MOVING BEYOND FEATURE DESIGN: DEEP ARCHITECTURES AND AUTOMATIC FEATURE LEARNING IN MUSIC INFORMATICS

Eric J. Humphrey, Juan Pablo Bello
Music and Audio Research Lab, NYU
{ejhumphrey, jpbello}@nyu.edu

Yann LeCun
Courant School of Computer Science, NYU
yann@cs.nyu.edu

ABSTRACT

The short history of content-based music informatics research is dominated by hand-crafted feature design, and our community has grown admittedly complacent with a few de facto standards. Despite commendable progress in many areas, it is increasingly apparent that our efforts are yielding diminishing returns. This deceleration is largely due to the tandem of heuristic feature design and shallow processing architectures. We systematically discard hopefully irrelevant information while simultaneously calling upon creativity, intuition, or sheer luck to craft useful representations, gradually evolving complex, carefully tuned systems to address specific tasks. While other disciplines have seen the benefits of deep learning, it has only recently started to be explored in our field. By reviewing deep architectures and feature learning, we hope to raise awareness in our community about alternative approaches to solving MIR challenges, new and old alike.

1. INTRODUCTION

Since the earliest days of music informatics research (MIR), content-based analysis, and more specifically audio-based analysis, has received a significant amount of attention from our community. A number of surveys (e.g. [8, 22, 29]) amply document what is a decades-long research effort at the intersection of music, machine learning and signal processing, with wide applicability to a range of tasks including the automatic identification of melodies, chords, instrumentation, tempo, long-term structure, genre, artist, mood, renditions and other similarity-based relationships, to name but a few examples. Yet, despite a heterogeneity of objectives, traditional approaches to these problems are rather homogeneous, adopting a two-stage architecture of feature extraction and semantic interpretation, e.g. classification, regression, clustering, similarity ranking, etc.

Feature representations are predominantly hand-crafted, drawing upon significant domain-knowledge from music theory or psychoacoustics and demanding the engineering acumen necessary to translate those insights into algorithmic methods. As a result, good feature extraction is hard to come by and even more difficult to optimize, often taking several years of research, development and validation. Due in part to this reality, the trend in MIR is to focus on the use of ever-more powerful strategies for semantic interpretation, often relying on model selection to optimize results. Unsurprisingly, the MIR community is slowly converging towards a reduced set of feature representations, such as Mel-Frequency Cepstral Coefficients (MFCC) or chroma, now de-facto standards. This trend will only become more pronounced given the growing popularity of large, pre-computed feature datasets [1].

We contend the tacit acceptance of common feature extraction strategies is short-sighted for several reasons: first, the most powerful semantic interpretation method is only as good as a data representation allows it to be; second, mounting evidence suggests that appropriate feature representations significantly reduce the need for complex semantic interpretation methods [2, 9]; third, steady incremental improvements in MIR tasks obtained through persistence and ingenuity indicate that the the costly practice of manual feature optimization is not yet over; and fourth, task-specific features are ill-posed to address problems for which they were not designed (such as mood estimation or melody extraction), thus limiting their applicability to these and other research areas that may emerge.

In this paper we advocate a combination of deep signal processing architectures and automatic feature learning as a powerful, holistic alternative to hand-crafted feature design in audio-based MIR. We show how deeper architectures are merely extensions of standard approaches, and that robust music representations can be achieved by breaking larger systems into a hierarchy of simpler parts (Section 3). Furthermore, we also show that, in light of initial difficulties training flexible machines, automatic learning methods now exist that actually make these approaches feasible, and early applications in MIR have shown much promise (Section 4). This formulation provides several important advantages over manual feature design: first, it allows for joint, fully-automated optimization of the feature extraction and semantic interpretation stages, blurring boundaries between the two; second, it results in general-purpose architectures that can be applied to a variety of specific MIR problems; and lastly, automatically learned features can offer objective insight into the relevant musical attributes for a given task. Finally, in Section 5, we

[1] Million Song Dataset: http://labrosa.ee.columbia.edu/millionsong/

conclude with a set of potential challenges and opportunities for the future.

