The Human Fingerprint in Machine Generated Music

Arne Eigenfeldt
arne_e@sfu.ca
Simon Fraser University, Vancouver, Canada

Keywords: Generative Music, Machine-Learning, Heuristics, Aesthetics of Generative Art.

Abstract: Machine-learning offers the potential for autonomous generative art creation. Given a corpus, the system can analyse it and provide rules from which to generate new art. The benefit of such a musical system is described, as well as the difficulties in its design and creation. This paper describes such a system, and the unintended heuristic decisions that were continually required.

xCoAx2013
Computation Communication Aesthetics and X. Bergamo, Italy. xcoax.org

1. Introduction

Machine-learning offers the potential for autonomous generative art creation. An ideal system may allow users to specify a corpus, from which the system derives rules and conditions in order to generate new art that reflects aspects of the corpus. High-level creativity may then be explored, not only by the careful selection of the corpus, but by the manipulation of the rules generated by the analysis.

Corpus-based re-composition has been explored most famously by Cope (Cope 2005), in which his system, EMI, was given representations of music by specific composers — for example, Bach and Mozart — and was successful in generating music within those styles (Cope 1991). Lewis used autoethnographic methods to derive rules for the creation of free jazz in his *Voyager* real-time performance system with which he, and other improvising musicians, interacted in performance (Lewis 2000). My own work with genetic algorithms used musical transcriptions of Indonesian Gamelan music to generate new works for string quartet (Eigenfeldt 2012). In the above cases, artistic creation was of paramount concern; as such, no attempt would have been made to *avoid* aesthetic decisions that would influence the output of the system (in fact, they would have been encouraged).

Using machine-learning for style modeling has been researched previously (Dubnov et al. 2003), however, their goals were more general in that composition was only one of many possible suggested outcomes from their initial work. Their examples utilized various monophonic corpora, ranging from "early Renaissance and baroque music to hard-bop jazz", and their experiments were limited to interpolating between styles rather than creating new, artistically satisfying music.

The concept of style extraction for reasons other than artistic creation has been researched more recently by Collins (Collins 2011), who tentatively suggested that, given the state of current research, it *may* be possible to successfully generate compositions within a style, given an existing database. This paper will describe our efforts to do just that, albeit with a liberal helping of heuristics.

2. Background

People unfamiliar with the aesthetics of generative art might be somewhat perplexed as to why any artist would want to surrender creative decision-making to a machine. Just as John Cage pursued chance procedures to eliminate the ego of the artist (Nyman 1999), I would suggest that generative artists have similarly turned to software in a search for new avenues of creativity outside of their own aesthetic viewpoints. The benefit of corpus-based generation avoids Cage's modernist reliance upon randomness, and investigates a post-modernist aesthetic of recombination.

As a creator of generative music systems for over twenty years, I have attempted — as have most other generative artists — to balance a systems' output between determinism and unpredictability. In other words, I approach the design process as both a composer — I want some control over the resulting music — and a listener — I want to hear music that surprises me with unexpected, but musically meaningful, decisions. Surprise is generally agreed to be an integral condition of creative systems (Bruner 1992).

Following in the footsteps of forerunners of interactive music systems (Chadabe 1984, Lewis 1999), my early systems equated 'surprise' with randomness, or, more specifically, constrained randomness (Eigenfeldt 1989). Randomness can generate complexity, and complexity is an over-reaching goal of contemporary music (Salzman 1967).

However, it becomes apparent rather quickly that while randomness — even constrained randomness — may generate unpredictability, the resulting complexity is, using a term posited by Weaver in 1948, *disorganized* (Weaver 1948), versus *organized* complexity that results from interaction of its constituent parts. In other words, randomness could never replicate the musical complexity exhibited in a work of music that plays with listener anticipations and expectations (Huron 2006). These expectations potentially build upon centuries of musical practice that involve notions of direction, motion, intensity, relaxation, resolution, deception, consonance and dissonance — none of which can be completely replaced by random methods.

