Towards Deep Learning Strategies for Transcribing Electroacoustic Music

Matthias Nowakowski[1], Christof Weiß[2], and Jakob Abeßer[3]

[1] Media Informatics, University of Applied Sciences, Düsseldorf
[2] International Audio Laboratories Erlangen
[3] Semantic Music Technologies Group, Fraunhofer IDMT, Ilmenau
matthias.nowakowski@gmail.com

Abstract. Electroacoustic music is experienced primarily through auditory perception, as it is not usually based on a prescriptive score. For the analysis of such pieces, transcriptions are sometimes created to illustrate events and processes graphically in a readily comprehensible way. These are usually based on the spectrogram of the recording. Although the manual generation of transcriptions is often time-consuming, they provide a useful starting point for any person who has interest in a work. Deep-learning algorithms that learn to recognize characteristic spectral patterns using supervised learning represent a promising technology to automatize this task. This paper investigates and explores the labeling of sound objects in electroacoustic music recordings. We test several neural-network architectures that enable classification of sound objects using musicological and signal-processing methods. We also show future perspectives how our results can be improved and applied to a new gradient-based visualization approach.

Keywords: electroacoustic music · acousmatic music · transcription · deep learning

1 Introduction

Scientific discourse is based on an intersubjectively accessible object and its representations. Musicology usually treats music as sound or score. Especially when studying electroacoustic art music, the approach must always be an auditive one since the peculiarity of this type of music is to fix the sound and not its prescriptions. Sound material is either synthetically produced or consists of electronically processed recordings [1,2]. Thereby, timbre and its temporal progression become important structural elements, in contrast to harmony [3] or metre. In order to make this music comparable, researchers often create transcriptions, which are mostly colorations or linguistic annotations of spectrograms [4].

There are few attempts to automate this process for electroacoustic music. In particular, the lack of a uniform nomenclature to describe sound objects is an issue to be discussed. Analyzing such properties with signal-processing algorithms has been addressed sparsely [5].

In recent years neural networks have shown promising results for tasks such as genre classification or chord recognition, etc., within the analysis of tonal music [6]. In particular, end-to-end visualization techniques, as well as better scalability on dataset size make this state-of-the-art technology interesting for exploring this task. The few papers dealing with machine learning applied to electroacoustic music either treat this subject superficially, or some advantages where not yet widely accessible at the time of publication, like widely accessible programming libraries or computing infrastructure [7].

Such an endeavor can not only be fruitful for the musicological discourse. Transcriptions are also an communicative device that can be used in an explorative way (find morphologies which were not heard before, reveal macro-forms) or explanatively (backing up individual transcriptions by technical means) and can thus enhance accessibility.

2 Previous Work

In the past, common Music Information Retrieval (MIR) techniques were used to analyze and visualize certain characteristics of elecotracoustic pieces and are also implemented in software [8, 9]. Also, further features that capture style-relevant properties could be used for electroacoustic music as well [3, 5]. But the interest in using expensive machine- learning algorithms has been rather low so far, although deep learning approaches provide state-of-the-art results in different music related tasks [6].

In a recent study, Collins [10] made thorough analyses of electroacoustic pieces after the release of a large corpus, which is available online.[4] He used fully-connected networks on previously extracted features to estimate publication years, but was not pursuing this approach in subsequent publications [11] since a k-Nearest-Neighbor algorithm outperformed the neural networks in accuracy.

Klien et al. critically discuss the use of machine learning from an aesthetic standpoint [12]. In their view, fully automated annotations are not able to overcome the semantic gap between the "signal" and the "meaning," since electroacoustic (or acousmatic) music tries do defy the definition of music itself. Any algorithm used for analysis therefore should not attempt to "understand" music. Despite their position, we agree that a human annotator is needed to make reasonable assertions about musical structure. In contrast, one could consider the complexity of the task to be particularly suitable for a deep-learning approach.

Using interactive hierarchical clustering, Guluni et al. [13, 14] let experts categorize sound events of different timbres of a synthetic data set by putting its results into a feedback loop. The authors use a Support Vector Machine (SVM) classifier, which is fed with the feature coefficients. Both monophonic and polyphonic compositions had results with F-measures ≥ 0.85.