2. CLASSIC APPROACHES TO CLASSIC PROBLEMS

2.1 Two-Stage Models

In the field of artificial intelligence, computational perception can be functionally reduced to a two-tiered approach of data representation and semantic interpretation. A signal is first transformed into a data representation where its defining characteristics are made invariant across multiple realizations, and semantic meaning can subsequently be inferred and used to assign labels or concepts to it. Often the goal in music informatics is to answer specific questions about the content itself, such as "is this a C major triad?" or "how similar are these two songs?"

More so than assigning meaning, the underlying issue is ultimately one of organization and variance. The better organized a representation is to answer some question, the simpler it is to assign or infer semantic meaning. A representation is said to be *noisy* when variance in the data is misleading or uninformative, and *robust* when it predictably encodes these invariant attributes. When a representation explicitly reflects a desired semantic organization, assigning meaning to the data becomes trivial. Conversely, more complicated information extraction methods are necessary to compensate for any noise.

In practice, this two-stage approach proceeds by feature extraction – transforming an observed signal to a hopefully robust representation – and either classification or regression to model decision-making. Looking back to our recent history, there is a clear trend in MIR of applying increasingly more powerful machine learning algorithms to the same feature representations to solve a given task. In the ISMIR proceedings alone, there are twenty documents that focus primarily on audio-based automatic chord recognition. All except one build upon chroma features, and over half use Hidden Markov Models to stabilize classification; the sole outlier uses a Tonnetz representation, which are tonal centroid features derived from chroma. Though early work explored the use of simple binary templates and maximum likelihood classifiers, more recently Conditional Random Fields, Bayesian Networks, and Support Vector Machines have been introduced to squeeze every last percentage point from the same features.

If a feature representation were truly robust, the complexity of a classifier – and therefore the amount of variance it could absorb – would have little impact on performance. Previous work in automatic chord recognition demonstrates the significance of robust feature representations, showing that the appropriate filtering of chroma features leads to a substantial increase in system performance for the simplest classifiers, and an overall reduction of performance variation across all classifiers [9]. Additionally, researchers have for some time addressed the possibility that we are converging to glass ceilings in content-based areas like acoustic similarity [2]. Other hurdles, like the issue of hubs and orphans, have been shown to be not merely a peculiarity of the task but rather an inevitability of the feature representation [20]. As we consider the future of MIR, it is necessary to recognize that diminishing returns in performance are far more likely the result of sub-optimal features than the classifier applied to them.

2.2 From Intuition to Feature Design

Music informatics is traditionally dominated by the hand-crafted design of feature representations. Noting that design itself is a well-studied discipline, a discussion of feature design is served well by the wisdom of "getting the right design and the design right" [6]. Reducing this aphorism to its core, there are two separate facets to be considered: finding the right conceptual representation for a given task, and developing the right system to produce it.

Consider a few signal-level tasks in MIR, such as onset detection, chord recognition or instrument classification, noting how each offers a guiding intuition. Note onsets are typically correlated with transient behavior. Chords are defined as the combination of a few discrete pitches. Classic studies in perception relate timbre to aspects of spectral contour [12]. Importantly, intuition-based design hinges on the assumption that someone can know what information is necessary to solve a given problem.

Having found conceptual direction, it is also necessary to craft the right implementation. This has resulted in substantial discourse and iterative tuning to determine better performing configurations of the same basic algorithms. Much effort has been invested in determining which filters and functions make better onset detectors [3]. Chroma – arguably the only music-specific feature developed by our community – has undergone a steady evolution since its inception, gradually incorporating more levels of processing to improve robustness [28]. Efforts to characterize timbre, for which a meaningful definition remains elusive, largely proceed by computing numerous features or, more commonly, the first several MFCCs [11].

In reality, feature design presents not one but two challenges – concept and implementation – and neither have proven easy to solve. First off, our features are ultimately constrained to those representations we can conceive or comprehend. Beyond relatively obvious tasks like onset detection and chord recognition, we can only begin to imagine what abstractions might be necessary to perform rather abstract tasks like artist identification. Furthermore, recognizing that feature extraction is still an open research topic, the considerable inertia of certain representations is cause for concern: 19 of 26 signal-based genre classification systems in the ISMIR proceedings are based on MFCCs, for example, many using publicly-available implementations. While sharing data and software is a commendable trend, now is a critical point in time to question our acceptance of these representations as we move toward the widespread use of pre-computed feature collections, e.g. the Million Song Dataset. Finally, above all else, the practice of hand-crafted feature design is simply not sustainable. Manually optimizing feature extraction methods proceeds at a glacial pace and incurs the high costs of time, effort and funding. Somewhat ironically, the MIR community has collectively recognized the benefits of automatically fitting our classifiers, but feature optimization – the very data those methods depend on – remains largely heuristic.

