MULTIPITCH ANALYSIS WITH HARMONIC NONNEGATIVE MATRIX APPROXIMATION

Stanisław A. Raczyński Nobutaka Ono Shigeki Sagayama
The University of Tokyo
Graduate School of Information Science and Engineering
E-mail: {raczynski,onono,sagayama}@hil.t.u-tokyo.ac.jp

ABSTRACT

This paper presents a new approach to multipitch analysis by utilizing the Harmonic Nonnegative Matrix Approximation, a harmonically-constrained and penalized version of the Nonnegative Matrix Approximation (NNMA) method. It also includes a description of a note onset, offset and amplitude retrieval procedure based on that technique. Compared with the previous NNMA approaches, specific initialization of the basis matrix is employed – the basis matrix is initialized with zeros everywhere but at positions corresponding to harmonic frequencies of consequent notes of the equal temperament scale. This results in the basis containing nothing but harmonically structured vectors, even after the learning process, and the activity matrix's rows containing peaks corresponding to note onset times and amplitudes. Furthermore, additional penalties of mutual uncorrelation and sparseness of rows are placed upon the activity matrix. The proposed method is able to uncover the underlying musical structure better than the previous NNMA approaches and makes the note detection process very straightforward.

1 INTRODUCTION

The problem of automatic polyphonic music transcription (extracting underlying musical structure from sampled music) has been addressed numerous times, and it still seems there is a long way to go before arriving at a robust and universal technique. This paper tries to lay another brick towards this goal.

Automatic music transcription of recorded music is usually a two stage process. The first stage is the event detection phase, where music events (note onsets, note offsets, pitch changes) are detected and identified. In the second stage, these events are transformed into a musical score. This paper focuses on the event detection stage, main part being multipitch analysis, which aims to uncover the fundamental frequencies of simultaneously played harmonic sounds. It is a difficult task, since each sound, besides the fundamental tone, consists of many harmonic tones, some of them having the same frequencies as the fundamental frequencies of other sounds (e.g. in the case of tonal music). It is necessary to distinguish between the fundamental tones and their overtones.

A large variety of methods has been used to tackle the multipitch analysis problem (e.g. [1], [2], [4], [9], [11], [12], [13]; an exhaustive list of methods would be very long and we are not going to include it here, but for a good summary, see [6]), but so far none of them solving the problem in a satisfactorily precise and universal way. While our lab has recently developed a powerful method based on Harmonic Temporal Structured Clustering (HTC) for this purpose [4], the procedure proposed in this paper is built upon a method from the family of Nonnegative Matrix Approximations (NNMA), which, under different names and in different varieties, has recently received much attention, also from the music transcription community. To the best of our knowledge, however, none of the NNMA-based methods were developed specifically for analysis of musical signals. As it will be shown later in this paper, nature of music can be exploited to increase the transcription potential of the algorithm. The goal of this paper was to propose a NNMA variation most suitable for multipitch analysis.

Different variations and extensions of the NNMA algorithm have been used for multipitch analysis: the regular NNMA [13], its penalized versions, such as the Nonnegative Sparse Coding (NNSC) [2, 1], or NNMA with basis vectors extended to contain spectrotemporal signatures (a number of consequent data frames), such as Nonnegative Matrix Factor 2-D Deconvolution (NMF2D) and Sparse Nonnegative Matrix Factor 2-D Deconvolution (SNMF2D) [11]. These methods have, however, a few drawbacks. They do not guarantee to yield basis vectors with harmonic structure. NMF2D and SNMF2D use a single signature for every note (of a single instrument), making use of the shift-similarity of logarithmic frequency scale spectra of notes played on a single instrument. This might be an oversimplification resulting in an inadequate model. The note spectra are similar, but not identical, with significant differences for some specific instruments (e.g. flute). Moreover, spectrotemporal atoms cannot account for different note lengths, which results in multiple activity peaks when notes are longer than the signature, and lower activity peaks when notes are shorter than the signature. All of the previous work published on that subject do not propose a complete transcription procedure, simply reporting

good results after visual comparison of the activities and symbolic data used to generate the analyzed music.

