A DISCRIMINATIVE APPROACH TO POLYPHONIC PIANO NOTE TRANSCRIPTION USING SUPERVISED NON-NEGATIVE MATRIX FACTORIZATION

Felix Weninger[1], *Christian Kirst*[1,2], *Björn Schuller*[1], *Hans-Joachim Bungartz*[2]

[1] Machine Intelligence & Signal Processing Group, MMK, [2] Department of Informatics
Technische Universität München, Germany

ABSTRACT

We introduce a novel method for the transcription of polyphonic piano music by discriminative training of support vector machines (SVMs). As features, we use pitch activations computed by supervised non-negative matrix factorization from low-level spectral features. Different approaches to low-level feature extraction, NMF dictionary learning and activation feature extraction are analyzed in a large-scale evaluation on eight hours of piano music including synthesized and real recordings. We conclude that the proposed method delivers state-of-the-art results and clearly outperforms SVMs using simple spectral features.

Index Terms— Transcription, sparse coding, non-negative matrix factorization, music information retrieval

1. INTRODUCTION

Transcription of polyphonic music is one of the key applications in music information retrieval [1, 2], as it converts unstructured waveform data to a symbolic, musically meaningful representation. In this work, we formulate the problem of polyphonic music transcription as joint onset detection and multi-pitch estimation, where note onsets have to be detected along with the correct pitch.

A popular approach to multi-pitch estimation and polyphonic music transcription is based on non-negative matrix factorization (NMF) applied to the spectrogram, modeling the short-time spectra of the signal frames as linear combinations of dictionary atoms with non-negative activation coefficients [3–7]. In many approaches, both the atoms and their time-varying activation coefficients are estimated jointly by the expectation-maximization principle in an unsupervised fashion [8]. However, without further constraints it cannot be guaranteed that the atoms resulting from this procedure have an actual musical meaning (e. g., representing different pitches of different instruments). As a result, interpretation of the atoms and their activations, and hence transcription based on the NMF decomposition itself, can become challenging. Introducing musical constraints into NMF, such as in [9, 10], appears to be promising, yet from the results it seems that transcription using purely NMF (and hence, maximum likelihood) based techniques remains a notoriously difficult task [9, 10].

As an alternative, discriminative approaches [11–15] have been proposed, delivering most robust results [13]. In discriminative music transcription, a classifier is trained on positive and negative examples corresponding to signal frames where a given pitch is present or absent. This can be done in a 'one-versus-all' training paradigm for pitch-specific classifiers as in this study, based on the principle of [12], or in a multi-task learning fashion as in [13]. These paradigms avoid the combinatorial explosion reported by [16] when all possible combinations of pitches are modeled as classes.

To combine the benefits of discriminative training with explicit signal decomposition and information reduction by NMF, we use the NMF activations computed from onset and non-onset parts as positive and negative data points for the SVM classifier. In order to avoid matching of unsupervisedly estimated dictionary atoms to pitches, we employ supervised NMF where spectra corresponding to pitch-instrument pairs are pre-defined. As a result, the method is capable of low delay on-line processing, in contrast to unsupervised or weakly supervised approaches [17]. As in many previous studies [3, 12–14, 16] we limit ourselves to piano music—the main reason being that for this task, large annotated evaluation databases are available. In the ongoing, we will first describe our approach in detail before discussing parameterization, evaluation databases and metrics, and presenting experimental results.

2. METHODOLOGY

A flowchart of the proposed method is given in Figure 1. As a first step, the audio signal is converted to the time-frequency domain by either short-time Fourier transformation (STFT) or by the constant Q transformation (CQT) [18]. The time-frequency representation is mapped to frame-wise activations of note templates by means of supervised NMF. From these 'raw' activations, we derive 'high level' activation features which are then fed into a set of support vector machine (SVM) classifiers that perform onset detection for each pitch-instrument pair. The frame level decisions of these classifiers are finally post-processed by a simple clustering method. Starting from this broad picture, let us now flesh out the details of each processing step.