Alternatively, data-driven approaches in *deep learning* have recently shown promise toward alleviating each and every one of these issues. Proven numerical methods can adapt a system infinitely faster than is attainable by our current research methodology, and the appropriate conceptual representations are realized as a by-product of optimizing an objective function. In the following section, we will illustrate how robust feature representations can be achieved through deep, hierarchical structures.

3. DEEP ARCHITECTURES

3.1 Shallow Architectures

Time-frequency analysis is the cornerstone of audio signal processing, and modern architectures are mainly comprised of the same processing elements: linear filtering, matrix transformations, decimation in time, pooling across frequency, and non-linear operators, such as the complex modulus or logarithmic compression. Importantly, the combination of time-domain filtering and decimation is often functionally equivalent to a matrix transformation – the Discrete Fourier Transform (DFT) can be easily interpreted as either, for example – and for the sake of discussion, we refer to these operations collectively as *projections*.

Now, broadly speaking, the number of projections contained within an information processing architecture determines its *depth*. It is critical to recognize, however, that the extraction of meaningful information from audio proceeds by transforming a time-varying function – a signal – into an instantaneous representation – features; at some specificity, all signals represent static concepts, e.g., a single piano note versus the chorus of a song. Therefore, the depth at which a full signal is summarized by a stationary feature vector is characteristic of a signal processing architecture, and is said to be particularly *shallow* if an entire system marginalizes the temporal dimension with only a single projection.

This is a subtle, but crucial, distinction to make; *feature* projections, lacking a time dimension, are a subset of *signal* projections. As we will see, shallow signal processing architectures may still incorporate deep feature projections, but the element of time warrants special attention. A signal projection that produces a finite set of stationary features attempts to capture *all* relevant information over the observation, and any down-stream representations are constrained by whatever was actually encoded in the process. Importantly, the range of observable signals becomes infinite with increasing duration, and it is progressively more taxing for signal projections – and therefore shallow architectures – to accurately describe this data without a substantial loss of information.

To illustrate the point further, consider the two signal processing architectures that produce Tonnetz and MFCC

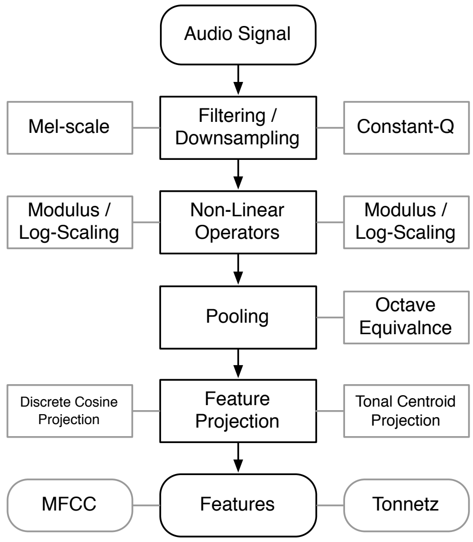

Figure 1: Tonnetz and MFFCs from Shallow Architectures

features. As shown in Figure 1, the processing chains are nearly identical; note that the penultimate representation when computing Tonnetz features is chroma. Both begin with a signal projection that maps a time-domain signal to an instantaneous estimation of frequency components, and conclude with a feature projection that reorganizes the estimated frequencies in task-specific ways. The overwhelming majority of music signal processing architectures operate in this paradigm of shallow signal transformations. Subject to the Fourier uncertainty principle, these systems exhibit time-frequency trade-offs and are constrained in practice to the analysis of short observations.

The vast majority of musical experiences do not live in short signals however, and it is therefore necessary to characterize information over longer durations. Previous efforts recognize this deficiency and address it through one of a few simple methods: a *bag of frames (BoF)* models features as a probability distribution, *shingling* concatenates feature sequences into a vector, or *delta-coefficients* represent low-order derivatives calculated over local features. These naive approaches are ill-posed to characterize the temporal dynamics of high-level musical concepts like mood or genre, and arguably contribute to the "semantic gap" in music informatics. It will become clear in the following discussion why this is the case, and how deeper architectures can alleviate this issue.