The paper is organized as follows. Section 2 presents a theoretical introduction to the Nonnegative Matrix Approximation and its extension through the addition of constraints and penalties placed upon both the basis matrix and the activity matrix. An overview of the proposed procedure in given in section 3, including description of the proposed Harmonic Nonnegative Matrix Approximation (HNNMA) technique (3.3) and note detection method (3.4). The procedure is evaluated and compared with the results of regular NNMA methods in section 4.

2 THEORETICAL BACKGROUND

2.1 Definitions and basic properties

For clarity, the following notation was used in this paper: $\left|\cdot\right|$ is a sum of all the elements of a matrix, $\odot$ is the Hadamard product (calculated element-wise) and $\mathbf{1}$ is a matrix (of appropriate dimensions) containing nothing but ones.

A few easy to prove properties were later used. If $\mathbf{A} \in \mathbb{R}^{N\times M}$, then:

$$\nabla_{\mathbf{A}}\left|\mathbf{A}\odot\mathbf{A}\right| = 2\mathbf{A}, \tag{1}$$

$$\nabla_{\mathbf{A}}\left|\mathbf{A}^T\mathbf{A}\right| = 2\mathbf{A}^T\mathbf{1}, \tag{2}$$

$$\nabla_{\mathbf{A}}\left|\mathbf{B}\mathbf{A}\right| = \mathbf{B}^T\mathbf{1}, \tag{3}$$

$$\nabla_{\mathbf{A}}\left|f_1(\mathbf{A})+f_2(\mathbf{A})\right| = \nabla_{\mathbf{A}}\left|f_1(\mathbf{A})\right| + \nabla_{\mathbf{A}}\left|f_2(\mathbf{A})\right|, \tag{4}$$

$$\nabla_{\mathbf{A}}\left|\mathbf{B}\odot(\mathbf{C}\upsilon(\mathbf{A}))\right| = \upsilon'(\mathbf{A})\odot\left(\mathbf{C}^T\mathbf{B}\right), \tag{5}$$

where $\upsilon : \mathbb{R}^+ \to \mathbb{R}^+$ is an element-wise function and υ' is its derivative, and $f_1, f_2 : \mathbb{R}^{+,N\times M} \to \mathbb{R}^{+,N\times M}$ are any matrix functions.

2.2 Generalized Nonnegative Matrix Approximation

Generalized Nonnegative Matrix Approximation (described in [3] and later developed in [14]), is a method for decomposition of a nonnegative (having only nonnegative elements) matrix $\mathbf{X}$ (later referred to as the data matrix) into a multiplication of two, also nonnegative, matrices $\mathbf{A}$ and $\mathbf{S}$ (later refereed to as the basis matrix and the activity matrix, respectively):

$$\mathbf{X} \cong \mathbf{AS} = \widetilde{\mathbf{X}}. \tag{6}$$

The Generalized NNMA solves this problem by minimizing a Bregman divergence between the data matrix $\mathbf{X}$ and its approximation $\widetilde{\mathbf{X}}$. A Bregman divergence between two matrices is defined as

$$D(\mathbf{P},\mathbf{Q}) = \left|\varphi(\mathbf{P}) - \varphi(\mathbf{Q}) - \varphi'(\mathbf{Q})\odot(\mathbf{P}-\mathbf{Q})\right|, \tag{7}$$

where $\varphi : S \subseteq \mathbb{R} \to \mathbb{R}$ is a strictly convex function with continuous first derivative, calculated here for each element of a matrix separately. If $\varphi(p) = p\log p - p$, then the Bregman divergence becomes the I-divergence (generalized Kullback-Leibler divergence):

$$D_{KL}(\mathbf{P},\mathbf{Q}) = \left|\mathbf{P}\odot\log\frac{\mathbf{P}}{\mathbf{Q}} - \mathbf{P} + \mathbf{Q}\right|, \tag{8}$$

where the logarithm and the division are calculated element-wise. This situation leads to the simple NNMA, also known as Nonnegative Matrix Factorization (NMF) [8]. Lee and Seung in [8] has proposed a very fast algorithm for minimizing the I-divergence that can be derived by using auxiliary functions. A function $G(\mathbf{P},\mathbf{P}')$ is an auxiliary function for function $F(\mathbf{P})$ if:

1. $G(\mathbf{P},\mathbf{P}) = F(\mathbf{P})$,
2. $G(\mathbf{P},\mathbf{P}') \geq F(\mathbf{P})$.