Given a sound scene (being musical or not) in which sounds change characteristics according to auditory distance or distortion, form coherent textures

[4] http://www.ubu.com/sound/electronic.html

with other sounds, or split from them gradually, it might be helpful to view this task as one lying close to the tasks of sound event detection and sound scene analysis. One of the main challenges is high intra-class variance due to the quality of sounds, which may not be clearly separable from each other and appear in multi-source environments [15]. As shown in the regularly held "Detection and Classification of Acoustic Scenes and Events" (DCASE) challenges, best results are produced by neural networks which are trained on some type of a spectrogram representation [16, 17]. From this point of view, deep learning seems like a viable approach for elecotracoustic music. The main problem is to develop a suitable set of labels, which show satisfactory results in classification tasks before temporal sound event detection can even take place. Using deep learning also gives the possibility to employ methods for interpreting deep neural networks to generate visualizations [18]. This might show what portions of a spectrogram were actually important while training on sound event classification and so gives visual cues to examine the dataset.

3 Data & Annotation

In this section, we describe the creation and annotation of a dataset of electroacoustic music excerpts. Since there is no consistent or commonly used nomenclature for categorizing sound objects, analysis frameworks can help to develop labels that can be used and understood by musicological experts. For better comparison with previous approaches, we use label names in accordance with commonly used features, which are used in other classification algorithms. Although there are some historically relevant frameworks like Pierre Schaeffer's *Typomorphology* [19], we draw our inspiration for the labels used here from Denis Smalley's *Spectromorphology* [20]. He developed this framework as a tool for describing sound shapes, i. e. the unit of spectral content and their development in time, based on aural perception. Adopting this viewpoint can be helpful to identify such sound-shapes in a spectrogram which is our base baseline feature.

We chose the five descriptors *flux*, *spread*, *noise*, *density*, and *realness* as attributes to describe both the static and dynamic aspects of spectromorphological expectation. For each attribute, the extreme values (0/1) represent poles within a description space.

Flux 0: Stationary; 1: Fluctuating
Spread 0: Narrow spectral range; 1: Wide spectral range
Noise 0: Harmonic; 1: White Noise
Density 0: Single event; 1: Multiple events (uncountable)
Realness 0: Synthetic; 1: Real world sound source

Flux and *density* were selected to reflect the development of a sound event over time. In contrast, *spread* and *noise* describe static sound characteristics. All attributes can be combined to form a label set to provide more complex descriptions of sound events. Each attribute in a label is represented by its initial letter. We obtain 32 possible classes from all combinations of the five

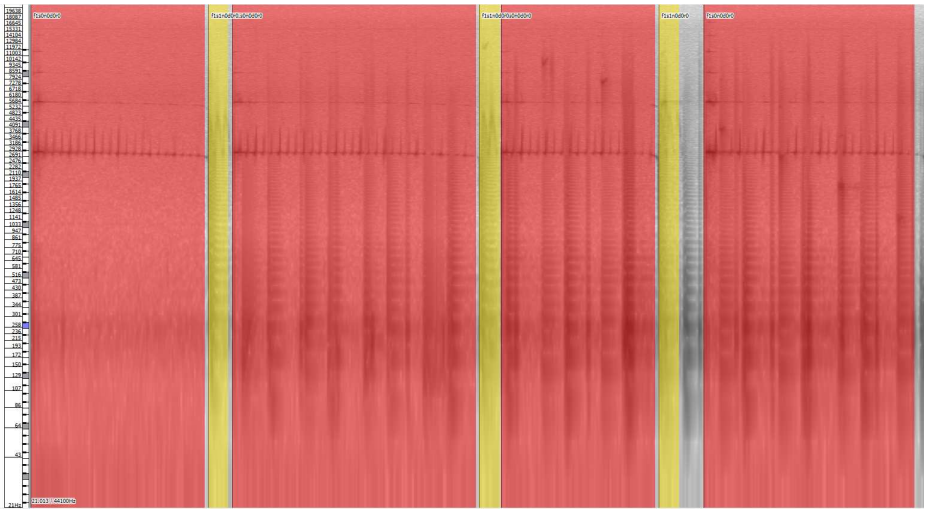

Fig. 1: Example annotation in the log-spectrogram of the first 12 seconds of the movement *Polyrythmie* in *Vibrations Composés* by François Bayle. Red boxes represent recurring/similar events annotated as *f1s0n0d0r0*, while yellow boxes represent recurring/similar events annotated as *f1s1n0d0r0*. Although more events can be seen (e.g. the dark colored band in the lower half of the spectrogram), all frequencies are used as an input feature for each annotated event as indicated by the boxes.

