A COMPOSITIONAL HIERARCHICAL MODEL FOR MUSIC INFORMATION RETRIEVAL

Matevž Pesek
University of Ljubljana
Faculty of computer
and information science
matevz.pesek@fri.uni-lj.si

Aleš Leonardis
Centre for Computational
Neuroscience and Cognitive Robotics
School of Computer Science
University of Birmingham
ales.leonardis@fri.uni-lj.si

Matija Marolt
University of Ljubljana
Faculty of computer
and information science
matija.marolt@fri.uni-lj.si

ABSTRACT

This paper presents a biologically-inspired compositional hierarchical model for MIR. The model can be treated as a deep learning model, and poses an alternative to deep architectures based on neural networks. Its main features are generativeness and transparency that allow clear insight into concepts learned from the input music signals. The model consists of multiple layers, each is composed of a number of parts. The hierarchical nature of the model corresponds well with the hierarchical structures in music. Parts in lower layers correspond to low-level concepts (e.g. tone partials), while parts in higher layers combine lower-level representations into more complex concepts (tones, chords). The layers are unsupervisedly learned one-by-one from music signals. Parts in each layer are compositions of parts from previous layers based on statistical co-occurrences as the driving force of the learning process. We present the model's structure and compare it to other deep architectures. A preliminary evaluation of the model's usefulness for automated chord estimation and multiple fundamental frequency estimation tasks is provided. Additionally, we show how the model can be extended to event-based music processing, which is our final goal.

1. INTRODUCTION

The field of music information retrieval (MIR) has reached a significant expansion in tasks and solutions in the short timespan of its existence [3, 10]. The tasks include extraction of high-level music descriptors from music, such as melody, chords and rhythm, as well as highly perceptual tasks involving mood estimation, genre recognition and artist influence. Solutions have not come to a perfect one for any of the described tasks yet; however, numerous approaches proposed each year are improving the state-of-the-art rapidly. Recently, deep belief networks as an alternative single model for a variety of tasks, have been successfully introduced to the field.

This paper presents a biologically-inspired compositional hierarchical model for music information retrieval. The proposed model poses an alternative to recent deep learning architecture approaches [6, 9]. Its main difference from the latter is in its transparent structure, thus allowing representation and interpretation of the signal's information extracted on different levels. We show the usefulness of our proposed approach in a preliminary evaluation of the model for the tasks of automated chord estimation and multiple fundamental frequency estimation. We also show how the model can be extended to event-based music processing, and point out how the model's transparency enables other applications of the model, e.g. for music analysis, synthesis and visualization.

2. DEEP ARCHITECTURES FOR MIR

The concept of deep learning has grown in popularity in the fields of signal processing [15], audio processing [9] and MIR. Lee [7] presented one of the first attempts of using deep belief networks (DBNs) on audio signals, where convolutional DBNs were applied to the speaker identification task. A DBN was used as a feature extractor, and a support vector machine for classification.

Later, Hamel and Eck [5], evaluated DBNs for genre recognition using a five-layer DBN with three hidden layers for feature extraction. The support vector machine was used for classification, where as raw spectral data was used as input to the DBN. DBNs show great potential for many tasks that involve high-level feature extraction, such as emotion recognition, since there is usually no trivial spectral or temporal feature that could be used to model the high-level representation in question. Schmidt and Kim [13] showed promising results by using a 5-layer DBN for extraction of emotion-based acoustic features. Other approaches modeled temporal aspects of the audio signal. Conditional DBN-s were used by Battenberg and Wessel [1] for drum pattern analysis. Schmidt [12] took a step further and showed that DBNs can be trained for discriminating rhythm and melody.

Overall, recent research has shown great interest and success in using features learned from music signals, in contrast to previously used hand-crafted features. The research reviewed in this subsection took place only in the last few years; thus, there is a vast expansion of deep learning in MIR to be expected, as anticipated by Humphrey [6].