2.1. Machine-Learning and Art Production

It makes sense, then, that in order to replicate intelligent human-generated artistic creation, it would be appropriate to apply elements of artificial intelligence towards this goal. Machine-learning, a branch of AI in which a system can learn to generalize its decision-making based upon data on which it has been trained, seems ideal for our purposes: not surprisingly, adventurous artists already have explored its potential, and with some initial success.

However, as is often the case with AI, such moderate initial successes have tended to plateau, and tangible artistic production examples are harder to find. ISMIR[1], the long-running conference concerned with machine-learning in music, has, since 2011, included concerts of music that incorporate machine-learning in some way; based upon attendee's informal responses, these concerts have proven to be somewhat unconvincing artistically. Music Information Retrieval (MIR), as evidenced by the vast majority of papers at ISMIR, is currently focused upon music recommendation and content analysis, two avenues with high profit potential. Those few papers with a musicological bent usually include a variation on the following caveat: "the audio content analysis used here cannot be claimed to be on a par with the musicologist's ear" (Collins 2012).

1. http://www.ismir.net/

The problem that is facing researchers in this particular field is that it is extremely difficult to derive meaningful information from the necessary data: audio recordings. Computational Audio Scene Analysis (Wang and Brown 2006) is a sub-branch of machine-learning that attempts to understand sound — or in this case music — using methods grounded in human perception. For example, an input signal must be broken down into higher level musical constructs, such as melody, harmony, bass line, beat structures, phrase repetitions and formal structures — an exceedingly difficult task, one which has not yet been solved. Our own research into transcribing drum patterns and extracting formal sections from recordings of electronic dance music (EDM) generated no higher than a 0.84 success rate, a rate good enough for publication (Eigenfeldt and Pasquier 2011), but lacking in usability. Therefore, we have resorted to expert human transcription: graduate students in music were hired to painstakingly transcribe all elements of the EDM tracks, including not only all instrumental parts, but signal processing and timbral analysis as well. This information can then be analysed as symbolic data, a much easier task.

3. The Generative Electronica Research Project

The Generative Electronica Research Project (GERP) is an attempt by our research group[2] — a combination of scientists involved in artificial intelligence, cognitive science, machine-learning, as well as creative artists — to generate stylistically valid EDM using human-informed machine-learning. We have employed experts to hand-transcribe 100 tracks in four genres: Breaks, House, Dubstep, and Drum and Bass. Aspects of transcription include musical details (drum beats, percussion parts, bass lines, melodic parts), timbral descriptions (i.e. 'low synth kick, mid acoustic snare, tight noise closed hihat'), signal processing (i.e. the use of delay, reverb, compression and its alteration over time), and descriptions of overall musical form. This information is then compiled in a database, and analysed to produce data for generative purposes.

2. http://www.metacreation.net/

Applying generative procedures to electronic dance music is not novel; in fact, it seems to be one of the most frequent projects undertaken by nascent generative musician/programmers. EDM's repetitive nature, explicit forms, and clearly delimited style suggest a parameterized approach. As with many cases of creative modeling, initial success will tend to be encouraging to the artist: generating beats, bass lines, and synth parts that resemble specific dance genres is not that difficult. However, progressing to a stage where the output is indiscernible from the model is another matter. In those cases, the 'artistic voice' argument tends to emerge: why spend the enormous effort required to accurately emulate someone else's music, when one can easily insert algorithms that reflect one's personal aesthetic? The resulting music, in such cases, is merely *influenced* by the model — a goal that is, arguably, more artistically satisfying than emulation, but less scientifically valid.

Our goal is, as a first step, to produce generative works that are modeled on a corpus, and indistinguishable from that corpus' style. There are two purposes to our work: the first purely experimental, the second artistic. In regards to the first, can we create high quality EDM using machine-learning? Without allowing for human/artistic intervention, can we extract formal procedures from the corpus and use this data to generate all aspects of the music so that a perspicacious listener of the genre will find it acceptable? We have already undertaken validation studies of other styles of generative music (Eigenfeldt et al. 2012), and now turn to EDM.

It is, however, the second purpose which dominates the motivation. As a composer, I am not interested in creating mere test examples that validate our methods. Instead, the goals remain artistic: can we generate EDM tracks and produce a full-evening event that is artistically satisfying, yet entertaining for the participants?