attributes. For instance, *f0s0n1d1r0* represents a stationary and narrow sound, which is very noisy, has a high density and a synthetic sound characteristic. As an analogy, one could think of a pass-filtered noise band. Similarly, we can define four separate classes by combining only two attributes. An example of such an annotation w.r.t. *spread* and *noise* could be *s0n1*, which defines a filtered noise-like sound without specifying its temporal characteristics. On the one hand, this way of choosing attributes to form new label sets allows to refine classes during the annotation process. On the other hand, a binary attribution can lead to fuzzy class boundaries, so that event labeling becomes imprecise. For instance, labeling *density* of a drum roll may diverge into labeling each event itself or the whole texture depending on the event frequency. While each event is probably static, the texture could gain fluctuating characteristics due to playing style or instrument manipulation, like a timpani glissando. Therefore, during the annotation process, we focused on sound objects that can be clearly defined by the selected attributes. Note that silence can not be reflected.

The compiled dataset consists of excerpts of 29 electroacoustic pieces from the 1960s to the early 2000s. Longer pieces were cut off at the 5 minutes mark, while whole pieces were taken if they were shorter or slightly longer than 5 Minutes. This adds up to a total duration of 2.5 hours.

Each relevant sound event was then annotated manually including the attack and end of release. Since almost all recordings are polyphonic, some sound events may appear in multiple longer events. This leads to a total of 3.7 hours of extracted material. 3016 separate events were annotated ranging from 0.05 seconds to 4.5 minutes. We enlarged the dataset using data augmentation techniques. To this end, we applied mild pitch shifting using step sizes between minus and plus 2 semitones in order to not distort the spectral characteristics. In total, the dataset contains 18.5 hours of segmented and augmented material.

Since some classes are stronger represented than others, all the extracted events were resampled to the mean duration of all 32 classes. Resampling on these classes also scales to all other possible label sets. Longer classes were shortened by randomly deleting events. For shorter classes, events were duplicated in turn with slight random modifications to the signal.

In our experiment, we repeat a random dataset split into training, validation, and test set three times using a split ratio of 60% / 20% / 20% (by number of events). We ensure that the events in the three evaluation sets come from different recordings to reduce overfitting.

4 Experiments

In this paper we focus on reporting results from configurations made with a 4-class label set consisting of the attributes *spread* and *noise* to investigate the impact of a combination of static spectral attributes at first. By using a label set consisting of two attributes, we reduce chances of wrong labeling and have a more manageable number of attributes to compare. For all experiments, the following labels were used: *s0n0*, *s0n1*, *s1n0*, *s1n1*. The deep-learning architectures were implemented using the Keras framework[5], wheras feature extraction algorithms were implemented after [21] or used directly through the librosa library.[6] For our experiments we have focused primarily on the performance of the classification.

4.1 Metrics

Test F1-score This measure is used to score the overall classification of a sound event. The F1-score equals the harmonic mean between precision and recall. If classification is done on multiple patches of an event, the mean is computed.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{1}$$

Training validation accuracy difference (ΔAcc) The accuracy for both training and validation set is computed during the training itself. The goal is to have

[5] https://keras.io/
[6] https://librosa.github.io/librosa/

training and validation accuracy as close as possible. A lower value for ΔAcc means less overfitting.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$\Delta Acc = |Acc_{train} - Acc_{val}| \tag{3}$$

All metrics are averaged over the number of folds in the respective configuration. Early stopping on validation accuracy was employed for all the deep-learning experiments after 20 epochs due to usually very short training (one digit numbers of epochs) when using early stopping on the validation loss.

4.2 Baseline

Because we use a self-developed annotation scheme and dataset for this paper, the definition of a baseline performance is challenging. For four classes, we expect the random baseline to be $P(4) = 0.25$. To compare our results to classical approaches, we used a Random Forest (RF) classifier with 100 estimators. For this we used "spectral variation" (also called "spectral flux") for *flux* and "spectral spread" for *spread* as described in [21]. *Noise* is described by the absence of the estimated harmonicity according to a detected fundamental frequency. As a feature for *density* we used the residual part of Harmonic–Percussive–Residual Separation [22]. In lack of a descriptor for *realness* we just used the mean of the mel-spectrogram. All coefficients were computed on the complete event by averaging over time.

