a separate corpus of sound material. We first proposed the architecture of the system in a paper for the conference on New Interfaces for Musical Expression (Ó Nuanáin, Jordà, and Herrera 2016a), followed by papers evaluating it in terms of its algorithmic performance (Ó Nuanáin, Herrera, and Jordà 2016) and a thematic analysis of users' experience (Ó Nuanáin, Jordà, and Herrera 2016b). This article thus represents an expanded synthesis of the existing literature, our developments motivated by some detected shortcomings, and the illustration of an evaluation strategy.

State of the Art in Concatenative Synthesis

Other authors have previously provided insightful summaries of research trends in concatenative synthesis (e.g., Schwarz 2005; Sturm 2006). These surveys are over ten years old, however (but see Schwarz 2017 for a continuously updated online survey), so we offer here a more recent compendium of state-of-the-art systems as we see them, based on our investigations of previous publications up until now.

Before music, concatenative synthesis enjoyed successful application in the area of speech synthesis; Hunt and Black (1996) first reported a unit selection scheme using hidden Markov models (HMMs) to automatically select speech phonemes from a corpus and combine them into meaningful and realistic sounding sentences. Hidden Markov models extend Markov chains by assuming that "hidden" states output visible symbols, and the Viterbi algorithm (Rabiner 1989) can return the most probable sequence of states given a particular sequence of symbols. In concatenative synthesis, the maximum probabilistic model is inverted to facilitate minimal cost computations.

The *target cost* of finding the closest unit in the corpus to the current target unit becomes the emission probability, with the *concatenation cost* representing the transition probability between states. The Viterbi algorithm thus outputs indices of database units corresponding to the optimal state sequence for the target, based on a linear combination of the aforementioned costs. Diemo Schwarz (2003) directly applied this approach for musical purposes in his Caterpillar system.

Schwarz notes, however, that the HMM approach can be quite rigid for musical purposes because it produces one single optimized sequence without the ability to manipulate the individual units. To address these limitations, he reformulates the task into a constraint-satisfaction problem, which offers more flexibility for interaction. A constraint-satisfaction problem models a problem as a set of variables, values, and a set of constraints that allows us to identify which combinations of variables and values are violations of those constraints, thus allowing us to quickly reduce large portions of the search space (Russell and Norvig 2009).

Zils and Pachet (2001) first introduced constraint satisfaction for concatenative synthesis in what they describe as musical mosaicking—or, to use their portmanteau, *musaicing.* They define two categories of constraints: *segment* and *sequence* constraints. Segment constraints control aspects of individual units (much like the target cost in an HMM-like system) based on their descriptor values. Sequence constraints apply globally and affect aspects of time, continuity, and overall distributions of units. The constraints can be applied manually by the user or learned by modeling a target. The musically tailored "adaptive search" algorithm performs a heuristic search to minimize the total global cost generated by the constraint problem. One immediate advantage of this approach over the HMM is the ability to run the algorithm several times to generate alternative sequences, whereas the Viterbi process always outputs the most optimal solution.

A simpler approach is presented in MatConcat (Sturm 2004), using feature vectors comprising six descriptors and computing similarity metrics between target units and corpus units. Built for the MATLAB environment for scientific computing, the interface is quite involved, and the user has control over minute features such as descriptor tolerance ranges, relative descriptor weightings, as well as window types and hop sizes of output transformations. On Sturm's Web site are short compositions generated by the author using excerpts from a Mahler symphony as a target, and resynthesized using various unrelated sound sets, for instance, pop vocals, found sounds, and solo instrumental recordings from saxophone and trumpet (www.mat.ucsb.edu/~b.sturm/music/CVM.htm).

As concatenative synthesis methods matured, user modalities of interaction and control became more elaborate and real-time operations were introduced. One of the most compelling features of many concatenative systems is the concept of the *interactive timbre space.* With the release of CataRT (Schwarz et al. 2006), these authors provided an interface that arranges the units in an interactive two-dimensional timbre space. The arrangement

of these units is according to a user-selectable descriptor on each axis. Instead of using a target sound file to inform the concatenation procedure, the user's mouse cursor becomes the target. Sounds that are within a certain range of the mouse cursor are sequenced according to some triggering options (one-shot, loop, and—most crucially—with real-time output).

Bernardes takes inspiration from CataRT and from Tristan Jehan's Skeleton (Jehan 2005) to build his EarGram system for the Pure Data (Pd) environment (Bernardes, Guedes, and Pennycook 2013). Built on top of William Brent's excellent feature-extraction library timbreID (Brent 2010), it adds a host of interesting features for visualization and classification. For instance, as well as the familiar waveform representation and previously described 2-D timbre representation (with various clustering modes and dimensionality-reduction implementations), there are similarity matrices that show the temporal relations in the corpus over time. Some unique playback and sequencing modes also exist, such as the *infiniteMode*, which generates endless playback of sequences, and the *soundscapeMap*, which features an additional 2-D control of parameters pertaining to sound scene design. Another system that adapts a 2-D timbre space is AudioGarden by Frisson, Picard, and Tardieu (2010), which offers two unique mapping procedures. The first of these, "disc" mode, places units by assigning the length of the audio file to the radius of the unit from the center, with the angle of rotation corresponding to a principal component of timbre, mel-frequency cepstrum coefficients (MFCCs). In the other mode, called "flower" mode, a point of the sound is positioned in the space according to the average MFCCs of the entire sound file. Segments of the particular sound are arranged in chronological fashion around this center point.

There have been some concatenative systems tailored specifically with rhythmic purposes in mind. Pei Xiang proposed Granuloop for automatically rearranging segments of four different drum loops into a 32-step sequence (Xiang 2002). Segmentation is done manually, without the aid of an onset detector, using the Recycle sample editor from Propellerhead Software. Segmented sounds are compared using the inner product of the normalized frequency spectrum, supplemented with the weighted energy. These values become weights for a Markov-style probability transition matrix. Implemented in Pd, the user interacts by moving a joystick in a 2-D space, which affects the overall probability weightings determining which loop segments are chosen. The system presents an interesting approach but is let down by its lack of online analysis. Ringomatic (Aucouturier and Pachet 2005) is a real-time agent specifically tailored for combining drum tracks, expanding on many of the constraint-based ideas from their prior musaicing experiments. They applied the system to real-time performance following symbolic feature data extracted from a human MIDI keyboard player. They cite, as an example, that a predominance of lower-register notes in the keyboard performance applies an inverse constraint that creates complementary contrast by specifying that high-frequency heavy cymbal sounds should be concatenated.