2.1. Calculation of the NMF Activation Matrix

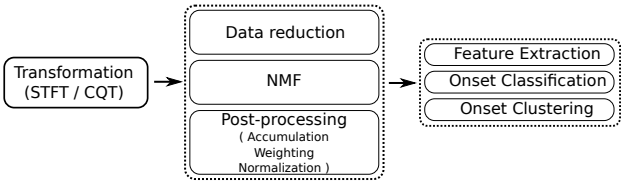
The magnitude of the time-frequency spectrogram (STFT or CQT) is computed, yielding a non-negative matrix $\mathbf{X}'$ (with observations in columns). This matrix is then 'down-sampled' by a factor N_c by merging time steps, yielding a matrix $\mathbf{X}$:

$$\mathbf{X}_t = \max\{\mathbf{X}'_{(t-1)N_c+1}, \ldots, \mathbf{X}'_{tN_c}\} \quad (1)$$

Then, NMF is applied to decompose $\mathbf{X}$ into the two factors $\mathbf{W}$ and $\mathbf{H}$; the first one represents note templates and the second one the activity of notes over time. In our *supervised* NMF approach, the matrix $\mathbf{W}$ is pre-computed in a training phase (cf. the following section); during the transcription phase, only the matrix $\mathbf{H}$ has to

The research leading to these results has received funding from the German Research Foundation (DFG) through grant no. SCHU 2508-2/1. The authors would like to thank Sebastian Böck for helpful discussions. Correspondence should be addressed to weninger@tum.de.

Fig. 1: Overview of the proposed transcription method, consisting of low-level spectral feature extraction, calculation and post-processing of NMF activation features, classification by support vector machines and decision level post-processing.



be calculated. To this end, a given cost function c is minimized iteratively by a multiplicative update algorithm. In this study, we use the Kullback-Leibler divergence $d(\cdot|\cdot)$, i. e.,

$$c(\mathbf{H}) = d(\mathbf{X}|\mathbf{WH}).$$

Starting from a random solution for $\mathbf{H}$, the update rule

$$\mathbf{H}_t \leftarrow \mathbf{H}_t \otimes \frac{\mathbf{W}^T(\mathbf{X}_t/\mathbf{WH}_t)}{\mathbf{W}^T\mathbf{1}} \tag{2}$$

is applied until the solution has converged or a maximum number of K iterations has been reached. There, $\mathbf{1}$ denote an all-one matrix of the appropriate dimensions, and the subscript t represents the t-th column of a matrix. $\otimes$ and $/$ indicate element-wise operations. It is important to note that the frame-wise rule (2) is equivalent to the standard multiplicative update algorithm [19], but makes the supervised NMF algorithm suitable for on-line processing, since the matrix $\mathbf{W}$ is constant [17].

2.2. Base Matrix Estimation

In order to build the matrix $\mathbf{W}$, we exploit NMF in a weakly supervised fashion as follows. We assume that there are recordings of isolated notes available for the instrument we want to transcribe (one per pitch). Then, 'characteristic spectra' for each of these notes are calculated from the spectrograms $\mathbf{X}_p$, where p is the pitch index. A naïve approach is to simply apply unsupervised NMF on $\mathbf{X}_p$, using a rank r of 1 and keeping the first factor (i. e., a column vector) as dictionary atom for pitch p. However, since we are mostly interested in detecting the onsets of the notes, we can also use an *onset sharpening* method to extract spectra representing the attack and the decay phases of a note, motivated by the observations made by [20] in the context of weakly supervised NMF on piano music. In our approach, we consider the activations per frame t, obtained by unsupervised NMF, as a row vector $\mathbf{a}$, and compute the maximum activation a^* and its position t^*. Then, we set the frame index t' to the first frame after the maximum ($t' > t^*$) where

$$\mathbf{a}_{t'} < \sigma \cdot a^* \tag{3}$$

We then obtain an 'onset atom' by applying unsupervised NMF only on the first t' columns of $\mathbf{X}_p$. Analogously, from the rest of $\mathbf{X}_p$ we obtain a 'decay atom'. We will evaluate the usage of onset and decay atoms later. Finally, the matrix $\mathbf{W}$ is simply the column-wise concatenation of the atom(s) estimated per pitch p, normalized to unity L2 norm per column.