Using this classifier also gives the possibility to determine importances of all feature coefficients. Although using five features we expect higher importances for the ones corresponding to our attributes.

4.3 Convolutional Neural Network (CNN)

We now want discuss the CNN architectures used for our experiments. As input representation, we chose mel-spectrograms to keep the aspect ratio of sound shapes independent of vertical position. Time–frequency transformations were made from wavefiles with a sample rate of 22050 Hz using a window size of 1024 frames, a hop length of 512 frames and a vertical resolution of 96 bins. Each spectrogram was then cut into patches of 100 frames (around 2 seconds) with 50% overlap of each subsequent patch. Shorter events were padded by adding low positive noise (up to 10^{-4}) on the spectrogram to reach the desired minimum patch length. Values where scaled by appling zero-mean unit-variance normalization.

For the CNN configurations we chose to use a shallow version of the VGG-Network with 7 Layers [23]. Here, we compare architectures using 2D and 1D convolutional layers. Each architecture consists of two convolutional layers with 32 kernels, followed by two layers with 64 kernels. After each convolution, we

applied batch normalization and max pooling. Classification was done with subsequent fully connected layers using one dense layer with 512 nodes as well as a final dense layer with four output nodes and softmax activation functions. We used dropout (0.25), ℓ_2 kernel regularization (0.01) after the first dense layer, as well as adding Gaussian noise (0.1) on the input in order to regularize the model and reduce overfitting. For the 2D convolutional architecture, we use 3x3 kernels and apply global average pooling before the Fully connected Layer (FCN). Accordingly, we use convolution kernels of size 3 for the 1D architectures before the CNN output is flattened and forwarded to the fully connected layers. Both architectures were chosen due to their different approach on computing features of the given input. While the 2D convolution is able to detect position invariant patterns, 1D convolution focuses on local patterns in a sequence.

4.4 Convolutional-Recurrent Neural Network (CRNN)

For the CRNN, a bi-directional Gated Recurrent Unit (GRU) layer with 32 units was added after the CNN processing for temporal modeling. We chose GRU over Long Short-Term Memory (LSTM) units, because of faster model convergence while showing similar performance to LSTM [24]. The first advantage of using CRNN over CNN alone is that this architecture can better model long-term dependencies. Secondly, such a network architecture can be trained with variable-length input data, while CNNs require fixed-size input. However, we observed a strong model overfitting when training CRNNs on full spectrograms of a piece. Therefore, we will focus on reporting results from the CNN model trained with a fixed input tensor size first. Then, we evaluate, whether the classification results improve if the CRNN is instead initialized with the trained parameters from a CNN model.

5 Results

The feature importance values for the baseline experiment in Table 2 show slight tendency towards the mel spectrogram with 0.22, while the noise feature had the least impact on the classification with 0.16. Importances of the remaining features are balanced.

Comparing our approach (Table 1) with the baseline performance, deep learning improves classification results to some extent. Only using CNN layers for computation, 2D convolution gave best results with an F1-value of 0.335 over 1D convolution with 0.315. Accuracy differences are still relatively high so that we can observe some amount of overfitting. Taking all folds into account, the standard deviation over all accuracy differences in CNN 2D with 0.14 is relatively high as compared to CNN 1D with 0.089. For each fold, a higher ΔAcc usually correlates with higher numbers of training epochs, being a maximum of 98 epochs for CNN 2D and 38 for CNN 1D (Table 3).

Adding the GRUs to the architectures and initializing weights with the aforementioned models results in improved classification results. Especially, CRNN

Table 1: Results of the 4-class classification experiments

	Architecture	*F1*	ΔAcc
Random Baseline	-	0.25	-
Shallow Classifier	Random Forest, 100 Estimators	0.27	-
CNN 2D	2 x Conv 2D 32, 2 x Conv 2D 64, 512 FCN	0.335	0.152
CNN 1D	2 x Conv 1D 32, 2 x Conv 1D 64, 512 FCN	0.315	0.207
CRNN 2D	2 x Conv 2D 32, 2 x Conv 2D 64, Bidirectional GRU 32, 512 FCN	0.362	0.112
CRNN 1D	2 x Conv 1D 32, 2 x Conv 1D 64, Bidirectional GRU 32, 512 FCN	0.385	0.022

Table 2: Feature importance values for the baseline experiment (Random Forest classifier)

flux	*spread*	*noise*	*density*	*mel*
0.21	0.21	0.16	0.21	0.22

1D outperforms all experiments with a F1-value of 0.385, increasing by 0.07, whereas the F1-value for the CRNN 2D increases just by 0.027 to 0.362. ΔAcc decreases for both experiments. For CRNN 1D by 0.185 and 0.04 for CRNN 2D. Also the maximum training time decreased for for both experiments being it 12 epochs for CRNN 2D and 14 epochs for CRNN 1D. Overall, a high F1-value correlates with a lower ΔAcc pointing to less overfitting.