As demonstrated in EarGram, concatenative synthesis has been considered useful in sound design tasks, allowing the sound designer to build rich and complex textures and environments that can be transformed in many ways, both temporally and timbrally. Cardle, Brooks, and Robinson (2003) describe their Directed Sound Synthesis software as a means of providing sound designers and multimedia producers a method of automatically reusing and synthesizing sound scenes in video. Users select one or more regions of an existing audio track and can draw probability curves on the timeline to influence resynthesis of these regions elsewhere (one curve per region). Hoskinson and Pai (2001), in a nod to granular synthesis, refer to the segments used in their Soundscapes software as "natural grains," and they seek to synthesize endless streams of soundscapes. The selection scheme by which segments are chosen is based on a representation of each segment as a transition state in a Markov chain. Its interface features knobs and sliders for interactively controlling gain and parameters of multiple samples. To evaluate the platform they conducted an additional study (Hoskinson and Pai 2007) to reveal whether listening subjects found the concatenated sequences convincing compared with genuinely recorded soundscapes.

Figure 1. Block diagram of functionality in the RhythmCAT system.

More-specific and applied-use cases of concatenative synthesis include work by Ben Hackbarth, who explores the possibilities of concatenative synthesis in large-scale music composition (Hackbarth, Schnell, and Schwarz 2011). Hackbarth has worked with Schwarz to provide an alternative interface for exploring variations based on a force-directed graph. John O'Connell describes a graphical system for Pd that demonstrates the use of higher-level perceptual concepts like mood (happy versus sad) for informing selection in audio mosaics (O'Connell 2011).

Commercial implementations also exist for concatenative synthesis. Of particular note is Steinberg's Loopmash, a software plug-in and mobile application for automatically creating mashups from existing looped content (www.steinberg.net/loopmash). The interface consists of a number of tracks in a timeline arrangement. One track is set as a master, and slices in the master are replaced with matching slices from the other slave tracks. Users interact by manipulating "similarity gain" sliders that control the influence of each track in the slice selection algorithm. Other applications exist more as MIDI sampler systems attempting to model the performance qualities of natural sources such as orchestral ensembles (e.g., SynfulOrchestra, www.synful.com) or the human voice (e.g., Vocaloid, www.vocaloid.com).

There are many other concatenative systems that are too numerous to discuss in detail here. We have, however, compiled a table in a previous publication summarizing all the systems we have come across in our research, with remarks on interaction and visualization features, support for rhythm, and whether any user evaluation was carried out (Ó Nuanáin, Jordà, and Herrera 2016b).

Design and Implementation

In this section, we will describe our implementation of the RhythmCAT system, beginning with an explanation of the musical analysis stages of onset detection, segmentation, and feature extraction. This is followed by an examination of the interactive user interface and the pattern-generation process.

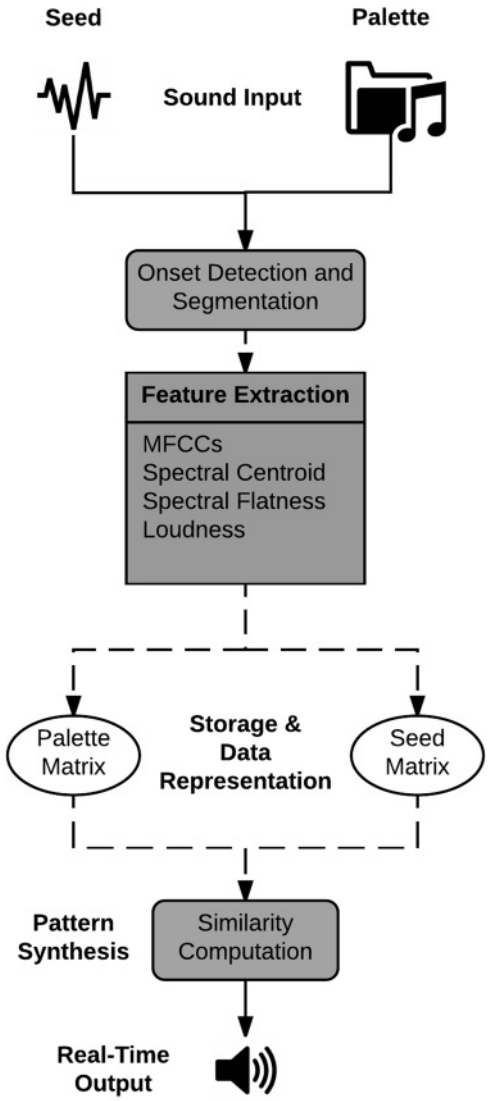


Figure 1 gives a diagrammatic overview of these important stages, which can be briefly summarized as:

1. Sound Input
2. Onset Detection and Segmentation
3. Audio Feature Extraction
4. Storage and Data Representation
5. Pattern Synthesis
6. Real-Time Audio Output

The system is developed in C++ using the JUCE framework (www.juce.com), the Essentia musical analysis library (Bogdanov et al. 2013), and the OpenCV computer vision library for matrix operations (Bradski 2000).

Sound Input

The first stage in building a concatenative music system generally involves gathering a database of sounds from which selections can be made during the synthesis procedure. This database can be manually assembled, but in many musical cases the starting point is some user-provided audio that may range in length from individual notes to phrases to complete audio tracks.

The two inputs to the system are the *sound palette* and the *seed sound*. The sound palette refers to the pool of sound files we want to use as the sample library for generating our new sounds. The seed sound refers to the short loop that we wish to use as the similarity target for generating those sounds. The final output sound is a short (one to two bars) loop of concatenated audio that is rendered in real time to the audio host.

Onset Detection and Segmentation

In cases where the sounds destined for the sound palette exceed note or unit length, the audio needs to be split into its constituent units using onset detection and segmentation.

Onset detection is a large topic of continuous study, and we would encourage the reader to examine the excellent review of methods summarized by Simon Dixon (2006). Currently, with some tuning of the parameters, Sebastien Bock's Superflux algorithm represents one of the best-performing state-of-the-art detection methods (Böck and Widmer 2013). For our purposes, we have experienced good results with the standard onset detector available in Essentia, which uses two methods based on analyzing signal spectra from frame to frame (at a rate of around 11 msec). The first method involves estimating the high-frequency content in each frame (Masri and Bateman 1996) and the second method involves estimating the differences of phase and magnitude between each frame (Bello and Daudet 2005).

The onset detection process produces a list of onset times for each audio file, which we use to segment into new audio files corresponding to unit sounds for our concatenative database.

Audio Feature Extraction

In MIR systems, the task of deciding which features are used to represent musical and acoustic properties is a crucial one. It is a trade-off between choosing the richest set of features capable of succinctly describing the signal, on the one hand, and the expense of storage and computational complexity, on the other. When dealing specifically with musical signals, there are a number of standard features corresponding roughly to certain perceptual sensations. We briefly describe the features we chose here (for a more thorough treatment of feature selection with relation to percussion, see Herrera, Dehamel, and Gouyon 2003; Tindale et al. 2004; and Roy, Pachet, and Krakowski 2007).

Our first feature is the loudness of the signal, which is implemented in Essentia according to Steven's Power Law, namely, the energy of the signal raised to the power of 0.67 (Bogdanov et al. 2013). This is purported to be a more perceptually effective measure for human ears. Next, we extract the spectral centroid, which is defined as the weighted mean of the spectral bins extracted using the Fourier transform. Each bin is then weighted by its magnitude.