2.3. Activation Post-Processing

Before performing onset detection on the NMF activations, a three-step post-processing stage is applied. First, the activations are summed up for each pitch p in case that multiple dictionary atoms per pitch are used. The outcome of this step will be denoted by $\mathbf{h}(t)$ in the ongoing, and its components by $h_p(t)$. Second, since we found that certain pitches had overall higher activations than others despite the normalized atoms in $\mathbf{W}$, we apply an 'inverse document frequency' normalization to the activation vectors per time step:

$$h_p(t) \leftarrow h_p(t)/w_p, \forall p$$

where w_p is the average activation of pitch p when NMF is applied to the training data. Finally, $\mathbf{h}(t)$ is normalized by its L1 norm adding a constant c_n:

$$\mathbf{h}(t) \leftarrow \mathbf{h}(t)/(c_n + \|\mathbf{h}(t)\|_1)$$

The constant c_n is needed because a naïve normalization would yield erroneous activations for segments without onsets. In the ongoing $c_n = 6$ will be used.

2.4. Activation Feature Extraction

In a baseline approach, we use a single activation difference feature per pitch. Precisely, defining $T_{\text{span}}(t)$ as the set of frame indices corresponding to the span in milliseconds after the frame with index t, we compute

$$f_1(h_p(t)) = h_p(t) - \max_{t' \in T_{-50}(t)} h_p(t'),$$

i. e., the difference of the current activation to the maximum activation within the last 50 milliseconds.

Besides, we consider a multi-dimensional feature set adding

$$\begin{aligned}
f_2(h_p(t)) &= \max_{t' \in T_{50}(t)} h_p(t') - \min_{t' \in T_{-50}(t)} h_p(t'),\\
f_3(h_p(t)) &= h_p(t) - \min_{t' \in T_{-100}(t)} h_p(t'),\\
f_4(h_p(t)) &= \max_{t' \in T_{100}(t)} h_p(t'),\\
f_5(h_p(t)) &= \min_{t' \in T_{-100}(t)} h_p(t'), \text{and}\\
f_6(h_p(t)) &= \max_{t' \in T_{250}(t)} h_p(t') - h_p(t).
\end{aligned}$$

2.5. Classification and Onset Detection

For each pitch p, a support vector machine (SVM) classifier is trained on a labelled set of feature vectors $\{\mathbf{f}_t\}$. For the multi-dimensional feature set, $\mathbf{f}(t) = (f_1(t), \ldots, f_6(t))^T$. In case of the single-dimensional feature set, we also use a SVM for consistency—this corresponds to a threshold decision on activation differences, where the threshold is optimized by a maximum margin criterion on the training data. To obtain a set of positive examples for the SVM, we use feature vectors inside a detection window of 100 ms around the ground truth that have a maximum acceleration of the activation, since rising activations indicate onsets and an attack phase may include several points of rising activations. Mathematically, we add $(\mathbf{f}(t^*), 1)$ to the training set with $t^* = \arg\max_t\{h_p(t) - h_p(t-1) - (h_p(t-1) - h_p(t-2))\}$. Negative examples are taken from intervals outside the detection window. To reduce redundancy caused by many similar data points representing silence, all data points yielding an activation difference less than the average are discarded with a probability close to one. Still, the above procedure yields a large training set. For example, the training set introduced in Section 3.1 corresponds to 14 h of music, resulting in 2.5 million data points. For efficiency reasons, and since usage of non-linear SVM kernels did not significantly improve

Table 1: Evaluation databases: MIDI, MAPS MIDI and MAPS D(isklavier). Total recording lengths of the partitions given in hours:minutes:seconds.

Dataset	Partition	# pieces	# notes	Length
MIDI	training	200	519 477	14:18:18
	validation	26	59 835	1:59:33
	test 1	35	71 242	2:20:03
	test 2	23	58 223	1:25:05
MAPS MIDI	training	155	334 974	9:41:18
	validation	21	48 921	1:45:23
	test 1	23	36 075	1:23:47
	test 2	11	41 018	0:54:04
MAPS D	training	36	86 010	2:23:56
	validation	6	16 487	0:43:21
	test 1	3	5 675	0:11:08
	test 2	15	46 180	1:03:17

transcription results, classifier training is done with LibLinear [21], providing an efficient method to train linear SVM.