6 Discussion

In this paper, we presented a musicologically informed approach for sound object classification of electroacoustic music. Using deep learning, we could show that some improvement could be achieved by architectures more sensitive to sequential data, which can facilitate classifying data as described by our morphological labels. Despite reducing the label space, feature importance values for the selected attributes *spread* and *noise* do not have any significant impact on the classification in the baseline experiment. *Noise* even had the lowest importance. This shows that the semantic implications of the chosen attributes are not transferable to common descriptors easily, so that more complex feature sets might be required.

Although we can see that CNN 2D achieved better classification results, using CNN 1D resulted in constant generalization throughout the folds (Table 3). This indicates that feature generalization depend less on position-invariant sound shapes than on the vertical integrity of the spectrum. Or at least more spectral context is needed than previously expected. One approach to validate

Table 3: ΔAcc for each fold in CNN 2D and CNN 1D

	CNN 2D		*CNN 1D*	
	Epochs	ΔAcc	*Epochs*	ΔAcc
Fold 1	3	0.123	16	0.265
Fold 2	98	0.361	38	0.275
Fold 3	1	0.028	3	0.081
Standard Dev.	-	0.14	-	0.089

this in future experiments is to apply horizontally-shaped CNN filters instead of symmetrical ones to incorporate a larger spectro-temporal context. The importance of temporal succession over the isolated position in the spectrum is then pronounced by the improved scores using CRNN.

Since there are still many questions and many configurations to be tested, this paper is merely a suggestion and baseline on further investigation into this field and even this approach can be still evaluated on more or different attributes, features and architectures. In general, results lagged far behind our initial hopes, which can be attributed to our more explorative approach of this problem.

With regard to the transcription of sound objects, the outcome of such experiments can be used for visualization, using e.g. gradient-based network analysis algorithms. These show portions of the spectrogram, which were relevant for the classification. We suspect these means to be helpful for detecting and displaying sound objects in the spectrogram. For our purposes, we tried layer-wise relevance propagation (LRP) [18] which resulted in relevancy maps which are merely hints to what the network actually learns. But the classification scores are quite low so that mappings at this point are mostly inconclusive and still have to be evaluated.

While developing and evaluating the experiments we noticed some issues that we want to address and propose some future solutions.

Labeling approach During the annotation procedure, only one person familiar with electroacoustic music worked on the labels. Thus, no validation could be made. To reduce the bias of the annotator, a group of people could cross-check their annotations. In different classification tasks, as e.g. discriminating cats and dogs, we can expect human experts to classify all samples right. Such bias values for a complex task like the one presented here do not exist so that cross-checking labels could also be a basis for more research. In addition, one could think about continuous values for annotating attributes. This could lead to embeddings of such sound objects which might help constructing new labels or label families.

Dataset size The dataset used in this paper is relatively small. Therefore, more elaborate transfer-learning techniques [25, 26] could be employed following the assumption that suitable low-level features can be extracted from different music related tasks [27] or even different domains such as images [28]. Beside feature