Perceptually speaking, the spectral centroid relates mostly to the impression of the brightness of a signal. In terms of percussive sounds, one would expect the energy of a kick drum to be more concentrated in the lower end of the spectrum and hence have a lower centroid than that from a snare or crash cymbal.

Another useful single-valued spectral feature is the spectral flatness. It is defined as the geometric mean of the spectrum divided by the arithmetic mean of the spectrum. A spectral flatness value of 1.0

means the energy spectrum is flat, whereas a value of 0.0 would suggest spikes in the spectrum indicating harmonic tones (with a specific frequency). The value intuitively implies a discrimination between noisy or inharmonic signals and signals that are harmonic or more tonal. Kick-drum sounds (especially those generated electronically) often comprise quite a discernible center frequency, whereas snares and cymbals are increasingly broadband in spectral energy.

Our final feature is MFCCs. These can be considered as a compact approximation of the spectral envelope and is a useful aid in computationally describing and classifying the timbre of a signal. It has been applied extensively in speech processing, genre detection (Tzanetakis, Essl, and Cook 2001), and instrument identification (Loughran et al. 2004). The computation of MFCCs, as outlined by Beth Logan (2000), is basically achieved by computing the spectrum, mapping the result into the more perceptually relevant mel scale, taking the log, and then applying the discrete cosine transform.

It is difficult to interpret exactly what each of the MFCC components mean, but the first component is generally regarded as encapsulating the energy. Because we are already extracting the loudness using another measure, we have discarded this component in our system. For detailed explanations and formulae pertaining to the features introduced here, as well as others, we direct the reader to Geoffroy Peeters's compendium (Peeters 2004).

Storage and Data Representation

Further on in this article we will describe in greater detail how the seed or target audio signal is actually received from the Virtual Studio Technology host, but in terms of analysis on that seed signal, the process is the same as before: onset detection and segmentation followed by feature extraction.

The resulting feature vectors are stored in two matrices: the palette matrix and the target matrix. The palette matrix stores the feature vectors of each unit of sound extracted from the sound palette, and the target matrix similarly stores feature vectors of units of sound extracted from the seed loop.

Pattern Synthesis and Real-Time Audio Output

This section details the visible, aural, and interactive elements of the system as they pertain to the user. Figure 2 provides a glimpse of the user interface in a typical pattern generation scenario.

Workflow

The layout of the interface was the result of a number of iterations of testing with users who, while praising the novelty and sonic value of the instrument, sometimes expressed difficulty understanding the operation of the system. One of the main challenges faced was how best to present the general workflow to the user in a simple and concise manner. We decided to represent the flow of the various operations of the software emphatically by using a simple set of icons and arrows, as seen in Figure 2a.

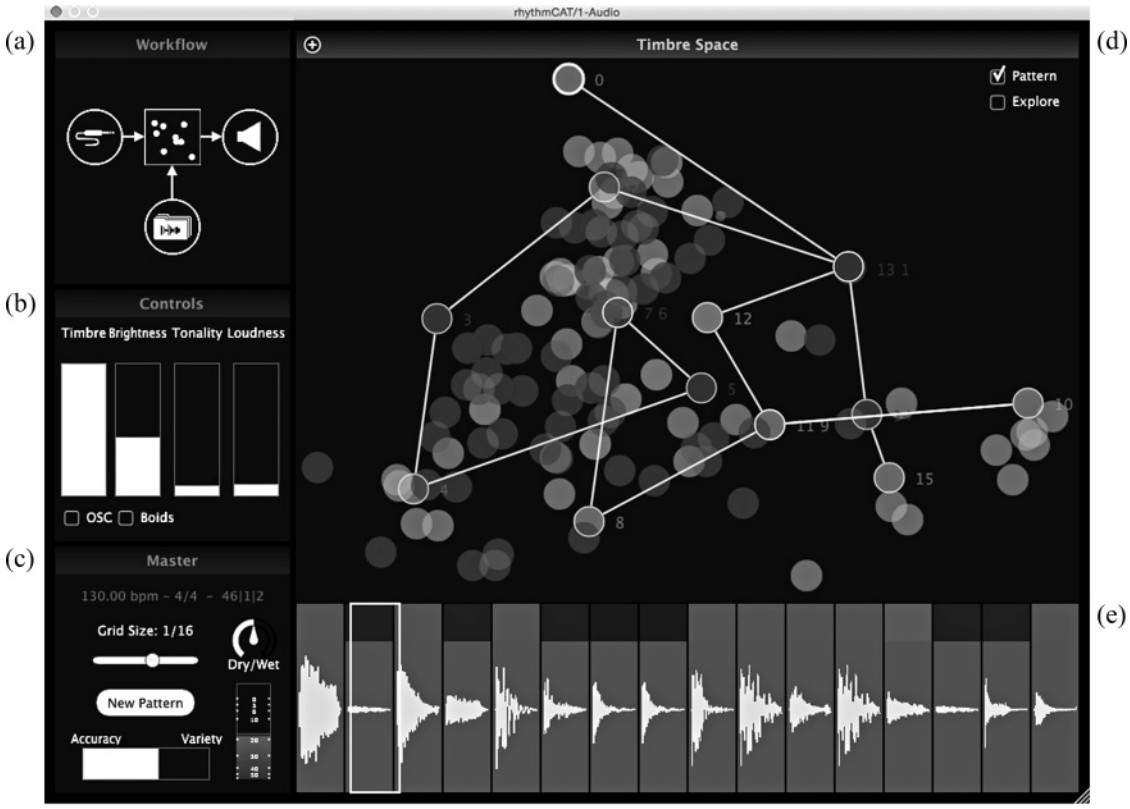
The icons indicate the four main logical operations that the user is likely to implement, and opens up related dialog screens:

Palette Dialog – indicated by the folder icon
Seed Dialog – indicated by the jack cable icon
Sonic Dialog – indicated by the square feature space icon
Output Dialog – indicated by the speaker icon

Sound Palette

The user loads a selection of audio files or folders containing audio files that are analyzed to create the sound palette, as has previously been discussed. Next, dimensionality reduction is performed on each feature vector of the units in the sound palette using principal component analysis (PCA). Two PCA components are retained and scaled to the visible area of the interface to serve as coordinates for placing a circular representation of the sound in two-dimensional space. We call these visual representations, along with their associated audio content, *sound objects*. They are clearly visible in the main Timbre Space window, Figure 2d.

Figure 2. The main user interface for RhythmCat consists of panels for workflow (a), slider controls (b), master controls (c), the main timbre space interface (d), and waveform representation (e).



Seed Input

Seed audio is captured and analyzed by directly recording the input audio of the track on which the instrument resides in the audio host. Using the real-time tempo and information about bar and beat position provided by the host, the recorder will wait until the next measure starts to begin capture and will only capture complete measures of audio. This audio is analyzed as before, with one exception. Because the goal of the instrument is to integrate with an existing session and generate looped material, we assume that the incoming audio is quantized and matches the tempo of the session. Thus, onset detection is not performed on the seed input; instead, segmentation takes place at the points in time determined by the grid size (lower left of the screen).