Making use of the convexity of φ [14], it can be shown that:

$$G(\mathbf{A},\mathbf{A}') = \left|\varphi(\mathbf{X}) + \tilde{\mathbf{X}} - \mathbf{X} - \frac{\mathbf{X}}{\tilde{\mathbf{X}}'}\odot\left[\left(\mathbf{A}'\odot\log\frac{\mathbf{A}}{\mathbf{A}'}\right)\mathbf{S} + \varphi(\tilde{\mathbf{X}}')\right]\right|, \tag{9}$$

where $\widetilde{\mathbf{X}}' = \mathbf{A}'\mathbf{S}$, is an auxiliary function for

$$\begin{aligned} F(\mathbf{A}) &= D_{KL}(\mathbf{X},\mathbf{AS}) = D_{KL}(\mathbf{X},\widetilde{\mathbf{X}}) \\ &= \left|\mathbf{X}\odot\log\frac{\mathbf{X}}{\tilde{\mathbf{X}}} - \mathbf{X} + \tilde{\mathbf{X}}\right| \end{aligned} \tag{10}$$

and

$$G(\mathbf{S},\mathbf{S}') = \left|\varphi(\mathbf{X}) + \widetilde{\mathbf{X}} - \mathbf{X} - \frac{\mathbf{X}}{\widetilde{\mathbf{X}}'}\odot\left[\mathbf{A}\left(\mathbf{S}'\odot\log\frac{\mathbf{S}}{\mathbf{S}'}\right) + \varphi(\widetilde{\mathbf{X}}')\right]\right|, \tag{11}$$

where this time $\widetilde{\mathbf{X}}' = \mathbf{AS}'$, is an auxiliary function for

$$F(\mathbf{S}) = D_{KL}(\mathbf{X},\mathbf{AS}). \tag{12}$$

It can also be shown [14] that $F(\mathbf{S}')$ is non-increasing under the update

$$\mathbf{S}' \leftarrow \arg\min_{\mathbf{S}} G(\mathbf{S},\mathbf{S}'). \tag{13}$$

To solve this optimization problem, we calculate the gradient of the auxiliary function and force it to zero:

$$\nabla_{\mathbf{S}}G(\mathbf{S},\mathbf{S}') = \nabla_{\mathbf{S}}\left|\varphi(\mathbf{X}) + \widetilde{\mathbf{X}} - \mathbf{X} - \frac{\mathbf{X}}{\widetilde{\mathbf{X}}'}\odot\left[\mathbf{A}\left(\mathbf{S}'\odot\log\frac{\mathbf{S}}{\mathbf{S}'}\right) + \varphi(\widetilde{\mathbf{X}}')\right]\right| = \mathbf{0}. \tag{14}$$

Using properties (3), (4) and (5), we can easily calculate that gradient as:

$$\nabla_{\mathbf{S}}\left|\mathbf{AS} - \frac{\mathbf{X}}{\widetilde{\mathbf{X}}'}\odot\left[\mathbf{A}\left(\mathbf{S}'\odot\log\frac{\mathbf{S}}{\mathbf{S}'}\right)\right]\right| = \mathbf{0}, \tag{15}$$

$$\mathbf{A}^T\mathbf{1} - \frac{\mathbf{S}'}{\mathbf{S}} \odot \left(\mathbf{A}^T\left(\frac{\mathbf{X}}{\widetilde{\mathbf{X}}'}\right)\right) = \mathbf{0}, \tag{16}$$

$$\mathbf{S} = \mathbf{S}' \odot \frac{\mathbf{A}^T\left(\frac{\mathbf{X}}{\widetilde{\mathbf{X}}'}\right)}{\mathbf{A}^T\mathbf{1}}. \tag{17}$$