3.1. Initial success

As this is an artistic project using scientific methods (as opposed to pure scientific research), we are generating music at every stage, and judging our success not by quantitative methods, but qualitative ones. When analysis data was sparse in the formative stages of research, we had to make a great deal of artistic hypotheses. For example, after listening to the corpus many times, we made an initial assumption that a single 4-beat drum pattern existed within a track, and prior to its full exposition, masks were used to mute portions of it (i.e. the same pattern, but only the kick drum being audible): our generative system then followed this assumption. While any given generated track resembled the corpus, there was a sense of homogeneity between all generated tracks. With more detailed transcription, and its resulting richer data, the analysis engine produced statistically relevant information on exactly how often our assumption proved correct, as well as data as to what actually occurred within the corpus when our assumptions were incorrect (see Table 1). This information, used by the generative engine, produced an output with greater diversity, built upon data found within the corpus.

Table 1. Actual data on beat pattern repetition within 8 bar phrases.
Phrase patterns are the relationships of single 4-beat patterns within an 8-bar phrase.

Unique beat patterns in track	Unique phrase patterns in track	Probability
1	1	.29
> 1	1	.21
> 1	> 1	.5

4. Heuristic Decisions

What has proved surprising is the number of heuristic decisions that were deemed necessary in order to make the system produce successful music. New approaches in AI, specifically Deep Learning (Arel et al. 2010) suggest that unsupervised learning methods may be employed in order to derive higher-level patterns from within the data itself; in our case, not only should Deep Learning derive the drum patterns, but should be able to figure out *what* a beat variation actually is, and *when* it should occur. While one of our team members was able to use Deep Learning algorithms to generate stylistically accurate drum beats, the same result can be accomplished by my undergraduate music technology students after a few lessons in coding MaxMSP[3]. I would thus suggest that the latest approaches in AI can, at best, merely replicate a basic (not even expert) understanding of higher-level musical structures. In order for such structures to appear in corpus-based generative music, heuristic decisions remain necessary. One such example is in determining overall form.

3. A common music coding language, available at www.cycling74.com

4.1. Segmentation

As music is a time-based art-form, controlling how it unfolds over time is of utmost importance (and one of the most difficult aspects to teach beginning composition students). While it may not be as apparent to casual listeners as the surface details — such as the beat — form is a paramount organizing aspect that determines all constituent elements. As such, large-scale segmentation is often the first task in musical analysis; in our human transcription, this was indeed the case.

All the tracks in the repertoire exhibited, at most, five unique segments:

- Lead-in — the initial section with often only a single layer present: synth; incomplete beat pattern; guitar, etc.;
- Intro — a bridge between the Lead-in and the Verse: more instruments are present than the Lead-in, but not as full as the Verse;
- Verse — the main section of the track, in which all instruments are present, which can occur several times;
- Breakdown — a contrasting section to the verse in which the beat may drop out, or a filter may remove all mid– and high–frequencies. It will tend to build tension, and lead back to the verse;
- Outro — the fade-out of the track.

Many of these descriptions are fuzzy: at what point is does the Lead-In become the Intro? Is the entry of the drums required? (Sometimes.) Does one additional part constitute the change, or are more required? (Sometimes, and sometimes.) Interestingly, during the analysis, no discussion occurred as to what constitutes a segment break: they were intuitively assumed by our expert listeners. Apart from one or two instances, none of the segmentations were later questioned. Subsequent machine analysis of the data relied upon this labeling: for example, the various beat patterns were categorized based upon their occurrence within the sections, and clear differences were discovered. In other words, intuitive decisions were made that were later substantiated by the data. However, attempts to derive the segmentations autonomously proved less than successful, and relied upon further heuristic decisions as to what should even be searched for (Eigenfeldt and Pasquier 2011).

4.2. Discovering repetition

EDM contains a great deal of repetition — it is one of its defining features. It is important to point out that, while the specific patterns of repetition may not *define* the specific style, they do determine the uniqueness of the composition. Thus, for generative purposes, as opposed to mere style replication, such information is necessary for successful generation of *musical* material.