3. THE COMPOSITIONAL HIERARCHICAL MODEL

3.1 Motivation and concept

DBNs brought an improvement to many MIR tasks with their unsupervised learning of features and generative modeling. However, they require a large set of hidden units per layer, and consequently, large training sets. Also, the hidden nature of units offers no clear explanation of the undergoing feature extraction process and the meaning of extracted features. It is our goal to overcome these limitations by developing a white-box compositional hierarchical model with shareable parts, thus reducing the number of parts and learning data needed, as well as reaching transparency in terms of interpretable internal structure of the model.

The proposed model provides a hierarchical representation of the audio signal, from the signal components on the lowest level, up to individual musical events on the highest levels. It is built on the assumption that a complex signal can be decomposed into a hierarchy of building blocks - *parts*. These parts exist at various levels of granularity and represent sets of entities describing the signal. According to their complexity, parts can be structured across several layers from less to the more complex. Parts on higher layers are expressed as compositions of parts on lower layers (e.g.: a chord is composed of several pitches, each pitch of several harmonics etc.). A part can therefore describe individual frequencies in a signal, their combinations, as well as pitches, chords and temporal patterns, such as chord progressions.

The structure of our model is inspired by work in computer vision, specifically the hierarchical compositional model presented by Leonardis and Fidler [8]. Their model represents objects in images in a hierarchical manner, structured in layers from simple to complex image parts. The model is learned from the statistics of natural images and can be employed as a robust statistical engine for object categorization and other computer vision tasks. We believe that such approach can also be used for music representation and analysis, however the transformation of the model to a different domain is not trivial.

3.2 Model structure

The compositional hierarchical model consists of several layers. Each layer contains a set of parts. A part is a composition of two or more parts from a lower layer and may itself be part of any number of compositions on a higher layer. Thus, the compositional model forms a hierarchy of parts, where each part represents a composition of lower-layer parts, as seen in Figure 1. Connections in the figure represent compositions of parts.

3.2.1 Input layer

The input layer of the model is derived from the time-frequency representation of the music signal. We denote this layer as layer $\mathcal{L}_0$. It contains a single atomic part, which is activated (produces output) at locations of all frequency components in the signal at a given time instance. An example is given in Figure 1, although not all activations are shown for clarity. More formally, a part's activation is defined by two values: location L_P that corresponds to frequency, and magnitude A_P, that corresponds to magnitude of the frequency component.

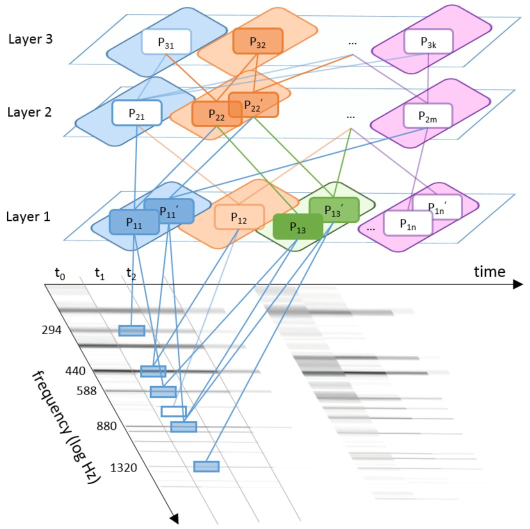

Figure 1. Compositional hierarchical model. Parts on the input layer correspond to signal components in the time-frequency representation. Parts on higher layers are compositions of lower-layer parts (denoted as links in the figure). A part may be contained in several compositions, e.g. P_{11} on the first layer is part of compositions P_{21}, P_{22} and P_{2m} on the second layer. Several depictions of the same part (e.g. part instances P_{11} and P'_{11}) denote several activations of the part on different locations (all instances of a part on a layer are marked with the same outlined color). Parts activated in t_1 are shown filled with color.

Any time-frequency representation can be used for the input layer, although logarithmic frequency spacing produces more compact models due to the relative nature of part compositions on higher layers (as described further on).

3.2.2 Subsequent layers

Higher layers of the model $\mathcal{L}_n$ contain sets of *compositions* - parts composed of parts from lower layers. Each composition can contain any number of parts from the lower layers (for clarity we only use two-part compositions to explain the model). A composition can be part of any number of compositions on higher layers. Compositions are denoted as links between parts in Figure 1.