This suggest a multiplicative update rule:

$$\mathbf{S} \leftarrow \mathbf{S} \odot \frac{\mathbf{A}^T\left(\frac{\mathbf{X}}{\mathbf{AS}}\right)}{\mathbf{A}^T\mathbf{1}}. \tag{18}$$

We can come up with a similar update rule for the basis matrix $\mathbf{A}$:

$$\nabla_{\mathbf{A}} G(\mathbf{A}, \mathbf{A}') = \mathbf{0}, \tag{19}$$

$$\mathbf{1}\mathbf{S}^T - \frac{\mathbf{A}'}{\mathbf{A}} \odot \left(\frac{\mathbf{X}}{\widetilde{\mathbf{X}}'}\mathbf{S}^T\right) = \mathbf{0}, \tag{20}$$

$$\mathbf{A} \leftarrow \mathbf{A} \odot \frac{\frac{\mathbf{X}}{\mathbf{AS}}\mathbf{S}^T}{\mathbf{1}\mathbf{S}^T}. \tag{21}$$

So, by using simple multiplicative rules from (21) and (18), we can find matrices $\mathbf{A}$ and $\mathbf{S}$ that minimize the I-divergence from equation (8).

2.3 Penalized NNMA

Because $H(\mathbf{P}, \mathbf{P}') = G(\mathbf{P}, \mathbf{P}') + \alpha(\mathbf{P})$ is an auxiliary function for $F(\mathbf{P}) + \alpha(\mathbf{P})$ (the proof is straightforward), the NNMA algorithm can be extended by placing additional penalties upon both estimated decomposition matrices, expressed as additional element in the objective function.

$$\nabla_{\mathbf{A}} H(\mathbf{A}, \mathbf{A}') = \nabla_{\mathbf{A}} G(\mathbf{A}, \mathbf{A}') + \nabla_{\mathbf{A}} \alpha(\mathbf{A}), \tag{22}$$

$$\nabla_{\mathbf{A}} H(\mathbf{A}, \mathbf{A}') = \mathbf{1}\mathbf{S}^T - \frac{\mathbf{A}'}{\mathbf{A}} \odot \left(\frac{\mathbf{X}}{\mathbf{X}'}\mathbf{S}^T\right) + \nabla_{\mathbf{A}} \alpha(\mathbf{A}). \tag{23}$$

While it is very difficult to solve this non-linear equation with respect to $\mathbf{A}$, the following approximation can be used [14]:

$$\nabla_{\mathbf{A}} \alpha(\mathbf{A}) \cong \nabla_{\mathbf{A}} \alpha(\mathbf{A})\Big|_{\mathbf{A}=\mathbf{A}'}, \tag{24}$$

which is asymptotically true, as difference between $\mathbf{A}$ in consequent iterations tends to $\mathbf{0}$. Now:

$$\mathbf{1}\mathbf{S}^T - \frac{\mathbf{A}'}{\mathbf{A}} \odot \left(\frac{\mathbf{X}}{\mathbf{X}'}\mathbf{S}^T\right) + \nabla_{\mathbf{A}'} \alpha(\mathbf{A}') = \mathbf{0}, \tag{25}$$

which yields an update rule:

$$\mathbf{A} \leftarrow \mathbf{A} \odot \frac{\frac{\mathbf{X}}{\mathbf{AS}}\mathbf{S}^T}{\mathbf{1}\mathbf{S}^T + \nabla_{\mathbf{A}} \alpha(\mathbf{A})}. \tag{26}$$

Similarly:

$$\nabla_{\mathbf{S}} H_S(\mathbf{S}, \mathbf{S}') \cong \nabla_{\mathbf{S}} G(\mathbf{S}, \mathbf{S}') + \nabla_{\mathbf{S}'} \beta(\mathbf{S}'), \tag{27}$$

$$\mathbf{S} \leftarrow \mathbf{S} \odot \frac{\mathbf{A}^T\frac{\mathbf{X}}{\mathbf{AS}}}{\mathbf{A}^T\mathbf{1} + \nabla_{\mathbf{S}} \beta(\mathbf{S})}. \tag{28}$$

It must be noted that the new update rules may result in the matrices $\mathbf{A}$ and $\mathbf{S}$ becoming negative, so caution must be taken while constructing the objective function.