Table 2. Comparing the number of beat patterns per track, by style.

Style	Average # of patterns per track	Standard Deviation
Breaks	2.58	1.82
Dubstep	2.5	1.08
Drum & Bass	2.33	2.14
House	1.58	0.57

For example, Table 2 displays some cursory analysis of beat patterns per track, separated by style. Apart from the fact that House has a lower average, and there is significantly more variation in Drum & Bass, the number of patterns per track does not seem to be a discriminating indicator of style (see Table 2).

However, in order to *generate* music in this style, the number of patterns per track will need to be addressed: *when* do the patterns change (i.e. in which sections), and *where* do they change (i.e. within which phrase in a section)? As we were attempting to generate music based upon the Breaks corpus, further analysis of this data suggested that patterns tended to change more often directly at the section break, or immediately before it. Statistical analysis was then done in order to derive the probability of pattern changes occurring immediately on the section change, at the end of the section, or somewhere else within the section. Generation then took this into account.

The decision to include this particular feature occurred because we were attempting to emulate the specific musical characteristics of a style, Breaks; as such, it became one (of many) determining elements. However, it may *not* be important when attempting to generate House. House, which relies much more upon harmonic variation for interest, will require analysis of harmonic movement, which isn't necessary for Breaks. As such, heuristics were necessary in determining which features were important for the given style, a fact discovered by Collins when attempting to track beats in EDM (Collins 2006).

4.3. Computational Models of Style vs. Corpus-based Composition

As mentioned, our research is not restricted to re-creating a particular style of music, but creating music generatively within a particular style. The subtle difference is in intention: our aim is not to produce new algorithms in machine-learning to deduce, or replicate, style, but to explore new methods of generative music. As such, our analysis cannot be limited to aspects of style, which Pascal defines as a "distinguishing and ordering concept, both consistent of and denoting generalities" (Pascal 2013). As discussed in Section 4.2, how beat patterns are distributed through a track is not a stylistic feature, but one necessary for generation.

Pascal also states that style "represents a range or series of possibilities defined by a group of particular examples": this suggests a further distinction in what we require from the data. Analysis derives the *range* of possibilities for a given parameter. For generative purposes, this range becomes the search space. Allowing our generative algorithms to wander through this space will result in stylistically accurate examples, but ones of limited musical quality. This problem is more thoroughly discussed elsewhere, but can be summarized as the generated music being 'successful', but lacking *surprise* through its homogeneity (Eigenfeldt and Pasquier 2009).

Our new approach considers restricted search spaces, particularly in regard to consecutive generated works: composition A may explore one small area of the complete search space, while composition B may explore another area. This results in contrast between successive works, while maintaining consistency of style (see Figure 1).

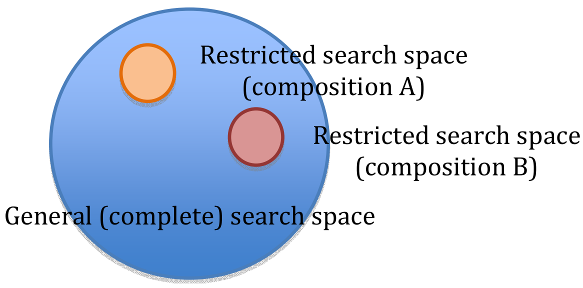

Fig. 1. Restricting search spaces for generative purposes.

5. Future Directions

Our current goal is the creation of a virtual Producer: a generative EDM artist that is capable of generating new EDM works based upon a varied corpus, with minimal human interaction. Using the restricted search space model suggested in Section 4.3, a wide variety of output is being generated, and can be found online[4]. The next step will be to create a virtual DJ: a generative EDM performer that assembles existing tracks created by the Producer into hour-long sets. Assemblage would involve signal analysis of every generated track's audio in order to determine critical audio features; individual track selection would then be carried out based upon a distance function between the track data and a generated timeline, which may or may not be derived from analysis of a given corpus consisting of DJ sets. This timeline could be varied in performance based upon real-time data: for example, movement analysis of the dance-floor could determine the ongoing success of the selected tracks.