Composition i on layer $\mathcal{L}_n$ can be formally defined as a structure containing parts from a layer below: a central part C, and a secondary part S. We name the parts forming a composition subparts. A composition can be defined as:

$$P_{n,i} = \{C_{n-1,j}, S_{n-1,k}, (\mu_{n,i}, \sigma_{n,i})\}, \tag{1}$$

where $C_{n-1,j}$ and $S_{n-1,k}$ are the central and secondary subparts from layer $n-1$, while $\mu_{n,i}$ and $\sigma_{n,i}$ define a Gaussian limiting the difference between locations of subpart activations (see definition of activation below). For clarity, we shall omit subscripts in the following equations and use P, C, S, μ and σ to denote a part and its components.

A composition is *activated* (propagates output to higher layers) when all of its subparts are activated. This strict condition can be softened with hallucination, as explained in section 3.3. Part activation is composed of two values: activation location L_P, which represents the location (frequency) at which the part is activated, and activation magnitude A_P, which represents the strength of activation. The location of part's activation is defined simply as the location of activation of its central subpart:

$$L_P = L_C. \tag{2}$$

Thus, central parts of compositions on different layers propagate their locations upwards through the hierarchy. The magnitude of activation is defined as:

$$A_P = tanh[G(L_C - L_S, \mu, \sigma) \cdot (A_C + A_S)], \tag{3}$$

where *tanh* stands for the hyperbolic tangent function that limits the magnitude to [0,1) and *G* represent the Gaussian that limits the difference in locations of the central part and the subpart according to μ and σ. As an example, $P_{2,2}$ in Figure 1 is defined as

$$P_{2,2} = \{P_{1,1}, P_{1,3}, (1200, 25)\}, \tag{4}$$

where μ and σ are given in cents. Therefore, it will be activated whenever $P_{1,1}$ and $P_{1,3}$ will be activated at locations approximately one octave (1200 cents) apart. Two such activations are shown in the figure, one at 294 Hz and one at 440 Hz.

3.3 Inference

The model can be used as a feature extractor over any desired dataset. An audio signal, transformed into a time-frequency representation, serves as input for layer $\mathcal{L}_0$. Activations are then calculated layer-by-layer according to Equations 2 and 3. Additionally, two biologically-inspired mechanisms govern the inference process and increase robustness of the model: *hallucination* and *inhibition*.

Before we define both mechanisms, we need to introduce the concept of coverage. Coverage $c(P, L_P)$ of part P active at location L_P represents all signal information (frequency components) covered by the part and its subtree of parts. It is calculated top-down from an active part to $\mathcal{L}_0$ as:

$$c(P, L_P) = \bigcup\{c(C, L_P), c(S, L_P + \mu)\}. \tag{5}$$

For the $\mathcal{L}_0$ layer, coverage is defined as the set of parts with positive activations $A_P > 0$, thus representing the set of covered frequency components. An example from Figure 1: the coverage of $P_{2,2}$ active at 294 Hz is the set of frequencies: $\{294Hz, 588Hz, 880Hz\}$.

3.3.1 Hallucination

Hallucination deals with filling-in the missing or damaged information in the signal and is implemented by enabling part activation in presence of incomplete input. The missing information in the signal can be replaced with knowledge encoded in the model during learning by allowing activations of parts most fittingly covering the information present. This allows the model to produce hypotheses in situations with no straight result. Hallucination also boosts alternative explanations of input data, thus increasing its explanation power and robustness.

Hallucination is governed by parameter τ_1 which can be defined per layer and modified during the inference. It changes the conditions under which a part may be activated. The default condition, as explained in section 3.2, is that activation of a part is possible when all of its subparts are active. With hallucination, a part P may be activated at location L_P, when the number of frequency components it covers $|c(P, L_P)|$, divided by the maximal number of components it may cover is larger than τ_1. For example, a τ_1 of 0.75 means that $\frac{3}{4}$ of all possible frequency components must be covered by the part for it to be activated. A τ_1 of 1 represents the default behavior.

3.3.2 Inhibition

The second biologically-inspired mechanism provides a balancing factor by reducing redundant activations, similar to lateral inhibition performed by the human auditory system. Inhibition refines the set of parts that yield competing hypotheses of the same fragments of information in the input signal. Parts with greater activation magnitudes are retained and weaker activations inhibited. Inhibition also reduces activations that result from noise in the signal.