3 MULTIPITCH ANALYSIS PROCEDURE

3.1 Overview of the procedure

The NNMA algorithm decomposes the data matrix $\mathbf{X}$, which does not contain musical data, but rather some mid-level representation of it. In most cases its columns are power spectra of consecutive frames of time-domain musical data. In the proposed procedure a constant-Q transform is used. The central frequencies of the constant-Q filters can be set to correspond to the frequencies of the notes of the most common twelve-tone equal temperament (12-TET) scale or can further divide each semitone, which is a very useful property for analyzing musical signals. What is more, the dimensionality of a constant-Q-transformed data is much lower than the dimensionality of a Fourier-transformed data, which makes the computation of the NNMA faster. After calculating the constant-Q transform, the resulting data is fed through the HNNMA algorithm, which decomposes it to a product of the basis matrix and activity matrix. Activity matrix is analyzed in the last part of the procedure – the note detector, described in section 3.4.

3.2 Matrix initialization in HNNMA

Because zero-valued elements of basis vectors will remain zero-valued throughout the learning process, we could initialize them to have zeros everywhere but at the positions of fundamentals of notes from a specific range of the 12-TET scale and their harmonics. Furthermore, that would guarantee that the basis vectors are sorted by their fundamental frequencies, and that corresponding rows in the activity matrix contain activities of consequent notes from that range, resulting in a harmonicaly-constrained NNMA. This would make analysis of the results of the algorithm straightforward – one would only have to analyze the note activities and find peaks corresponding to instances of these notes.

In the proposed procedure, after initialization, each basis vector is multiplied by the normalized mean value of the constant-Q transform of the data at the bin corresponding to this note. This should discourage the HNNMA from learning these notes and using them to reconstruct the analyzed data. During the learning process, each row of the activity matrix is, as it is usually done in the learning process of the NNMA methods, normalized to unit squared sum, while the basis matrix is simply normalized by its maximal value to let the basis vectors, that correspond to notes not existing in analyzed music, freely decrease.

3.3 Additional penalties in HNNMA

HNNMA extends the regular NNMA to include additional penalties on the activity matrix $\mathbf{S}$. We would like to find such an activity matrix that would:

1. be sparse, i.e. each row should contain only very few non-zero elements (to reduce the low-valued noise),

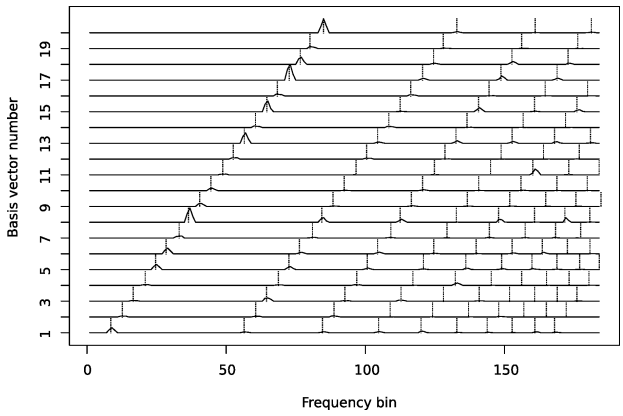


Figure 1. Basis vectors after analysis of the *Ode to Joy*. Its harmonic structure is clearly visible. Vertical dotted lines indicate expected harmonic peaks positions.