After obtaining a classifier decision for each time step, we compute the onset timing by a 'clustering' step on the 'raw' decisions. A cluster is defined by a set of positively classified data points, such that there is no negatively classified data point between two points of the set, and the two neighboring data points around this set are classified as negative. We predict an onset at the mid-point of each cluster.

3. EXPERIMENTAL SETUP

3.1. Evaluation

We use the MIDI database introduced in [12] and the MAPS (MIDI Aligned Piano Sounds) database [16]. The MIDI database consists of MIDI files collected from the Classical Piano MIDI Page[1]. The MIDI files are converted to waveforms with a sampling rate of 44.1 kHz using the freely available Maestro Concert Grand v2 sound font[2]. The MAPS database consists of synthesized music as well as real piano recordings. The first part (MAPS MIDI) is created with different software synthesizers, configurations and virtual recording conditions. The second part (MAPS D) dataset contains music played by a Yamaha Disklavier Mark III in realistic recording conditions ('ambient' and 'close'). For a detailed description of the database we refer to [16]. We treat the Disklavier part of the MAPS database as an individual corpus, since these are the only recordings of a real piano in the data sets considered.

For NMF, an instrument-dependent $\mathbf{W}$ matrix is built using isolated notes in the training sets of the databases; in case of the MIDI database, some missing isolated notes were synthesized using the above-mentioned sound font. Onset classifiers are trained on the activation features computed from the union of the training and validation sets. Statistics of the individual data sets are shown in Table 1. We use the same partitioning into training, validation and test sets as [12, 13]. However, note that [13] restricted their evaluation on the test set to a subset for which the alignments were manually verified (testing 1); we additionally evaluate on the full test set (testing $1 \cup 2$) which may contain alignment errors.

As our main evaluation measure, we choose accuracy which was introduced by [22] for onset detection and later picked up by [13] for polyphonic transcription. Accuracy is defined as TP / (TP + FP + FN), where TP is the number of true positives, i. e., notes identified with the correct pitch within a symmetric window around the ground truth onset time, FP is the number of false positives (i. e., a note of the wrong pitch is detected), and FN is the number of false negatives (i. e., a note is missing in the transcript). This is a somewhat 'harsh' measure, as it counts substitutions, i. e., pitch errors, twice (one false negative for the correct pitch and one false positive for the incorrect pitch). Additionally, we use the standard F-measure, which is the harmonic mean of recall (TP / (TP + FN)) and precision (TP / (TP + FP)) following the notion of TP, FN and FP introduced above. Following [12], the window of correct detection is set to 100 ms (ground truth timing $\pm$ 50 ms).

As a rule of thumb for the observed ranges of accuracy, accuracy improvements of more than one percent are statistically significant at the 0.1 % level according to a one-tailed z-test [23] with the number of instances corresponding to the number of notes in the data set.

Table 2: Threshold detection (1-dimensional SVM using $f_1(t)$) vs. SVM using 6-dimensional features ($\mathbf{f}(t)$), $N_c = 2$: Accuracy and F-measure (Fm).

[%]	MIDI		MAPS MIDI		MAPS D	
	Acc.	Fm	Acc.	Fm	Acc.	Fm
1-dim.	62.4	76.8	72.5	84.0	45.1	62.2
6-dim.	**74.3**	**85.3**	**79.4**	**88.5**	**68.0**	**81.0**

3.2. Parameterization

For the STFT, a window size of 3 072 samples and a step size of 512 samples are used as in a previous study using spectral features [12]. For CQT, we use the toolbox[3] presented in [18], using 24 bins per octave over seven octaves and the default parameters for window size and step size. NMF is applied for up to $K = 200$ iterations.