An important aspect to note: Because the instrument fundamentally operates in real time, we need to be careful about performing potentially time-consuming operations, such as feature extraction, when the audio system is running. Thus, we perform the audio-recording stage and feature-extraction process on separate threads, so the main audio-playback thread is uninterrupted. This is separate to yet another thread that handles elements of the user interface.

Sonic Parameters

Clicking on the square sonic icon in the center of the workflow component opens up the set of sliders shown in Figure 2b, which allows us to adjust the weights of the features in the system. Adjusting these weights has effects in terms of the pattern-generation process but also in the visualization. Presenting their technical names (centroid, flatness, and MFCCs) would be confusing

for the general user, so we relabeled them with what we considered the most descriptive subjective terms. With the pattern-generation process, these weights directly affect the features when performing similarity computation and unit selection, as we will see in the next section. Depending on the source and target material, different combinations of feature weightings produce noticeably different results. Informally, we have experienced good results using MFCCs alone, for example, as well as combinations of the flatness and centroid. In terms of visualization, when the weights are changed, dimensionality reduction is reinitiated and, hence, positioning of the sound objects in the timbre space changes. Manipulating these parameters can help disperse and rearrange the sound objects for clearer interaction and exploration by the user in addition to affecting the pattern generation process.

Once the palette and seed matrices have been populated, a similarity matrix between the palette and seed matrix is created. Using the feature weightings from the parameter sliders, a sorted matrix of weighted Euclidean distances between each onset in the target matrix and each unit sound in the palette matrix is computed.

Unit Selection and Pattern Generation

The algorithm for unit selection is quite straightforward. For each unit i in the segmented target sequence (e.g., a 16-step sequence) and each corpus unit j (typically many more), the target unit cost $C_{i,j}$ is calculated by the weighted Euclidean distance of each feature k.

These unit costs are stored in similarity matrix M. Next we create a matrix M' of the indices of the elements of M sorted in ascending order. Finally, a concatenated sequence can be generated by returning a vector of indices I from this sorted matrix and playing back the associated sound file. To retrieve the closest sequence V_0 one would only need to return the first row.

Returning sequence vectors as rows of a sorted matrix limits the number of possible sequences to the matrix size. This can be extended if we define a similarity threshold T and return a random index between 0 and $j - T$ for each step i in the new sequence.

Figure 3. Algorithm for generating a list of sound connections.

```
Procedure GET-ONSET-LIST
     for n in GridSize do
          R = Random number 0 < Variance
          I = Index from Row R of Similarity
              Matrix
          S = New SoundConnection
          S->SoundUnit = SoundUnit(I)
          Add S to LinkedList
     end for
return LinkedList
End Procedure
```

When the user presses the New Pattern button (Figure 2c), a new linked list of objects, called *sound connections*, is formed. This represents a traversal through connected sound objects in the timbre space. The length of the linked list is determined by the grid size specified by the user, so if the user specifies, for example, a grid size of 1/16, a one-measure sequence of 16th notes will be generated. The algorithm in Figure 3 details the exact procedure whereby we generate the list. The variance parameter affects the threshold of similarity by which onsets are chosen. With 0 variance, the most similar sequence is always returned. This variance parameter is adjustable from the Accuracy/Variety slider in the lower-left corner of the instrument (Figure 2c).

In the main timbre space interface (Figure 2d), a visual graph is generated in the timbre space by traversing the linked list and drawing line edges connecting each sound object pointed to by the sound connection in the linked list. In this case, a loop of 16 onsets has been generated, with the onset numbers indicated beside the associated sound object for each onset in the sequence. The user is free to manipulate these sound connections to mutate these patterns by touching or clicking on the sound connection and dragging to another sound object. Multiple sound connections assigned to an individual sound object can be selected as a group by slowly double-tapping and then dragging.

On the audio side, every time there is a new beat, the linked list is traversed. If a sound connection's onset number matches the current beat, the corresponding sound unit is played back. One addition

that occurred after some user experiments with the prototype is the linear waveform representation of the newly generated sequence (Figure 2e). Users felt the combination of the 2-D interface with the traditional waveform representation made the sequences easier to navigate and they also welcomed being able to manipulate the internal arrangement of sequence itself once generated.

Evaluation

In the course of our literature review of the state of the art, we were particularly interested in examining the procedures and frameworks used in performing evaluations of the implemented systems. Our most immediate observation was that evaluation is an understudied aspect of research into concatenative systems. With creative and generative systems, this is often the case; many such systems are designed solely with the author as composer in mind.

Some authors provide examples of use cases (Cardle, Brooks, and Robinson 2003). Authors, such as Sturm, have made multimedia examples available on the Web (see Zils and Pachet 2001; Xiang 2002; Sturm 2004). Frequently, researchers have made allusions to some concept of the "user," but only one paper has presented details of a user experiment (Aucouturier and Pachet 2005). One researcher, Graham Coleman, also highlighted this lack of evaluation strategies in concatenative synthesis in his doctoral dissertation (Coleman 2015). For the evaluation of his own system, he undertook a listening experiment with human participants in tandem with a thorough analysis of algorithmic performance and complexity.

We conducted extensive evaluation of our own system, both quantitatively and qualitatively. In the quantitative portion, we set out to investigate two key aspects. First, if we consider the system as a retrieval task that aims to return similar items, how accurate and predictable is the algorithm and its associated distance metric? Second, how does this objective retrieval accuracy correspond to the perceptual response of the human listener to the retrieved items?

The qualitative evaluation consisted of interactive, informal interviews with intended users—mostly active music producers but also music researchers and students—as they used the software. We gathered their responses and impressions and grouped them according to thematic analysis techniques. As alluded to in the introduction, both the quantitative evaluation and the qualitative evaluation have been previously reported in separate publications, but we include summaries of each here for reference.

System Evaluation

We describe here the qualitative portion of the evaluation, first by introducing the experimental setup, then presenting and comparing the results of the algorithm's retrieval accuracy with the listener survey.

Experimental Setup

Because the goal of the system is the generation of rhythmic loops, we decided to formulate an experiment using breakbeats (short drum solos taken from commercial funk and soul stereo recordings). Ten breakbeats were chosen in the range 75–142 bpm, and we truncated each of them to a single bar in length. Repeating ten times for each loop, we selected a single loop as the target seed and resynthesized it using the other nine loops (similar to holdout validation in machine learning) at four different distances from target to create 40 variations.

Each of the loops was manually labeled with the constituent drum sounds as we hear them. The labeling used was "K" for kick drum, "S" for snare, "HH" for hi-hat, "C" for cymbal, and "X" when the content was not clear (such as artifacts from the onset-detection process or some spillage from other sources in the recording). Figure 4 shows the distribution labels in the entire data set and the distribution according to step sequence. We can notice immediately the heavy predominance of hi-hat sounds, which is typical in kit-based drumming patterns. In addition, the natural trends

Figure 4. Distribution of sound labels in the source corpus (a). Distribution of sound labels by step number in the 16-step sequence (b).