Activation of part P at L_P is inhibited, when another part Q with activation L_Q on the same layer (or a set of parts) covers the same fragments of information in the input signal, but with higher activation. The condition can be expressed as:

$$\exists Q : \frac{|c(P, L_P) \backslash c(Q, L_Q)|}{|c(P, L_P)|} < \tau_2 \wedge A_Q > A_P, \tag{6}$$

where τ_2 defines the amount of inhibition. For example, a value of 0.5 means that activation of P is inhibited if half of its coverage is already covered by another, stronger part.

To sum up: inference yields a set of activations on all model layers by calculating activations considering hallucination and inhibition over all layers in a bottom-up order and over all time-frames of the input signal. Resulting activations represent model features and can be directly interpreted or used as inputs for discriminative tasks.

3.4 Learning

The model is learned in an unsupervised manner on a set of input signals. It is constructed layer-by-layer, similar to other deep architectures. The learning process relies on statistics of part activations, thus signal regularities are the driving force of the learning process.

When building layer $\mathcal{L}_n$, co-occurrences of activations of parts on $\mathcal{L}_{n-1}$ are observed. Compositions are formed from parts that frequently activate together at similar distances. All such parts are joined into compositions and added to the set of candidate compositions $\mathcal{P}$. When forming a composition of two frequently co-occurring parts, the part at the lower location represents the central part of the composition, while parameters μ and σ are estimated from all co-occurring activations of the two parts.

To reduce the number of compositions on each layer and keep only the most informative ones, the set of candidates $\mathcal{P}$ is refined. The goal of refinement is to reduce the number of compositions in the learned layer while maintaining sufficient coverage of information in the learning set.

Refinement is implemented with a greedy approach, where in each iteration, a part that contributes most to the coverage of information in the learning set, is selected and added to the layer. Refinement is concluded when one of the following two criteria are reached: a sufficient percentage of information in the learning set is covered (according to threshold τ_3), or no part remaining in the candidate set adds to the cumulative coverage of information. Algorithm 1 outlines the described approach.

Algorithm 1 Greedy approach for selection of compositions from the candidate set $\mathcal{P}$. Parts that add most to the coverage of information in the learning set are preferred. Function $perc$ calculates the percentage of information covered in the learning set by the given set of parts.

```
procedure REFINE($\mathcal{P}$)
    $prevCov \leftarrow 0$
    $coverages \leftarrow \emptyset$
    $\mathcal{L}_n \leftarrow \emptyset$
    repeat
        for $P \in \mathcal{P}$ do
            $coverages[P] \leftarrow perc(\mathcal{L}_n \cup P)$
        $Chosen \leftarrow \underset{P}{\text{argmax}}(coverages)$
        $\mathcal{L}_n \leftarrow \mathcal{L}_n \cup Chosen$
        $\mathcal{P} \leftarrow \mathcal{P} \setminus Chosen$
        if $coverages[Chosen] = prevCov$ then
            break //No added coverage - finish
        $prevCov \leftarrow coverages[Chosen]$
    until $prevCov > \tau_3 \vee \mathcal{P} = \emptyset$
```

3.5 Time

The model presented so far is time-independent. It operates on a frame-by-frame basis, where each time frame in the time-frequency representation is treated independently from others. Music, however, evolves in time and models that operate on such bases often fail to reflect the evolution of sound properly.

The proposed model can be naturally extended to include the time dimension. Our first step towards extending the model for time-dependent processing was to implement a short-time automatic gain control mechanism, similar to the automatic gain control contrast mechanism in human and other animal perceptual systems. The mechanism integrates part activations at similar locations over time. When a new part activation appears and persists, its value is initially boosted to accentuate the onset and later suppressed towards a stable value.

The mechanism operates on all layers, and has a short-term effect on lower layers, and longer-term effect on higher layers due to the upward propagation of activations. Its end effect is that it stabilizes activations, reduces noise, produces smoother model output and boosts event onsets.