4. soundcloud.com/loadbang

6. Conclusion

This paper has described the motivation for generating music using a corpus, and the difficulties inherent in the process. Our approach differs from others in that our motivations are mainly artistic. While attempting to eliminate the propensity to insert creative solutions, we have noticed that heuristic decisions remain necessary. We propose the novel solution of restricted search spaces, which further separate our research from style replication.

Acknowledgements: This research was funded by a grant from the Canada Council for the Arts, and the Natural Sciences and Engineering Research Council of Canada.

References

Arel, Itamar, Derek Rose, and Thomas Karnowski. *Deep Machine Learning — A New Frontier in Artificial Intelligence Research.* IEEE Computational Intelligence Magazine, November, 2010.

Bruner, Jerome. *The Conditions of Creativity.* Contemporary Approaches to Creative Thinking. H.E. Gruber, G. Terrell, and M. Wertheimer. USA: Atherton Press, 1962.

Chadabe, Joel. *Interactive Composing.* Computer Music Journal 8:1, 1984.

Collins, Nick. *Towards a style-specific basis for computational beat tracking.* International Conference on Music Perception and Cognition, 2006.

———. *Influence In Early Electronic Dance Music: An Audio Content Analysis Investigation.* Proceedings of the International Society for Music Information Retrieval, Porto, 2012.

Collins, Tom. *Improved methods for pattern discovery in music, with applications in automated stylistic composition.* PhD thesis, Faculty of Mathematics, Computing and Technology, The Open University, 2011.

Cope, David. *Computers and Musical Style.* Madison, WI: A-R Editions, 1991.

———. *Computer Models of Musical Creativity.* Cambridge, MA: MIT Press, 2005.

Chadabe, Joel. *Some Reflections on the Nature of the Landscape within which Computer Music Systems are Defined.* Computer Music Journal. 1:3, 1977.

Dubnov, Shlomo, Gerard Assayag, Olivier Lartillot and Gill Bejerano. *Using machine-learning methods for musical style modeling.* Computer, 36:10, 2003.

Eigenfeldt, Arne. *ConTour: A Real-Time MIDI System Based on Gestural Input.* International Conference of Computer Music (ICMC), Columbus, 1989.

———. *Corpus-based recombinant composition using a genetic algorithm.* Soft Computing — A Fusion of Foundations, Methodologies and Applications, 16:7, Springer, 2012.

Eigenfeldt, Arne and Philippe Pasquier. *A Realtime Generative Music System using Autonomous Melody, Harmony, and Rhythm Agents.* Proceedings of the XII Generative Art International Conference, Milan, 2009.

Eigenfeldt, Arne and Philippe Pasquier. *Towards a Generative Electronica: Human-Informed Machine Transcription and Analysis in MaxMSP.* Proceedings of Sound and Music Computing Conference, Padua, 2011.

Eigenfeldt, Arne, Philippe Pasquier and Adam Burnett. *Evaluating Musical Metacreation.* International Conference of Computation Creativity, Dublin, 2012.

Huron, David. *Sweet Anticipation: Music and the Psychology of Expectation.* Cambridge, MA: MIT Press, 2006

Lewis, George. *Interacting with latter-day musical automata.* Contemporary Music Review, 18:3, 1999.

———. *Too Many Notes: Computers, Complexity and Culture in Voyager.* Leonardo Music Journal 10, 2000.

Nyman, Michael. *Experimental Music: Cage and Beyond.* Cambridge University Press, 1999.

Pascal, Robert. *Style.* Grove Music Online. Oxford Music Online. Oxford University Press, accessed January 13, 2013,

Salzman, Eric. *Twentieth-Century Music: An Introduction.* Englewood Cliffs, New Jersey, Prentice-Hall, 1967.

Wang, DeLiang and Guy Brown. *Computational Auditory Scene Analysis: Principles, algorithms and applications.* IEEE Press/Wiley-Interscience, 2006.

Weaver, Warren. *Science and Complexity.* American Scientist, 36:536, 1948.