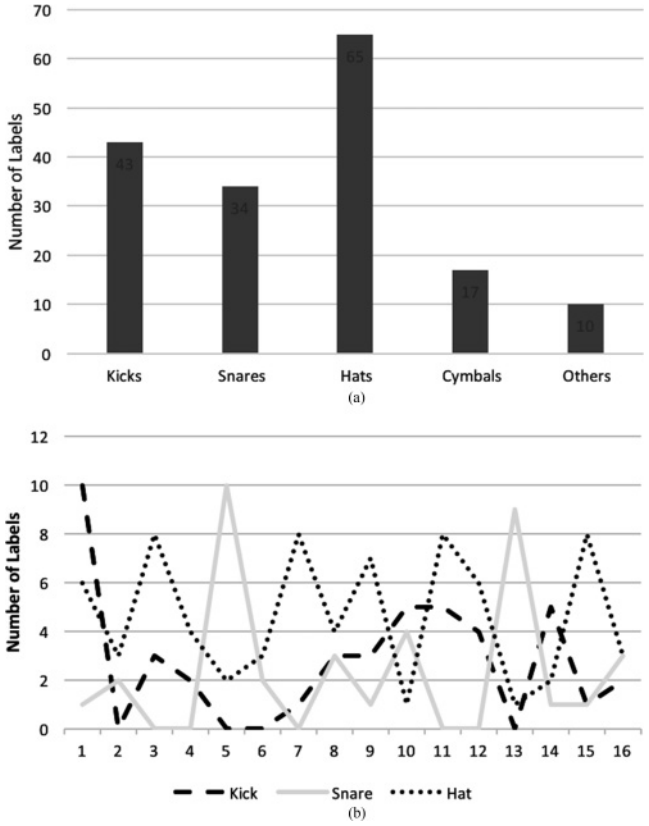
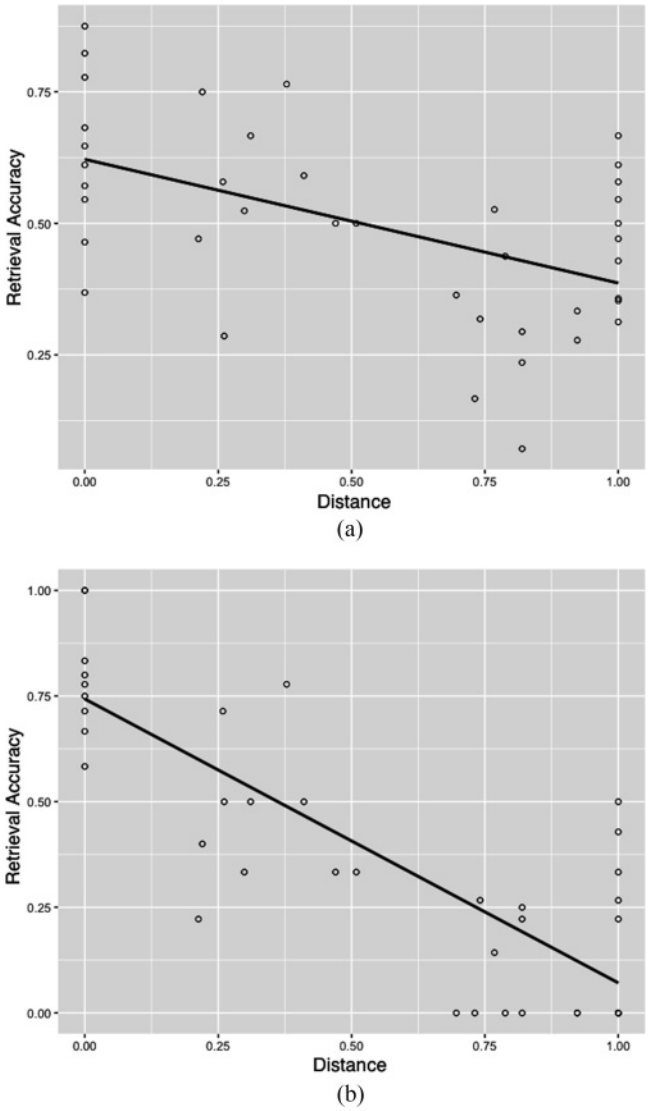


Figure 5. Scatter plot and linear regression of accuracy versus distance for all sound labels (a) and for the same sequence with kick drum and snare isolated (b).



of kit drumming are evident, namely, kick-drum placement on the first beat and offbeat peaks for the snares.

Retrieval Evaluation

We compared each of the labels in each 16-step position of the quantized target loop with the labels in each step of the newly generated sequences. The accuracy A of the algorithm is then given by the number of correctly retrieved labels divided by the total number of labels in the target loop, inspired by a similar approach adopted by Thompson, Dixon, and Mauch (2014).

Based on Pearson's correlation of the retrieval ratings and the distances of the generated patterns, we were able to confirm the tendency of smaller distances to produce more similar patterns in terms of the labeling accuracy. A moderate negative correlation of $r = -0.516$ (significance level $p < 0.001$) is visible by the regression line in Figure 5a. If we isolate the kick and snare (often considered the more salient events in drum performances, see Gouyon, Pachet, and Delerue 2000) the negative correlation value decreases sharply to $r = -0.826$, as shown in Figure 5b.

Listener Evaluation

Observing that the algorithm tends to reproduce labels in a predictable fashion, we sought to establish whether this conforms in reality to what a human listener perceives. An online listener survey was conducted using the same generated loops and targets from the retrieval evaluation. Twenty-one participants completed the survey, drawn mostly from music researchers and students from the institutions of Universitat Pompeu Fabra and the Escola Superior de Música de Catalunya in Barcelona, as well as friends with an interest in music. Twenty out of those indicated that they played an instrument, with nine specifying an instrument from the percussion family.

The participants were requested to audition a target loop and each of the generated loops in succession. They were then asked to rate, on a Likert scale of 1 to 5, the similarity of the generated loop to the target in terms of their timbre (i.e., do the kick drums, snares, and hi-hats sound alike?) as well as the rhythmic structure of the pattern (i.e., is the arrangement and placement of the sounds similar?). We also asked them to rate their aesthetic preference for the generated patterns, to determine any possible correlation with similarity.

Survey results were collated and analyzed using Spearman's rank correlation, comparing the mode of the participants' responses with the distance value of each loop. A moderate-to-strong negative correlation pattern emerged for all of the variables under consideration, namely, $r = -0.66$ for pattern similarity, $r = -0.59$ for timbral similarity, and $r = -0.63$ for their personal preference according to similarity (with significance levels of $p < 0.01$ in all instances). It should be evident that the listeners' judgments reflect what the results unearthed in the retrieval evaluation.