3.6 Relation to Deep Architectures

The compositional hierarchical model shares a great deal of similarities with other deep learning architectures. The structure of the model is similar in terms of learning a variety of signal abstractions on several layers of granularity. The model is learned in an unsupervised generative manner, thus, no annotated data is needed. The learning procedure is similar: the structure is built layer-by-layer. The proposed model can also be used for discriminative tasks by observing activations of parts on multiple layers.

We see the biggest advantage of the proposed compositional hierarchical model over other established deep architectures in its transparency. As parts are compositions of subparts, their activations are directly observable and interpretable. This opens the model up for a variety of interesting usages, as it not only produces features that can be used, but features that can be interpreted and explained. In addition, the inhibition and hallucination mechanisms make it possible to produce alternative explanations of the input by suppressing the winning explanation and search for alternatives. In comparison to DBNs, where the outputs of each layer can only be interpreted during the evaluation, the proposed model offers a deeper analysis of results by tracing the higher layer activations over all layers and investigating the impact of each subpart.

Another difference in comparison to DBNs is the shareability and *relativeness* of parts, which both lead to a small number of parts needed to represent complex signals. A part in the proposed model is defined by the relative distance between its subparts and can thus be activated on different locations along the frequency axis. Thus, the large amount of layer units that DBNs need to cover the entire spectrum is not necessary and is replaced by reusing the available parts. This relativeness is accompanied with the concept of part shareability: parts on a layer may be shared by many compositions on higher layers. For example, a chord is composed of at least three pitches which may be identical in their representation in our model.

We show the usefulness of the described model's features in the evaluation section, where the model is used as both feature extractor and a classifier. Other possible applications exploiting the the model's structure are presented in section 5.

4. EVALUATION OF THE MODEL

The presented model is applicable to different MIR tasks. To present the model's usefulness, we built a three-layer model and evaluated it on two tasks: automated chord estimation and multiple fundamental frequency estimation.

The input layer was the same for both tasks. A constant-Q transform was used to transform music signals onto 345 frequency bins between 55 and 8000 Hz, with a step size of 50 ms and maximal window size of 100 ms. Two layers of compositions $\mathcal{L}_1$ and $\mathcal{L}_2$ were learnt as described previously. Due to the shareability of parts, the they contain only 23 and 12 parts respectively. The small number of parts in the model should mean that the model could be trained on a small learning set. We tested this hypothesis and trained the model on large and small datasets, and observed few differences. We were therefore able to build the model by using only a small set of 88 piano key samples as our learning set. We used the $\mathcal{L}_2$ layer for the task of multiple fundamental frequency estimation. For the task of automated chord estimation, we provided an additional $\mathcal{L}_3$ *octave-invariant* layer. The latter consists of 48 parts, where $\mathcal{L}_3$ activations correspond to octave-invariant activations of the $\mathcal{L}_2$.

4.1 Automated Chord Estimation

The time-independent model was tested for the task of automated chord estimation on the standard *Beatles* dataset, kindly provided by C. Harte. We used activations of the octave-invariant $\mathcal{L}_3$ layer as features and made the classification by using a hidden Markov model (HMM) with 24 states, each representing a chord, as described by [2]. We used cross-validation for evaluation; one album was used for HMM training and the rest of the dataset for estimation.

Our per-frame classification accuracy on the given dataset was 67.14 % with 0.1525 standard deviation. Compared to other per-frame approaches, we find our results slightly lower than for example [11], which also used per-frame technique for feature extraction. Nevertheless, we performed the evaluation as a proof of concept with time-independent feature extraction and no fine-tuning of the model, its learning, nor tuning of HMM parameters. We anticipate significant results increase by extending the model to time-dependent evaluation, using the whole hierarchy for classification and parameter tuning.

4.2 Multiple fundamental frequency estimation

The model was also tested for the task of multiple fundamental frequency estimation (MFEE) on the two subsets of MAPS (MIDI Aligned Piano Sounds) dataset, provided by [4]. Activations of layer $\mathcal{L}_2$ were directly used as fundamental frequency estimations with no further processing.