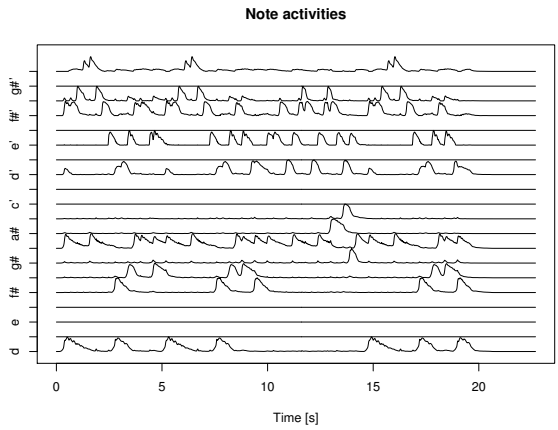


Figure 2. Note activities ($\mathbf{S}$) after analysis of the *Ode to Joy*. Peaks correspond to notes detected in the signal

2. contain mutually uncorrelated rows (to reduce the inter-row crosstalk, like e.g. octave errors).

The above can be reformulated, accordingly, in terms of a objective function β:

$$\begin{aligned}\beta(\mathbf{S}) &= -\mu_1\left|\log(1+\mathbf{S}\odot\mathbf{S})\right| \\ &\quad +\mu_2\left(\left|\mathbf{S}^T\mathbf{S}\right|-\left|\mathbf{S}\odot\mathbf{S}\right|\right). \end{aligned} \quad (29)$$

The first element, $\left|\log\left(1+\mathbf{S}\odot\mathbf{S}\right)\right|$, is one of the often used sparseness measures [5]. The second one is a measure of correlation between every pair of different matrix rows:

$$\left|\mathbf{S}^T\mathbf{S}\right|-\left|\mathbf{S}\odot\mathbf{S}\right| = \sum_i\sum_{j\neq i}\mathbf{s}_i^T\mathbf{s}_j, \quad (30)$$

where $\mathbf{s}_i^T$ is the i-th row and $\mathbf{s}_i$ is its transposition. Using properties (1), (2) and (5) from section 2.1, we can easily calculate the gradient:

$$\nabla_{\mathbf{S}}\beta(\mathbf{S}) = -2\mu_1\mathbf{S}/\left(1+\mathbf{S}\odot\mathbf{S}\right)+2\mu_2\mathbf{S}\left(\mathbf{1}-\mathbf{I}\right). \quad (31)$$

Similar penalties could be used for the basis matrix, but our experiments with sparsity, column uncorrelation and column shift-similarity showed that these constraints do not improve procedure's accuracy when the basis matrix was initialized in the way described in the next subsection. When the matrix was initialized with traditional noise, the constraints would result in basis vectors containing peaks, although the structure was not always purely harmonic. Thus, either further, much more complex constraint are required to enforce this structure, or we could take the advantage of the multiplicative nature of HNNMA algorithm update rules.

3.4 Note detector

Before analysis, each row of the activity matrix is multiplied by the height of the peak at the fundamental frequency in corresponding basis vector (all rows are being normalized to unit squared sum during the learning process). This should make the activities of notes that do not exist in the analyzed music significantly smaller than the activities of notes that occur in the music. After that each row with values higher than some arbitrarily chosen threshold are normalized to the maximal value in each of them.

It turns out that in preliminary experiments the resulting activities clearly correspond to notes in the analyzed musical piece. However, if the notes were played shortly one after another, their peaks blend to form a single peak with multiple sub-peaks. Because of that, a simple thresholding is not enough. A still simple, but much more robust thresholding method was used. First, the activities are thresholded to detect peaks and blended peaks (e.g. two blended peaks depicted on Figure 3). Then, for each detection all local maxima and local minima are found. Some of the maxima correspond to actual sub-peaks, while some are just fluctuations in the note activity. Two thresholds are set between the highest local maximum and the lowest local minimum. All maxima that are above the higher threshold (upper ligth-gray range on Figure 3) and has at least one minimum lower than the lower threshold (lower light-gray range on picture 3) are marked as sub-peaks and are assumed to correspond to individual notes. In similar fashion, all minima under the lower threshold that lay between two sub-peaks are assumed to be the offset time of the note corresponding to the left sub-peak and onset time of the note corresponding to the right sub-peak. The beginning and the end of the blended group of sub-peaks are assumed to be the onset time of the first note in the group and the offset time of the last note in the group.