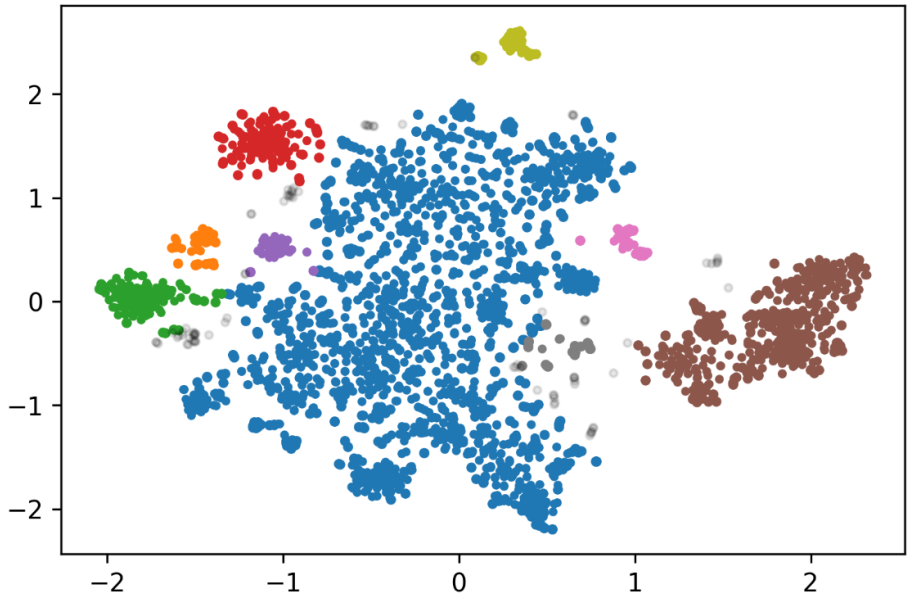

Fig. 2: Visualization using t-SNE for dimensionality reduction and DBSCAN for clustering. Each colored cluster consists of most segments of one piece used in the dataset. The blue cluster consists of segments of all other pieces.

transfer, one could also apply multitask-learning if labels for both source and target task are available [25]. The main idea is to train source and target task simultaneously, provided that both tasks are similar, to extract shared features. Even in the case of negative transfer, analyzing predicted targets could help in investigating the most helpful features. For a source task e.g. electroacoustic sound sources can be used such as *electronic*, *environment*, *instrument*, or *singing* while the target task labels remain morphological attributes.

Studio effect To validate the chosen labels, we wanted to see if unsupervised machine-learning techniques could help to figure out if some consistency in the data points can be found beyond our chosen descriptors. To this end, over 160 features were extracted for all segments according to [29] which where used in electroacoustic music related tasks in [13,14]. Using t-SNE [30] for dimensionality reduction and DBSCAN for clustering, we found that the studio effect (the effect of the production conditions for the feature extraction) had large impact on the clustering. In Fig. 2, each colored cluster consists of most segments of one piece used in the dataset, except blue which consists of segments of all other pieces. Grey transparent points are considered to be noisy data points by the clustering algorithm. To avoid this effect, we would need a more uniform dataset composed

by some experts especially for that task, such as in [13,14][7], to avoid this effect. This is not a statement about the performance of our descriptors. It rather is an example of a problem we came across and which we have to face when designing a dataset using samples from a field of music which is highly dependent on its medium.

Acknowledgements. This work has been supported by the German Research Foundation (AB 675/2-1, MU 2686/11-1). The International Audio Laboratories Erlangen are a joint institution of the Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) and Fraunhofer Institut für Integrierte Schaltungen IIS.

References

1. Stroh, W.M.: Elektronische Musik. Handbuch der musikalischen Terminologie 2, Steiner-Verlag, Stuttgart (1972)
2. Beiche, M.: Musique concrète. Handbuch der musikalischen Terminologie 4, Steiner-Verlag Stuttgart, (1994)
3. Weiß, C., Müller M.: Quantifying and Visualizing Tonal Complexity. In: Proceedings of the 9th Conference on Interdisciplinary Musicology (CIM), pp. 184–187, Berlin (2014)
4. Erbe M.: Klänge schreiben: die Transkriptionsproblematik elektroakustischer Musik. Apfel, Vienna (2009)
5. López-Serrano, P., Dittmar, C., Müller M.: Mid-Level Audio Features Based on Cascaded Harmonic-Residual-Percussive Separation. In: Proceedings of the Audio Engineering Society AES Conference on Semantic Audio, Erlangen (2017)
6. Purwins, H., Li, B., Virtanen, T., Schlüter, J., Chang S.-Y., Sainath, T.: Deep Learning for Audio Signal Processing. IEEE Journal of Selected Topics in Signal Processing 14/8, 1–14 (2019)
7. Pons, J.: Neural Networks for Music: A Journey Through Its History, https://towardsdatascience.com/neural-networks-for-music-a-journey-through-its-history-91f93c3459fb (2018)
8. Couprie, P.: Methods and Tools for Transcribing Electroacoustic Music. In: International Conference on Technologies for Music Notation and Representation – TENOR'18, pp. 7-16, Montréal (2018).
9. Park, T.H., Li, Z., Wu, W.: Easy Does It: The Electro-Acoustic Music Analysis Toolbox. In: Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR 2009), pp. 693–698, Kobe (2009)
10. Collins, N.: The UbuWeb Electronic Music Corpus: An MIR investigation of a historical database. Organised Sound 20/1, pp. 122–134 (2015)
11. Collins, N., Manning, P., Tarsitani, S.: A New Curated Corpus of Historical Electronic Music: Collation, Data and Research Findings. Transactions of the International Society for Music Information Retrieval 1/1, pp. 34–55 (2018)
12. Klien, V., Grill, T., Flexer, A.: On Automated Annotation of Acousmatic Music. Journal of New Music Research 41/2, 153–173 (2012).