3.2 Motivating Deeper Architectures

This previous discussion begs a rather obvious question: why are shallow architectures poorly suited for music signal processing? If we consider how music is constructed, it is best explained by a *compositional containment hierarchy*. The space of musical objects is not flat, but rather pitch and intensity combine to form chords, melodies and rhythms, which in turn build motives, phrases, sections and entire pieces. Each level uses simpler elements to produce an emergent whole greater than the sum of its parts, e.g., a melody is more than just a sequence of pitches.

In a similar fashion, deeper signal processing structures can be realized by stacking multiple shallow architectures, and are actually just extensions of modern approaches. For a signal projection to marginalize time with a minimal loss of information, the observation must be locally stationary, and clearly this cannot hold for long signals. Sequences of instantaneous features, however, are again time-varying data and, when appropriately sampled, *are* themselves locally stationary signals. There are two remarkable conclusions to draw from this. First, everything we know about one-dimensional signal processing holds true for a time-feature signal and can be generalized thusly. And furthermore, simply cascading multiple shallow architectures relaxes previous constraints on observation length by producing locally stationary signals at various time-scales.

This hierarchical signal processing paradigm is at the heart of deeper architectures. There are many benefits detailed at length in [4], but two are of principal importance here: one, multi-layer processing allows for the emergence of higher-level attributes for two related reasons: deep structures can break down a large problem into a series of easier sub-problems, and each requires far fewer elements to solve than the larger problem directly; and two, each layer can absorb some specific variance in the signal that is difficult or impossible to achieve directly. Chord recognition captures this intuition quite well. One could define every combination of absolute pitches in a flat namespace and attempt to identify each separately, or they could be composed of simpler attributes like intervals. Slight variations, like imperfect intonation, can be reconciled by a composition of intervals, whereas a flat chord-space would need to address this explicitly.

Both of these benefits are observed in the successful application of convolutional neural networks (CNN) to handwritten digit classification [25]. Most prior neural network research in computer vision proceeded by applying multilayer perceptrons (MLP) directly to a pixel values of an image, which struggles to cope with spatial variation. Adopting a CNN architecture introduces a hierarchical decomposition of small, locally-correlated areas, acting as signal projections in space rather than time. Emergent properties of images are encoded in the visual geometry of edges, corners, and so on, and the architecture is able to develop an invariance to spatial translations and scaling.

Within audio signal processing, wavelet filterbanks, as cascaded signal projections, have been shown to capture long-term information for audio classification [1]. These second-order features yielded better classification results than first-order MFCCs over the same duration, even allowing for convincing signal reconstruction of the original signals. This outcome is evidence to the fact that deeper signal processing architectures can lead to richer representations over longer durations. Observing that multilayer architectures are simply extensions of common approaches, it is fascinating to discover there is at least one instance in MIR where a deep architecture has naturally evolved into the common solution: tempo estimation.

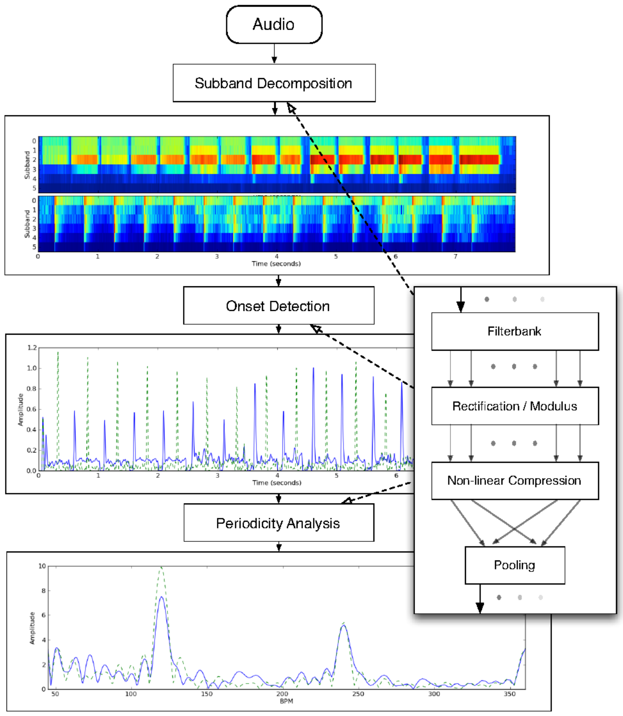

Figure 2: Tempo Estimation with Deep Signal Processing Architectures.