4. RESULTS AND DISCUSSION

In a first step, we evaluate the usefulness of our 6-dimensional feature set $\mathbf{f}(t)$ as opposed to a simple threshold decision (1-dimensional SVM) based on the NMF pitch activation differences $f_1(t)$. The accuracies and F-measures resulting from either method are shown in Table 2. We observe that for all three of the data sets, both measures are drastically increased by the proposed 6-dimensional feature set. This especially holds for the MAPS Disklavier set of real piano recordings, where 22.9 % absolute accuracy and 18.8 % F-measure are gained. As we generally observed very similar trends for accuracy and F-measure in our evaluations, we will focus on accuracy in the ongoing. Next, we evaluate the proposed merging of frames in the spectrogram matrix by the maximum operator (cf. Eqn. (1)). From the accuracies displayed in Figure 2, it can be seen that this technique consistently improves the performance over the baseline (no merging, $N_c = 1$), and best results are achieved by setting $N_c = 4$.

Next, in Figure 3, we evaluate the influence of the spectral representation (CQT or STFT). We cannot observe any improvement by using CQT instead of STFT, even if we increase the factor N_c respecting fact that the frame step chosen for CQT is larger than for STFT. In fact, the accuracy is significantly lower when using CQT rather than STFT spectra as input for the NMF step. This is probably because the linear scaling of frequency bins in the STFT domain provides better discrimination of pitches from different octaves. In the ongoing, STFT features will be used.

[1] http://www.piano-midi.de

[2] http://www.linuxsampler.org/instruments.html

[3] Software available at http://www.elec.qmul.ac.uk/people/anssik/cqt

Fig. 2: Effect of frame merging in the $\mathbf{X}$ matrix: Accuracy for different values of N_c (1), on test sets $1 \cup 2$.

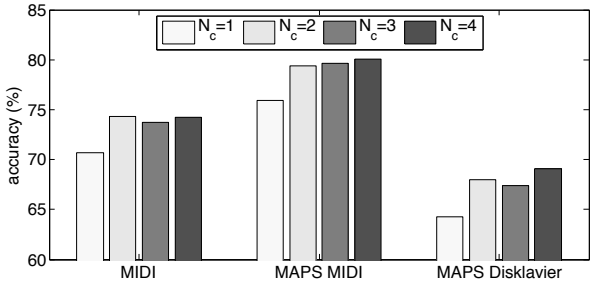


Fig. 3: Effect of spectral features: Accuracy using STFT vs. Constant-Q-Transform (CQT), for different values of N_c (1), on test sets $1 \cup 2$.

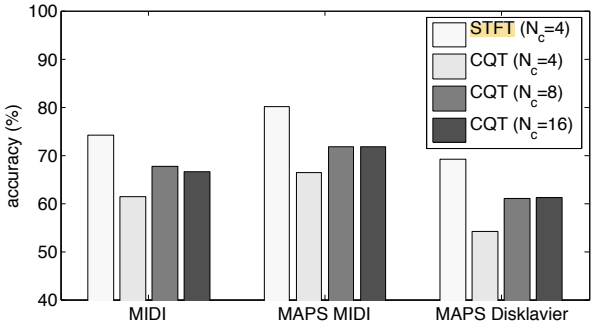
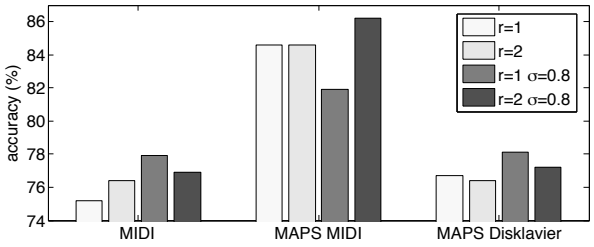


Fig. 4: Effect of dictionary size (r atoms per pitch) and onset sharpening ($\sigma = 0.8$, cf. (3)) on accuracy; evaluation on test set 1.