User Evaluation

The quantitative evaluation demonstrated the predictive performance of the algorithm based on retrieval accuracy and the corresponding listeners' judgments of similarity and likeness. Equally deserving of evaluative scrutiny is the users' experience of working with the software: gauging their responses to the interface, its modes of interactions, and its relevance and suitability for their own compositional styles and processes.

To this effect, a qualitative evaluation phase was arranged to gather rich descriptive impressions from related groups of users in Barcelona and Berlin during February 2016. In Barcelona, as with user profiles of the listener survey, most of the participants were researchers or students in the broad area of Sound and Music Computing. In Berlin we were able to gain access to artists involved in the Red Bull Music Academy as well as with employees of the music software company Native Instruments.

In broad terms, the overall sense of people's impressions was positive. Many participants were initially attracted to the visual nature of the software and were curious to discover its function and purpose. After some familiarization with its operation, people also remarked positively on its sonic output and ability to replicate the target loop:

> "It's an excellent tool for making small changes in real time. The interface for me is excellent. This two-dimensional arrangement of the different sounds and its situation by familiarity, it's also really good for making these changes."
>
> "I'm really interested in more-visual, more-graphical interfaces. Also, the fact that you can come up with new patterns just by the push of a button is always great."
>
> "It's inspiring because this mix makes something interesting still, but also I have the feeling I can steal it."
>
> "The unbelievable thing is that it can create something that is so accurate. I wouldn't believe that it's capable of doing such a thing."

Some of the negative criticism came from the prototypical nature of the instrument, and some users were not comfortable with its perceived indeterminacy:

"It was too intense and also [had] a prototype feeling. So I was like, 'Well, it's cool and very interesting but not usable yet.' "

"Right now it's still hard to find your way around, but that's something you can refine pretty easily."

Usage Scenarios

Participants were asked to consider how they would envisage themselves using the software. Most of them concurred that its strength would be in supporting musicians in their production and compositional workflows. Some users were curious about using it in live contexts, such as continuous analysis of instrumental performance or beat-boxing assistance:

"This is great! Ah, but wait . . . Does it mean I could, like, beat box really badly some idea that I have . . . and then bring my samples, my favorite kits and then it will just work?"

Continuous recording and analysis is within the realm of possibility, but can potentially be an operation that is prohibitively computationally expensive, depending on the granularity of the beat grid and the size of the corpus. Further benchmarking and tests are required to establish the upper bounds.

Another interesting observation was that many users did not want to start with a target, preferring to use the instrument as a new, systematic method of exploring their existing sounds:

"I've got this fully on wet straight away, which tells you the direction I'd be going with it."

"You just want to drag in a hundred different songs and you just want to explore without having this connection to the original group. Just want to explore and create sound with it."

Traditional Forms of Navigation

Our original intention was for users to solely be able to arrange their patterns through the 2-D timbre space. Through the course of our discussions with users we learned that, although they were eager to adapt the new visual paradigm, they still felt the need for a linear waveform to aid their comprehension. Because of this feedback, the waveform view was implemented early on in our development, as is evident in its inclusion in Figure 2.

"It's a bit hard to figure out which sixteenth you are looking for, because you are so used to seeing this as a step grid."

'You have a waveform or something. . . Then I know, okay, this is the position that I'm at."

"Is there also a waveform place to put the visualization? People are so used to having that kind of thing."

Shaping Sounds

A recurring issue, which cropped up mainly with producers and DJs, was the desire to shape, process, and refine the sounds once a desirable sequence was generated by the system. This way of composing seems emblematic of electronic music producers across the board; they start with a small loop or idea then vary and develop it exploiting the many effects processing and editing features provided by their tools. Most crucially, they desired the option to be able to control the envelopes of the individual units via drawable attack and decay parameters, which is currently being implemented.

". . . an attack and decay just to sort of tighten it up a little bit . . . get rid of some of the rough edges of the onsets and offsets."

"It would be great if you could increase the decay of the snare, for example. Which, if it's prototype, you can't expect to have all those functions there immediately, but in an end product, I think it would be a necessity."

Parameterization and Visualization

The most overarching source of negative criticism from all users was in how we presented the parameters of the system. Users are freely able to manipulate the individual weightings of the features, affecting their relative influence in

the similarity computation, but also in the PCA dimensional-reduction stage. In an effort to make this more "user friendly," we relabeled the feature names with more generally comprehensible terms like "timbre," "brightness," "harmonicity," and "loudness." Despite this, participants reported being confused and overwhelmed by this level of control, stating that they were a "a bit lost already," that there are "four parameters, and you don't know which thing is what," and that they "would prefer not to have too many controls."

Most users were quite content with the overall sonic output from the system without delving into the manipulation of feature parameters. For the visualization, however, there are certain configurations of the features that produce the best separation and clustering of the units (although MFCCs alone appear to be the most robust in our experience).

One option we are actively investigating would be to remove these parameter sliders and replace them with an optional "advanced" mode, giving users the ability to select specific parameters for the axes (as in CataRT) in addition to "automatic" arrangement configurations made possible by using dimensionality-reduction techniques. These configurations could be derived by analyzing different sound sets to find weighting combinations that give the best visual separation, depending on the corpus provided. Finally, we are currently using PCA for dimensionality reduction. There are also other approaches, including multidimensional scaling (Donaldson, Knopke, and Raphael 2007) and the recent t-distributed stochastic neighbor embedding algorithm (Frisson 2015; Turquois et al. 2016), which have been used in musically related tasks and that we are implementing and evaluating as alternatives.

Discussion

Evaluating systems for music creation and manipulation is a difficult, ill-defined, and insufficiently reported task. As we have stressed in the course of this article, this is also the case with systems for concatenative synthesis. After conducting our own evaluation, we considered what key points could be made to help inform future evaluations by interested researchers in the community. Our observations led us to indicate three distinct layers that should be addressed for a significant, full-fledged appraisal.

The most high-level and general "system" layer calls for user evaluations that go beyond "quality of experience" and "satisfaction" surveys. Such evaluations should strive to address creative productivity and workflow efficiency aspects particular to the needs of computer-music practitioners.

At the mid-level "algorithmic" layer, we examine the mechanics of developing solutions strategies for concatenative synthesis. We have identified three main trends in algorithmic techniques used for tackling tasks in concatenative synthesis, namely, similarity-matrix and clustering approaches (like ours), Markov models, and constraint-satisfaction problems. Each of these techniques exhibits its own strengths and weaknesses in terms of accuracy, flexibility, efficiency, and complexity. Comparing these algorithms within a single system and, indeed, across multiple systems, using a well-defined data set, a clear set of goals, and specific success criteria would represent a valuable asset in the evaluation methodology of concatenative synthesis. Additionally, we should pay attention to the distance and similarity metrics used, as there are other possibilities that are explored and compared in other retrieval problems (e.g., Charulatha, Rodrigues, and Chitralekha 2013).

At the lowest level, the focus is on the broader implications related to MIR of choosing appropriate features for the task at hand. In the course of our evaluation, we chose the features indicated in the implementation and did not manipulate them in the experiment. There are, of course, many other features relevant to the problem that can be studied and estimated in a systematic way, as is par for the course in classification experiments in MIR. Furthermore, tuning the weights was not explored and is an important consideration that depends greatly on different corpora and output-sequence requirements.