3.3 Deep Signal Processing in Practice

Upon closer inspection, modern tempo estimation architectures reveal deep architecture with strong parallels to CNNs and wavelets. Rhythmic analysis typically proceeds by decomposing an audio signal into frequency subbands [31]. This time-frequency representation is logarithmically scaled and subbands are pooled, reducing the number of components. Remaining subbands are filtered in time by what amounts to an edge detector, rectified, pooled along subbands and logarithmically scaled to yield a novelty function [23]. A third and final stage of filtering estimates tempo-rate frequency components in the novelty signal, producing a tempogram [13].

Over the course of a decade, the MIR community has collectively converged to a deep signal processing architecture for tempo estimation and, given this progress, it is possible to exactly illustrate the advantages of hierarchical signal analysis. In Figure 2, two waveforms with identical tempi but different incarnations – a trumpet playing an ascending D major scale and a series of bass drum hits, set slightly out of phase – are shown at various stages of the tempo estimation architecture. It is visually apparent that each stage in the architecture absorbs a different type of variance in the signal: pitch and timbre, absolute amplitude, and phase information, respectively. By first breaking the problem of tempo estimation into two sub-tasks – frequency estimation and onset detection – it becomes possible to characterize subsonic frequencies at both lower sampling frequencies and with a fewer number of components.

Realistically though, progress in tempo estimation is the result of strong intuition that could guide system design. The inherent challenge in building deep, hierarchical systems is that intuition and understanding quickly depart after more than even a few levels of abstraction. Therein lies the most exciting prospect of this whole discourse; given a well-defined objective function, it is possible to automatically learn both the right conceptual representation and the right system to produce it for a specific application.

4. FEATURE LEARNING

4.1 From Theory to Practice

For some time, a concerted effort in computer science has worked toward the development of convex optimization and machine learning strategies. Unfortunately, the initial surge of activity and excitement surrounding artificial intelligence occurred well before technology could handle the computational demands of certain methods, and as a result many approaches were viewed as being intractable, unreasonable, or both. Over the last two or so decades, the state of affairs in machine learning has changed dramatically, and for several reasons feature learning is now not only feasible, but in many cases, efficient.

Almost more importantly than its success as an image classification system, the work in [25] proved that stochastic gradient descent could be used to discriminatively train large neural networks in a supervised manner. Given a sufficient amount of labeled data, many applications in computer vision immediately benefited from adopting these approaches. Such datasets are not always available or even possible, and recent breakthroughs in unsupervised training of Deep Belief Networks (DBNs) have had a similar impact [17]. This work has also been extended to a convolutional variant (CDBNs), showing great promise for deep signal processing [26]. Additionally, auto-encoder architectures are a recent addition to the unsupervised training landscape and offer similar potential [21].

The significance of ever-increasing computational power is also not to be overlooked in the proliferation of automatic feature learning. Steady improvements in processing speed are now being augmented by a rise in parallel computing solutions and toolkits [5], decreasing training times and accelerating research. Taken together, these strategies encompass a set of deep learning approaches that hold significant potential for applications in music informatics.

4.2 Early Efforts in Music Informatics

It is necessary to note that leveraging data to automatically learn feature representations is not a new idea. The earliest effort toward automatic feature learning is that of [7, 33], where genetic algorithms were used to automatically learn optimized feature transformations.Though not a deep architecture in the classic sense, this work formally recognized the challenge of hand-crafting musical representations and pioneered feature learning in MIR.

With respect to deeper architectures, the first successful instance of deep feature learning is that of CNN-based onset detection by [24]. More recently, CNNs have been applied to automatic genre recognition [27], instrument classification [19] and automatic chord recognition [18]. Alternatively, DBNs have seen a noticeable rise in frame-level applications, such as instrument classification [15], piano transcription [30], genre identification [14] and mood prediction [32], out-performing other shallow, MFCC-based systems. Incorporating longer time-scales, convolutional DBNs have also been explored in the context of various speech and music classification tasks in [26], and for artist, genre and key recognition [10]. Predictive sparse coding has also been applied to genre recognition, earning "Best Student Paper" at ISMIR 2011 [16].