The following metrics were used for evaluation: per-frame precision and recall, precision and recall without penalising for octave errors, and pitch-class precision and recall. Results are shown in Table 1. Our results are significantly lower when compared to recent approaches, e.g. [14] which reported 77.1% classification accuracy on the subsets. However, the mentioned approach differs significantly from ours, as a severely larger dataset (approx. 4 times larger than the test sets) was used for training the support vector machine (SVM) classifier. In comparison, our model was trained only on a small set of piano key samples, so no parts of the MAPS dataset were used for training. It is also worth to mention that for this task, our model was used as a feature extractor and a classifier at the same time. We expect that accuracy would be improved if a classifier such as a SVM would be added on top of our model and would take features extracted on all layers for inputs. Our intention for this paper, however, is to present the general applicability of the model for multiple tasks and to avoid fine-tuning.

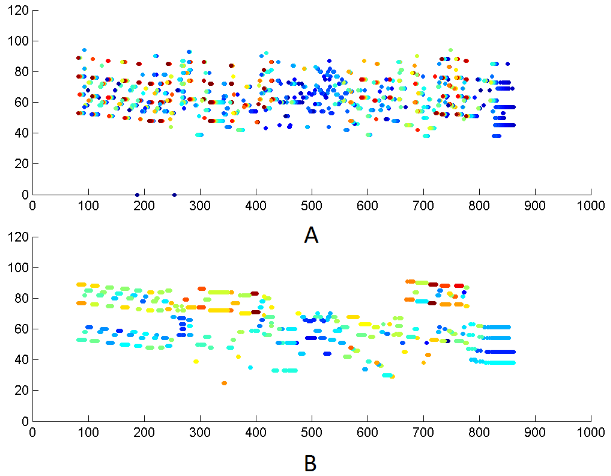

Figure 2. Hypotheses produced by our model for the task of multiple fundamental frequency estimation (A) and the ground truth (B). X axis represent time (in frames), and y midi pitches. Although the model produces many possible hypotheses per frame, only the ones with the highest magnitudes are used for comparison. Colors represent the magnitudes of activations in Fig. A or the MIDI velocity in Fig. B.

Table 1. Classification accuracy (CA) using all hypotheses provided by the model, precision (Pr) and recall (Re) values over a part of the MAPS dataset. Results without penalising octave errors and considering only pitch classes are marked with O and PC subscripts respectively.

Folder name	CA	Pr	Re
$AkPnBcht$	56.53 %	19.40 %	55.69 %
$AkPnBsdf$	66.17 %	22.05 %	61.27 %
$AkPnBcht_O$	67.08 %	35.37 %	64.55 %
$AkPnBsdf_O$	71.16 %	46.10 %	68.83 %
$AkPnBcht_{PC}$	86.20 %	51.83 %	86.59 %
$AkPnBsdf_{PC}$	88.23 %	58.68 %	70.99 %

5. OTHER APPLICATIONS OF THE MODEL

Our intention with developing the proposed model is to make an interpretable model that overcomes some of the limitations of DBNs and can be used for tackling various MIR tasks. Its transparency, however, also makes other uses of the model possible.

The hierarchical approach presented in this paper fits well with the hierarchical structure of music in frequency as well as in time domains. Each part of the model represents an explainable entity (e.g. tone partial, pitch, chord). In contrast to the DBNs, each part of the model can be visualized. Visualization not only exposes the layered structure of the model, but also discloses information processed by the observed part and its influence on other parts and their activations. This insight into the music signal can be used in several scenarios — music visualization, music analysis and music synthesis.

We have developed a real-time visualization of the model, enabling deeper understanding of the processed information. When observing an inferred audio signal, the output of all layers of the model is presented by visualizing activations of parts. This insight enables detailed analysis of each event in the music signal and may bring additional event details to light. For example, a chord inversion can be observed by looking into the activated subtree of the chord from top layers to bottom-ones. Thus, visualization of our model offers an innovative user interface for music analysis.

The transparency of the model can also be exploited for music processing and synthesis. Parts across all layers form a variety of harmonic structures, and can be used for signal manipulation and synthesis. By activating a set of parts at different locations, a new spectral representation is produced. Although the interface may not provide a sufficient amount of features for a standalone music performance, it can be used as a sound generator in a combination with a music instrument, e.g. a MIDI keyboard. The interface thus serves as an advanced tool for spectral modification, while the instrument provides the interface for performance.