In addition to this three-tiered evaluation methodology, an ideal component would be the availability of a baseline or comparison system that ensures new prototypes improve over some clearly identifiable aspect. Self-referential evaluations run the risk of

confirming experimenter bias without establishing comprehensive criticism.

Conclusion

In this article, we explored concatenative synthesis as a compositional tool for generating rhythmic patterns for electronic music, with a strong emphasis on its role in EDM musical styles. One of our first contributions was to present a thorough and up-to-date review of the state of the art, beginning with its fundamental algorithmic underpinnings and proceeding to modern systems that exploit new and experimental visual and interactive modalities. Although there are a number of commercial applications that encapsulate techniques of concatenative synthesis for the user, the vast majority of systems are frequently custom-built for the designer or are highly prototypical in nature. Consequently, there is a marked lack of evaluation strategies or reports of user experiences in the accompanying literature.

Based on these investigations, we set out to design a system that applied and extended many of the pervasive techniques in concatenative synthesis with a clear idea of its application and its target user. We built an instrument that was easily integrated with modern digital audio workstations and presented an interface that intended to be attractive and easy to familiarize oneself with. How to evaluate the system, not only in terms of its objective performance but also in its subjective aural and experiential implications for our users, was our final substantial contribution to this area. The results of our evaluations showed that our system performed as expected, and users were positive about its potential for assisting in their creative tasks, while also proposing interesting avenues for future work and contributions.

Resources

A demonstration version of the software is available online at http://github.com/carthach/rhythmCAT. A video example can be viewed at http://youtu.be/hByhgF_fzto.

References

Aucouturier, J.-J., and F. Pachet. 2005. "Ringomatic: A Real-Time Interactive Drummer Using Constraint-Satisfaction and Drum Sound Descriptors." *Proceedings of the International Conference on Music Information Retrieval*, pp. 412–419.

Bartók, B. 1993. "Hungarian Folk Music." In B. Suchoff, ed. *Béla Bartók Essays*. Lincoln: University of Nebraska Press, pp. 3–4.

Battier, M. 2007. "What the GRM Brought to Music: From *Musique Concrète* to Acousmatic Music." *Organized Sound* 12(3):189–202.

Bello, J., and L. Daudet. 2005. "A Tutorial on Onset Detection in Music Signals." *IEEE Transactions on Audio, Speech, and Language Processing* 13(5):1035–1047.

Bernardes, G., C. Guedes, and B. Pennycook. 2013. "EarGram?: An Application for Interactive Exploration of Concatenative Sound Synthesis in Pure Data." In M. Aramaki et al., eds. *From Sounds to Music and Emotions*. Berlin: Springer, pp. 110–129.

Böck, S., and G. Widmer. 2013. "Maximum Filter Vibrato Suppression for Onset Detection." In *Proceedings of the International Conference on Digital Audio Effects*. Available online at dafx13.nuim.ie/papers/09.dafx2013_submission_12.pdf. Accessed January 2017.

Bogdanov, D., et al. 2013. "ESSENTIA: An Audio Analysis Library for Music Information Retrieval." In *Proceedings of the International Conference on Music Information Retrieval*, pp. 493–498.

Bradski, G. 2000. "The OpenCV Library." *Dr. Dobb's Journal* 25(11):120–125.

Brent, W. 2010. "A Timbre Analysis and Classification Toolkit for Pure Data." In *Proceedings of the International Computer Music Conference*, pp. 224–229.

Cardle, M., S. Brooks, and P. Robinson. 2003. "Audio and User Directed Sound Synthesis." *Proceedings of the International Computer Music Conference*, pp. 243–246.

Charulatha, B., P. Rodrigues, and T. Chitralekha. 2013. "A Comparative Study of Different Distance Metrics That Can Be Used in Fuzzy Clustering Algorithms." *International Journal of Emerging Trends and Technology in Computer Science*. Available online at www.ijettcs.org/NCASG-2013/NCASG 38.pdf. Accessed January 2017.

Coleman, G. 2015. "Descriptor Control of Sound Transformations and Mosaicing Synthesis." PhD dissertation,

Universitat Pompeu Fabra, Department of Information and Communication Technologies, Barcelona.
Dixon, S. 2006. "Onset Detection Revisited." In *Proceedings of the International Conference on Digital Audio Effects*, pp. 133–137.
Donaldson, J., I. Knopke, and C. Raphael. 2007. "Chroma Palette?: Chromatic Maps of Sound as Granular Synthesis Interface." In *Proceedings of the Conference on New Interfaces for Musical Expression*, pp. 213–219.
Frisson, C. 2015. "Designing Interaction for Browsing Media Collections (by Similarity)." PhD dissertation, Université de Mons, Faculty of Engineering.
Frisson, C., C. Picard, and D. Tardieu. 2010. "Audiogarden?: Towards a Usable Tool for Composite Audio Creation." *QPSR of the Numediart Research Program* 3(2):33–36.
Gouyon, F., F. Pachet, and O. Delerue. 2000. "On the Use of Zero-Crossing Rate for an Application of Classification of Percussive Sounds." In *Proceedings of the International Conference on Digital Audio Effects*, pp. 3–8.
Greenberg, C. 1971. "Collage." In *Art and Culture: Critical Essays*. Boston, Massachusetts: Beacon Press, pp. 70–83.
Hackbarth, B., N. Schnell, and D. Schwarz. 2011. "AudioGuide?: A Framework for Creative Exploration of Concatenative Sound Synthesis." IRCAM Research Report. Available online at articles.ircam.fr/textes/Hackbarth10a/index.pdf. Accessed January 2017.
Herrera, P., A. Dehamel, and F. Gouyon. 2003. "Automatic Labeling of Unpitched Percussion Sounds." In *Proceedings of the 114th Audio Engineering Society Convention*. Available online at www.aes.org/e-lib/browse.cfm?elib=12599 (subscription required). Accessed January 2017.
Holmes, T. 2008. *Electronic and Experimental Music*. Abingdon-on-Thames, UK: Routledge.
Hoskinson, R., and D. Pai. 2001. "Manipulation and Resynthesis with Natural Grains." In *Proceedings of the International Computer Music Conference*, pp. 338–341.
Hoskinson, R., and D. K. Pai. 2007. "Synthetic Soundscapes with Natural Grains." *Presence: Teleoperators and Virtual Environments* 16(1):84–99.
Humphrey, E. J., D. Turnbull, and T. Collins. 2013. "A Brief Review of Creative MIR." In *Proceedings of the International Conference on Music Information Retrieval*. Available online at ismir2013.ismir.net/wp-content/uploads/2014/02/lbd1.pdf. Accessed January 2017.
Hunt, A. J., and A. W. Black. 1996. "Unit Selection in a Concatenative Speech Synthesis System Using a Large Speech Database." In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 373–376.
Jehan, T. 2005. "Creating Music by Listening." PhD dissertation, Massachusetts Institute of Technology, Media Arts and Sciences.
Knees, P., et al. 2016. "The GiantSteps Project: A Second-Year Intermediate Report." In *Proceedings of the International Computer Music Conference*, pp. 363–368.
Lochhead, J. I., and J. H. Auner. 2002. *Postmodern Music/-Postmodern Thought*: *Studies in Contemporary Music and Culture*. Abingdon-on-Thames, UK: Routledge.
Logan, B. 2000. "Mel Frequency Cepstral Coefficients for Music Modeling." In *Proceedings of the International Symposium on Music Information Retrieval*. Available online at ismir2000.ismir.net/papers/logan_paper.pdf. Accessed January 2017.
Loughran, R., et al. 2004. "The Use of Mel-Frequency Cepstral Coefficients in Musical Instrument Identification." In *Proceedings of the International Computer Music Conference*, pp. 42–43.
Mangani, M., R. Baldizzone, and G. Nobile. 2006. "Quotation in Jazz Improvisation: A Database and Some Examples." Paper presented at the International Conference on Music Perception and Cognition, August 22–26, Bologna, Italy.
Masri, P., and A. Bateman. 1996. "Improved Modeling of Attack Transients in Music Analysis-Resynthesis." In *Proceedings of the International Computer Music Conference*, pp. 100–103.
McLeod, K. 2009. "Crashing the Spectacle: A Forgotten History of Digital Sampling, Infringement, Copyright Liberation and the End of Recorded Music." *Culture Machine* 10:114–130.
Metzer, D. 2003. *Quotation and Cultural Meaning in Twentieth-Century Music*. Cambridge: Cambridge University Press.
Miller, P. D. 2008. *Sound Unbound: Sampling Digital Music and Culture*. Cambridge, Massachusetts: MIT Press.
O'Connell, J. 2011. "Musical Mosaicing with High Level Descriptors." Master's thesis, Universitat Pompeu Fabra, Sound and Music Computing, Barcelona.
Ó Nuanáin, C., P. Herrera, and S. Jordà. 2016. "An Evaluation Framework and Case Study for Rhythmic Concatenative Synthesis." In *Proceedings of the International Society for Music Information Retrieval Conference*, pp. 67–72.