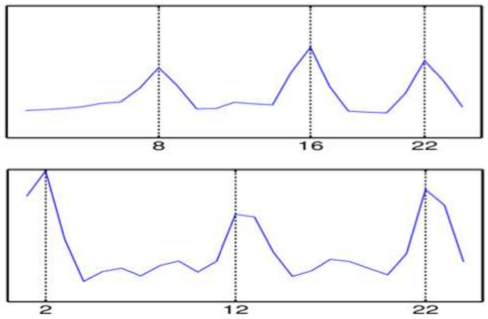

Figure 3: Learned Features for Genre Recognition (Reprinted with permission)

The most immediate observation to draw from this short body of work is that every system named above achieved state-of-the-art performance, or better, in substantially less time than it took to get there by way of hand-crafted representations. Noting that many of these systems are the first application of deep learning in a given area of MIR, it is only reasonable to expect these systems to improve in the future. For instance, DBNs have been primarily used for frame-level feature learning, and it is exciting to consider what might be possible when all of these methods are adapted to longer time scales and for new tasks altogether.

A more subtle observation is offered by this last effort in genre recognition [16]. Interestingly, the features learned from Constant-Q representations during training would seem to indicate that specific pitch intervals and chords are informative for distinguishing between genres. Shown in Figure 3, learned dictionary elements capture strong fifth and octave interval relationships versus quartal intervals, each being more common in rock and jazz, respectively. This particular example showcases the potential of feature learning to reformulate established MIR tasks, as it goes against the long-standing intuition relating genre to timbre and MFCCs.

5. THE FUTURE OF DEEP LEARNING IN MIR

5.1 Challenges

Realistically speaking, deep learning methods are not without their own research challenges, and these difficulties are contributing factors to limited adoption within our community. Deep architectures often require a large amount of labeled data for supervised training, a luxury music informatics has never really enjoyed. Given the proven success of supervised methods, MIR would likely benefit a good deal from a concentrated effort in the curation of sharable data in a sustainable manner. Simultaneously, unsupervised methods hold great potential in music-specific contexts, as they tend to circumvent the two biggest issues facing supervised training methods: the threat of over-fitting and a need for labeled data.

Additionally, there still exists a palpable sense of mistrust among many toward deep learning methods. Despite decades of fruitful research, these approaches lack a solid, foundational theory to determine how, why, and if they will work for a given problem. Though a valid criticism, this should be appreciated as an exciting research area and not a cause for aversion. Framing deep signal processing architectures as an extension of shallow time-frequency analysis provides an encouraging starting point toward the development of more rigorous theoretical foundations.

5.2 Impact

Deep learning itself is still a fledgling research area, and it is still unclear how this field will continue to evolve. In the context of music informatics, these methods offer serious potential to advance the discipline in ways that cannot be realized by other means. First and foremost, it presents the capacity for the abstract, hierarchical analysis of music signals, directly allowing for the processing of information over longer time scales. It should come as no surprise that determining the similarity of two songs based on small-scale observations has its limitations; in fact, it should be amazing that it works at all.

More practically, deep learning opens the door for the application of numerical optimization methods to accelerate research. Instead of slowly converging to the best chroma transformation by hand, an automatically trained system could do this in a fraction of the time, or find a better representation altogether. In addition to reframing well-known problems, deep learning also offers a solution to those that lack a clear intuition about how a system should be designed. A perfect example of this is found in automatic mixing; we know a "good" mix when we hear one, but it is impossible to articulate the contributing factors in a general sense. Like the work illustrated in Figure 3, this can also provide insight into what features are informative to a given task and create an opportunity for a deeper understanding of music in general.

6. REFERENCES

[1] J. Andén and S. Mallat. Multiscale scattering for audio classification. In *Proc. ISMIR*, 2011.

[2] J. J. Aucouturier. Music similarity measures: What's the use? In *Proc. ISMIR*, 2002.

[3] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. Sandler. A tutorial on onset detection in music signals. *IEEE Trans. Audio, Speech and Language Processing*, 13(5):1035–1047, 2005.

[4] Y. Bengio. Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2:1–127, 2009.

[5] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio. Theano: a CPU and GPU math expression compiler. In *Proc. SciPy*, 2010.

[6] B. Buxton. *Sketching User Experiences: Getting the Design Right and the Right Design*. Morgan Kaufmann, 2007.