The last step of the note detection process is acceptance decision for each of the detected peaks and sub-peaks. A peak is accepted and regarded a note only if its width multiplied by its height is greater than some threshold.

4 EXPERIMENTAL RESULTS

4.1 Experiment conditions

To validate our approach, we tested our procedure on a few recordings. All analyzed recordings were played on

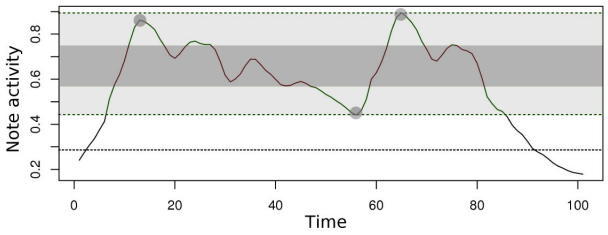


Figure 3. Results of note detection for a quarter note and an eighth blended together (example of real data). Dark-gray circles mark the detected sub-peaks and the offset time of the first note.

Composer	Title	Notes	Acc.	Corr.
L. Beethoven	Symphony in D minor, Op. 125, No. 9 (last movement, *Ode to joy*)	101	96%	86%
F. Chopin	Nocturne in E# major, Op. 9, No. 2 (part)	328	87%	72%
F. Chopin	Nocturne in Bb minor, Op. 9, No. 1 (part)	358	70%	74%
J. S. Bach	Minuet No 4 in G	102	97%	100%

Table 1. Piano pieces used for algorithm evaluation

piano, as listed in Table 1, but experiments show equally good results for acoustic guitar and violin (though lower detection accuracy for violin).

During the experiments, all the parameters of the algorithm were kept constant in order to evaluate its robustness, though by fine-tuning the parameters for each musical piece separately, much better results can be obtained. It has been noted that the best results were achieved by applying this method for shorter blocks of data (30-60 s), instead of the whole song at once.

The input data was first mixed down to a monaural signal and resampled to 11025 kHz. The constant-Q transform was calculated for frames shifted 12 ms. During learning, $\mu_1 = 1$, $\mu_2 = 10$ were used. During note detection phase, before normalization of activities, rows having maximal values lower than 0.125 of the maximal value of the activity matrix $\mathbf{S}$ were set to zero. The main detection threshold was set at 0.25, the lower threshold to 0.25 of difference between the lowest minimum and the highest maximum and the higher threshold to 0.75 of that difference.

After learning the basis matrix contained very well structured vectors, each one having a stronger peak for the fundamental tone and weaker peaks for the harmonics (Figure

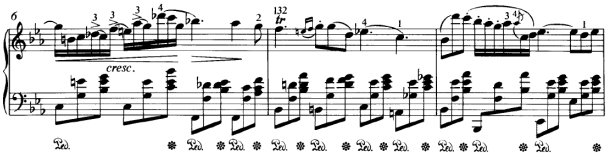
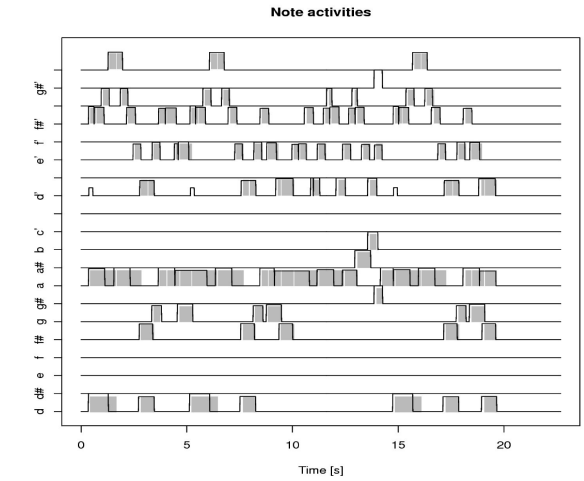
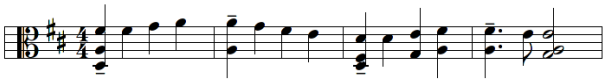