Finally, the method is evaluated using different dictionary sizes ($r = 1, 2$ atoms per pitch), and optionally using onset sharpening ($\sigma = 0.8$) to subdivide the training notes into attack and decay phase. For $\sigma = 0.8$ and $r = 1$, only the attack (onset) atoms are used. Results of the evaluation on the testing 1 set (for comparability with [13]) are shown in Figure 4. If we do not use onset sharpening, the number of atoms in the NMF dictionary only changes the outcome on the MIDI dataset (by 1 % absolute accuracy). Notably, the standard unsupervised NMF approach is outperformed by the proposed onset sharpening method, which delivers best results on each database. This indicates the usefulness of prior knowledge in the NMF dictionary learning process. However, on MAPS MIDI, we can only improve results over the baseline ($r = 1$, no onset sharpening) if we increase the dictionary size (and thereby computational complexity) by including the decay atoms as well. Note that the dictionary size does not influence the number of features in classification, so that the performance improvement by using $r = 2$ cannot be attributed simply to having more features.

We now compare our results (with $r = 2, \sigma = 0.8$) to the state-of-the-art in terms of accuracy, and display the results in Table 3.

Table 3: Comparison of algorithms on testing 1 set. [1]: Instrument-dependent training; [2]: Multi-instrument (closed-set) training.

Accuracy [%]	MIDI	MAPS MIDI	MAPS D
SVM[1] [12]	62.3	–	–
BLSTM[2] [13]	88.9	84.0	68.7
Boosting[1] [14]	87.4	–	–
Proposed (NMF+SVM)[1]	77.1	86.3	77.1

On the MIDI dataset, the proposed NMF+SVM method evidently delivers significantly higher accuracy (+ 14.8 % abs.) than the SVM method based on spectral features proposed by [12]. However, the method is outperformed by the boosting and BLSTM approaches proposed by [13, 14], respectively. We believe that this is due to our independent training of pitch-specific classifiers, and our method could be improved by exploiting the correlations of pitch activations, such as chord structures in tonal music. On the MAPS data set, our method is evidently superior to the results obtained by [13] (+2.3 % abs. on MIDI and +8.6 % on real piano). Yet, these results are not fully comparable since [13] uses a 'closed set' experimental setup where training data is collected from all pianos in the databases, and it is not known which piano is played in which test instance, whereas our results, similarly to the ones in [12, 14] are evaluated in an instrument-dependent setup.

We note, however, that we have good reason to believe that NMF provides a convenient and effective method to perform transcription in a closed set setup, by building a joint $\mathbf{W}$ matrix of the instrument-dependent bases, performing supervised NMF and then selecting the base with the highest overall activation for transcription. In a preliminary experiment, 83 % average recall of the ten pianos in the test databases could be achieved by deciding for the instrument whose base had the highest activation sum, respecting instrument-wise group sparsity constraints on the activations in analogy to the method proposed by [24] for speaker identification.

5. CONCLUSIONS AND OUTLOOK

We have presented an effective and efficient method for the task of polyphonic piano transcription, jointly performing multi-pitch estimation and onset detection based on NMF activation features and discriminative one-versus-all classification by SVM. The proposed method delivered state-of-the-art results on three test databases comprising synthesized MIDI as well as real piano recordings. Future work will combine the proposed feature extraction method with context modeling and multi-task learning by unidirectional LSTM networks to provide a real-time capable method using a small-sized feature set as output by NMF.

6. RELATION TO PRIOR WORK

Seminal work on NMF for polyphonic music transcription has been presented by [3, 4]. An example for non-pitched sound transcription (drums) by NMF is presented by [25]. [9, 10] introduce musically motivated constraints to unsupervised NMF for music transcription. [12, 14] perform discriminative transcription by classifying simple spectral features. [15] proposes unsupervised feature generation for classification-based transcription but does not use source separation based features as by NMF. [13] uses context information and multi-task learning by a neural network using spectral features, achieving overall best results so far on the piano transcription task.

7. REFERENCES

[1] P. Grosche, B. Schuller, M. Müller, and G. Rigoll, "Automatic Transcription of Recorded Music," *Acta Acustica united with Acustica*, vol. 98, no. 2, pp. 199–215(17), 2012.

[2] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri, "Automatic music transcription: Breaking the glass ceiling," in *Proc. of ISMIR*, Porto, Portugal, 2012, pp. 379–384, ISMIR.