6. CONCLUSION AND FUTURE WORK

This paper presents a compositional hierarchical model as an alternative to deep learning architectures based on neural networks. The model shares a great deal of similarities with other deep architectures, including a multi-layer structure, unsupervised generative learning and suitability for discriminative tasks. Furthermore, the white-box structure of the model offers new utilizations of the model. We highlighted three possible applications: feature extraction for MIR tasks, music visualization and music analysis/synthesis.

The model's internals rely on findings in the fields of neurobiology and cognitive sciences. By implementing biologically-inspired mechanisms into the model, we made an attempt to build a model which partially resembles a subset of functions of the human auditory system. We intend to retain and further develop this aspect of the model with an intention to bring the computational modeling closer to human auditory perception.

The paper presents an initial development of our model. We plan to further extend it with the focus on temporal modeling. Parts can namely be extended into the time domain, thus bringing their activations closer to event-based modeling. We also plan to tackle temporal tasks, such as onset detection, as well as beat tracking and tempo estimation. The proposed model is also going to be evaluated for pattern analysis of symbolic data, including discovery of repeated themes, and symbolic melodic similarity.

7. REFERENCES

[1] Eric Battenberg and David Wessel. Analyzing Drum Patterns using Conditional Deep Belief Networks. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 37–42, 2012.

[2] Juan P. Bello and Jeremy Pickens. A robust mid-level representation for harmonic content in music signals. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 304–311, London, 2005.

[3] J. Stephen Downie, Andreas F. Ehmann, Mert Bay, and M. Cameron Jones. The Music Information Retrieval Evaluation eXchange: Some Observations and Insights. In Wieczorkowska A.A. and Ras Z.W., editors, *Advances in Music Information Retrieval*, pages 93–115. Springer-Verlag, Berlin, 2010.

[4] Valentin Emiya, Roland Badeau, and Bertrand David. Multipitch Estimation of Piano Sounds Using a New Probabilistic Spectral Smoothness Principle. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6):1643–1654, August 2010.

[5] Philippe Hamel and Douglas Eck. Learning Features from Music Audio with Deep Belief Networks. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 339–344, 2010.

[6] Eric J. Humphrey, Juan P. Bello, and Yann LeCun. Moving beyond feature design: deep architectures and automatic feature learning in music informatics. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Porto, 2012.

[7] Honglak Lee, Peter Pham, Yan Largman, and Andrew Y. Ng. Unsupervised feature learning for audio classification using convolutional deep belief networks. In *Advances in Neural Information Processing Systems*, pages 1096–1104, 2009.

[8] Aleš Leonardis and Sanja Fidler. Towards scalable representations of object categories: Learning a hierarchy of parts. *Computer Vision and Pattern Recognition, IEEE*, pages 1–8, 2007.

[9] Abdel-rahman Mohamed, George E. Dahl, and Geoffrey Hinton. Acoustic Modeling using Deep Belief Networks. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(1):14–22, 2010.

[10] Nicola Orio. Music Retrieval: A Tutorial and Review. *Foundations and Trends® in Information Retrieval*, 1(1):1–90, 2006.

[11] Helene Papadopoulos and Geoffroy Peeters. Large-case Study of Chord Estimation Algorithms Based on Chroma Representation and HMM. *Content-Based Multimedia Indexing*, 53-60, 2007.

[12] Eric M. Schmidt and Youngmoo E. Kim. Learning Rhythm and Melody Features with Deep Belief Networks. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 21–26, 2013.

[13] Erik M. Schmidt and Youngmoo E. Kim. Learning emotion-based acoustic features with deep belief networks. In *2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 65–68. IEEE, October 2011.

[14] Felix Weninger, Christian Kirst, Bjorn Schuller, and Hans-Joachim Bungartz. A discriminative approach to polyphonic piano note transcription using supervised non-negative matrix factorization. In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 6–10, Vancouver, 2013.

[15] Dong Yu and Li Deng. Deep Learning and Its Applications to Signal and Information Processing [Exploratory DSP. *IEEE Signal Processing Magazine*, 28(1):145–154, January 2011.