Ó Nuanáin, C., S. Jordà, and P. Herrera. 2016a. "An Interactive Software Instrument for Real-Time Rhythmic Concatenative Synthesis." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 383–387.

Ó Nuanáin, C., S. Jordà, and P. Herrera. 2016b. "Towards User-Tailored Creative Applications of Concatenative Synthesis in Electronic Dance Music." In *Proceedings of the International Workshop on Musical Metacreation*. Available online at musicalmetacreation.org/buddydrive/file/nuanain_towards. Accessed January 2017.

Oswald, J. 1985. "Plunderphonics, or Audio Piracy as a Compositional Prerogative." Paper presented at the Wired Society Electro-Acoustic Conference, Toronto, Canada. Reprinted in *Musicworks*, Winter 1986, 34:5–8.

Peeters, G. 2004. "A Large Set of Audio Features for Sound Description (Similarity and Classification) in the CUIDADO Project." IRCAM Project Report. Available online at recherche.ircam.fr/anasyn/peeters/ARTICLES/Peeters_2003_cuidadoaudiofeatures.pdf. Accessed January 2017.

Rabiner, L. R. 1989. "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition." *Proceedings of the IEEE* 77(2):257–286.

Roads, C. 2004. *Microsound*. Cambridge, Massachusetts: MIT Press.

Rodgers, T. 2003. "On the Process and Aesthetics of Sampling in Electronic Music Production." *Organized Sound* 8(3):313–320.

Roy, P., F. Pachet, and S. Krakowski. 2007. "Analytical Features for the Classification of Percussive Sounds: The Case of the Pandeiro." In *Proceedings of the International Conference on Digital Audio Effects*, pp. 213–220.

Russell, S., and P. Norvig. 2009. *Artificial Intelligence: A Modern Approach*. Upper Saddle River, New Jersey: Prentice Hall.

Schwarz, D. 2003. "The Caterpillar System for Data-Driven Concatenative Sound Synthesis." In *Proceedings of the International Conference on Digital Audio Effects*, pp. 135–140.

Schwarz, D. 2005. "Current Research in Concatenative Sound Synthesis." In *Proceedings of the International Computer Music Conference*, pp. 9–12.

Schwarz, D. 2017. "Corpus-Based Sound Synthesis Survey." Available online at imtr.ircam.fr/imtr/Corpus-Based_Sound_Synthesis_Survey. Accessed February 2017.

Schwarz, D., et al. 2006. "Real-Time Corpus-Based Concatenative Synthesis with CataRT." In *Proceedings of the International Conference on Digital Audio Effects*, pp. 18–21.

Sewell, A. 2014. "Paul's Boutique and Fear of a Black Planet: Digital Sampling and Musical Style in Hip Hop." *Journal of the Society for American Music* 8(1):28–48.

Sturm, B. L. 2004. "Matconcat: An Application for Exploring Concatenative Sound Synthesis Using Matlab." In *Proceedings of the International Conference on Digital Audio Effects*, pp. 323–326.

Sturm, B. L. 2006. "Adaptive Concatenative Sound Synthesis and Its Application to Micromontage Composition." *Computer Music Journal* 30(4):46–66.

Tamagawa, K. 1988. "Echoes from the East: The Javanese Gamelan and Its Influence on the Music of Claude Debussy." DMA dissertation, University of Texas at Austin.

Thompson, L., S. Dixon, and M. Mauch. 2014. "Drum Transcription via Classification of Bar-Level Rhythmic Patterns." In *Proceedings of the International Society for Music Information Retrieval Conference*, pp. 187–192.

Tindale, A., et al. 2004. "Retrieval of Percussion Gestures Using Timbre Classification Techniques." *Proceedings of the International Conference on Music Information Retrieval*, pp. 541–545.

Turquois, C., et al. 2016. "Exploring the Benefits of 2D Visualizations for Drum Samples Retrieval." In *Proceedings of the ACM SIGIR Conference on Human Information Interaction and Retrieval*, pp. 329–332.

Tzanetakis, G., G. Essl, and P. Cook. 2001. "Automatic Musical Genre Classification of Audio Signals." In *Proceedings of the International Symposium on Music Information Retrieval*, pp. 293–302.

Xenakis, I. 1971. *Formalized Music*. Bloomington: Indiana University Press.

Xiang, P. 2002. "A New Scheme for Real-Time Loop Music Production Based on Granular Similarity and Probability Control." In *Proceedings of the International Conference on Digital Audio Effects*, pp. 89–92.

Zils, A., and F. Pachet. 2001. "Musical Mosaicing." In *Proceedings of the International Conference on Digital Audio Effects*, 1–6. Available online at www.csis.ul.ie/dafx01/proceedings/papers/zils.pdf. Accessed January 2017.