[7] G. Cabral and F. Pachet. Recognizing chords with EDS: Part One. *Computer Music Modeling and Retrieval*, pages 185 – 195, 2006.

[8] M. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney. Content-based music information retrieval: Current directions and future challenges. *Proc. IEEE*, 96(4):668–696, 2008.

[9] T. Cho, R. J. Weiss, and J. P. Bello. Exploring common variations in state of the art chord recognition systems. In *Proc. SMC*, 2010.

[10] S. Dieleman, P. Brakel, and B. Schrauwen. Audio-based music classification with a pretrained convolutional network. In *Proc. ISMIR*, 2011.

[11] S. Essid, G. Richard, and B. David. Musical instrument recognition by pairwise classification strategies. *IEEE Trans. Audio, Speech and Language Processing*, 14(4):1401–1412, 2006.

[12] J. M. Grey. Multidimentional perceptual scaling of musical timbre. *Jnl. Acoustical Soc. of America*, 61:1270–1277, 1977.

[13] P. Grosche and M. Müller. Extracting predominant local pulse information from music recordings. *IEEE Trans. Audio, Speech and Language Processing*, 19(6):1688–1701, 2011.

[14] P. Hamel and D. Eck. Learning features from music audio with deep belief networks. In *Proc. ISMIR*, 2010.

[15] P. Hamel, S. Wood, and D. Eck. Automatic identification of instrument classes in polyphonic and poly-instrument audio. In *Proc. ISMIR*, 2009.

[16] M. Henaff, K. Jarrett, K. Kavukcuoglu, and Y. LeCun. Unsupervised learning of sparse features for scalable audio classification. In *Proc. ISMIR*, 2011.

[17] G. E. Hinton, S. Osindero, and Y. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006.

[18] E. J. Humphrey, T. Cho, and J. P. Bello. Learning a robust tonnetz-space transform for automatic chord recognition. In *Proc. ICASSP*, 2012.

[19] E. J. Humphrey, A. P. Glennon, and J. P. Bello. Non-linear semantic embedding for organizing large instrument sample libraries. In *Proc. ICMLA*, 2010.

[20] I. Karydis, M. Radovanovic, A. Nanopoulos, and M. Ivanovic. Looking through the "glass ceiling": A conceptual framework for the problems of spectral similarity. In *Proc. ISMIR*, 2010.

[21] K. Kavukcuoglu, P. Sermanet, Y. Boureau, K. Gregor, M. Mathieu, and Y. LeCun. Learning convolutional feature hierarchies for visual recognition. In *Proc. NIPS*, 2010.

[22] A. Klapuri and M. Davy. *Signal Processing Methods for Music Transcription*. Springer, 2006.

[23] A. P. Klapuri, A. J. Eronen, and J. T. Astola. Analysis of the meter of acoustic musical signals. *IEEE Trans. Audio, Speech and Language Processing*, 14(1):342–355, 2006.

[24] A. Lacoste and D. Eck. A supervised classification algorithm for note onset detection. *EURASIP Jnl. on Adv. in Signal Processing*, pages 1–14, 2007.

[25] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

[26] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proc. ICML*, 2009.

[27] T. Li, A. Chan, and A. Chun. Automatic musical pattern feature extraction using convolutional neural network. In *Proc. IMECS*, 2010.

[28] M. Müller. *Information Retrieval for Music and Motion*. Springer Verlag, 2007.

[29] M. Müller, D.P.W. Ellis, A. Klapuri, and G. Richard. Signal processing for music analysis. *Jnl. Selected Topics in Sig. Proc.*, 5(6):1088–1110, 2011.

[30] J. Nam, J. Ngiam, H. Lee, and M. Slaney. A classification-based polyphonic piano transcription approach using learned feature representations. In *Proc. ISMIR*, 2011.

[31] E. D. Scheirer. Tempo and beat analysis of acoustic musical signals. *Jnl. Acoustical Soc. of America*, 103(1):588–601, 1998.

[32] E. M. Schmidt and Y. E. Kim. Modeling the acoustic structure of musical emotion with deep belief networks. In *Proc. NIPS*, 2011.

[33] A. Zils and F. Pachet. Automatic extraction of music descriptors from acoustic signals using EDS. In *Proc. AES*, 2004.