[3] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2003, pp. 177–180.

[4] S. A. Abdallah and M. D. Plumbley, "Polyphonic transcription by non-negative sparse coding of power spectra," in *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR)*, C. L. Buyoli and R. Loureiro, Eds., Barcelona, Spain, 2004, pp. 318–325, Audiovisual Institute Pompeu Fabra University.

[5] E. Vincent, N. Bertin, and R. Badeau, "Two nonnegative matrix factorization methods for polyphonic pitch transcription," in *Proc. Music Information Retrieval Evaluation eXchange (MIREX)*, Vienna, Austria, 2007, pp. 23–30.

[6] S. H. Park, S. Lee, and K.-M. Sung, "Automatic music transcription using non-negative matrix factorization," in *Proceedings of 20th International Congress on Acoustics, ICA*, 2010.

[7] G. Costantini, M. Todisco, and G. Saggio, "Automatic music transcription based on non-negative matrix factorization," in *Proceedings of the 14th WSEAS international conference on Systems*. 2010, ICS'10, pp. 288–291, ACM.

[8] S. A. Abdallah and M. D. Plumbley, "Unsupervised analysis of polyphonic music by sparse coding," *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 179–196, 2006.

[9] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[10] N. Bertin, R. Badeau, and E. Vincent, "Enforcing harmonicity and smoothness in bayesian non-negative matrix factorization applied to polyphonic music transcription," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 3, pp. 538–549, 2010.

[11] M. Marolt, "A connectionist approach to automatic transcription of polyphonic piano music," *IEEE Transactions on Multimedia*, vol. 6, no. 3, pp. 439–449, 2004.

[12] G. E. Poliner and D. P. W. Ellis, "A discriminative model for polyphonic piano transcription," in *EURASIP Journal on Advances in Signal Processing*, 2007.

[13] S. Böck and M. Schedl, "Polyphonic piano note transcription with recurrent neural networks," in *Proc. of ICASSP*, 2012, pp. 121–124.

[14] C. G. v. d. Boogaart and R. Lienhart, "Note onset detection for the transcription of polyphonic piano music," in *Proceedings of the 2009 IEEE international conference on Multimedia and Expo*, 2009, ICME'09, pp. 446–449.

[15] J. Nam, J. Ngiam, H. Lee, and M. Slaney, "A classification-based polyphonic piano transcription approach using learned feature representations," in *Proc. of ISMIR*, Miami, FL, USA, 2011, pp. 175–180, ISMIR.

[16] V. Emiya, R. Badeau, and B. David, "Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 6, pp. 1643–1654, 2010.

[17] C. Joder, F. Weninger, F. Eyben, D. Virette, and B. Schuller, "Real-time speech separation by semi-supervised nonnegative matrix factorization," in *Proc. International Conference on Latent Variable Analysis and Signal Separation (LVA ICA), Special Session "Real-world constraints and opportunities in audio source separation"*, Tel Aviv, Israel, 2012, pp. 322–329.

[18] C. Schörkhuber and A. Klapuri, "Constant-Q transform toolbox for music processing," in *Proceedings of the 7th Sound and Music Conference (SMC-2010)*, 2010.

[19] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. of NIPS*, Vancouver, Canada, 2001, pp. 556–562.

[20] S. Ewert and M. Müller, "Using score-informed constraints for NMF-based source separation," in *Proc. of ICASSP*, Kyoto, Japan, 2012, pp. 129–132.

[21] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[22] S. Dixon, "On the computer recognition of solo piano music," in *Proceedings of Australasian Computer Music Conference*, 2000, pp. 31–37.

[23] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural Computation*, vol. 10, pp. 1895–1923, 1998.

[24] A. Hurmalainen, R. Saeidi, and T. Virtanen, "Group sparsity for speaker identity discrimination in factorisation-based speech recognition," in *Proc. of Interspeech*, Portland, OR, USA, 2012, no pagination.

[25] J. Paulus and T. Virtanen, "Drum transcription with nonnegative spectrogram factorisation," in *Proc. of EUSIPCO*. 2005, pp. 4–8, EURASIP.