[7] We have requested this dataset, but unfortunately it was no longer provided by the creators.

13. Gulluni, S., Essid, S., Buisson, O., Richard, G.: An Interactive System for Electro-Acoustic Music Analysis. In: 12th International Society for Music Information Retrieval Conference (ISMIR 2011), pp. 145–150, Miami (2011)
14. Gulluni, S., Essid, S., Buisson, O., Richard, G.: Interactive Classification of Sound Objects for Polyphonic Electro-Acoustic Music Annotation. AES 42nd International Conference, Ilmenau (2011)
15. Virtanen, T., Plumbley, M.D., Ellis, D.P.W.: Computational analysis of sound scenes and events. Springer Verlag, Cham (2018)
16. Mesaros, A., Heittola, T., Diment, A., Elizalde, B., Shah, A., Vincent, E., Raj, B., Virtanen, T.: DCASE 2017 Challenge setup: Tasks, datasets and baseline system. DCASE 2017 - Workshop on Detection and Classification of Acoustic Scenes and Events (2017)
17. Adavanne, S., Virtanen, T.: A Report on Sound Event Detection with Different Binaural Features. DCASE2017 Challenge (2017)
18. Alber, M., Lapuschkin, S., Seegerer, P., Hägele, M., Schütt, K.T., Montavon, G., Samek W., Müller, K.-R., Dähne, S., Kindermans, P.-J.: iNNvestigate neural networks! CoRR (2018)
19. Thoresen, L., Hedman, A.: Spectromorphological Analysis of Sound Objects: An Adaptation of Pierre Schaeffer's Typomorphology. Organised Sound 12/2, pp. 129–141, Cambridge (2007)
20. Smalley, D.: Spectromorphology: Explaining Sound-shapes. Organised Sound 2/2, pp. 107–126, Cambridge (1997)
21. Peeters, G.: A Large Set of Audio Features for Sound Description (Similarity and Classification) in the CUIDADO Project, http://recherche.ircam.fr/anasyn/peeters/ARTICLES/Peeters_2003_cuidadoaudiofeatures.pdf (2004)
22. Drieger, J., Müller M., Disch S.: Extending Harmonic-Percussive Separation of Audio Signals. In: Retrieval Conference (ISMIR 2014), pp. 611–616, Taipei (2014)
23. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. ILCR (2015)
24. Chung, J., Gülçehre, C., Cho, K., Bengio, Y.: Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. NIPS 2014 Deep Learning and Representation Learning Workshop (2014)
25. Pan, S.J., Yang, Q.: A Survey on Transfer Learning. IEEE Transactions on Knowledge and Data Engineering 22/10, pp. 1345–1359 (2010)
26. Torrey, L., Shavlik, J.: Transfer Learning. In: Handbook of Research on Machine Learning ,Algorithms, Methods, and Techniques, pp. 242–264, IGI-Global (2009)
27. Choi, K., Fazekas, G., Sandler, M.B., Cho, K.: Transfer Learning for Music Classification and Regression Tasks. In: Proceedings of the 18th ISMIR Conference, pp. 141–149, Suzhou (2017)
28. Grzywczak, D., Gwardys, G.: Deep Image Features in Music Information Retrieval. Intl. Journal of Electronics and Telecommunications 60/4, 321–326 (2014)
29. Essid, S., Richard, G., David, B.: Musical Instrument Recognition by Pairwise Classification Strategies. IEEE Transactions on Audio, Speech, and Language Processing 14/4, 1401–1412 (2006)
30. van der Maaten, L., Hinton, G.: Visualizing Data using t-SNE. Journal of Machine Learning Research 9, 2579–2605 (2008)