Figure 4. Three bars from the middle of Chopin's Nocturne in E# major, Op. 9, No. 2



(a) Note activities obtained by NNMA



(b) The first 4 bars of the played score, as a reference

Figure 5. Note activities after note detection for *Ode to Joy* with grey squares being the original notes

1). The results of note detection for few example pieces of music are presented in Table 1. Correctness of transcription is the ratio of the difference between the number of notes in analyzed music and the number of deletions, to the number of notes in analyzed music. Accuracy is the ratio of the difference between the number of detected notes and the number of insertions, to the number of detected notes. The first and the last musical piece were relatively easy to analyze, containing notes from a rather short range (about 2 octaves). The two Chopin's nocturnes were, on the other hand, very difficult – played with a big dynamic and wide range of note lengths, and containing notes from within 5-6 octaves (see e.g. Figure 4).

4.2 Comparison with previous methods

Figure 6 depicts the basis matrix obtained using standard NNMA (NMF) method after fundamental frequency estimation and basis vector sorting. It does not contain clear harmonic structure – many of the vectors have two (or more) dominant peaks, sometimes with highest peaks being the overtones instead of the fundamental, sometimes having the same fundamental frequency as different basis vectors (Figure 6). Slightly better results are obtained by utilizing different penalized NNMA methods proposed in the literature (e.g NNSC [1] or Local Nonnegative Matrix Factorization [10]), however, the basis matrix never contains as highly harmonically structured vectors as the ones obtained with the proposed method. The activities obtained with these methods contain a lot of fluctuations and assigning note names to the activities depends on the highly dubious operation of fundamental frequency estimation of the basis vectors.

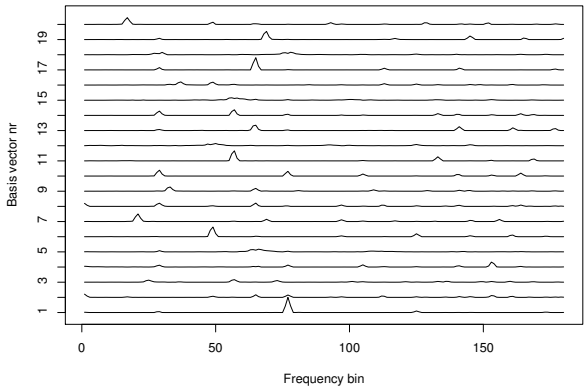


Figure 6. Basis obtained with standard NMF. Note detection relies on pitch estimation of these, often multi-peaked, basis vectors.

It seems that the activity matrix in the proposed procedure contains very easy to analyze and at the same time almost complete information about the underlying musical structure of the analyzed signal. This method is a good compromise between full basis estimation methods (such as NMF and other NNMA-based approaches) and methods that use pre-learned basis vectors (e.g. [12] or [9]). The achieved results are similar to the results of different recently developed music transcription techniques (e.g. [4]), but by fine-tuning the method's parameters, even greater accuracy could be achieved. The proposed procedure uses a relatively simple method of analyzing the activity matrix, making room for future research in more advanced techniques, such as modeling the temporal envelopes of notes or using models of musical rhythm and harmony.

5 CONCLUSION

In this paper, we discussed the use of Harmonic Nonnegative Matrix Approximation for multipitch analysis of polyphonic music signals. By initializing the basis matrix with harmonic structure and using new penalties of sparsity and uncorrelation of rows of the activity matrix, this approach yielded higher note detection accuracy compared with previous extensions of the Nonnegative Matrix Approximation algorithm. The future work includes improving the post-processing of the HNNMA results by incorporating models of musical rhythm and harmonicity.

6 REFERENCES

[1] Abdallah, S.A. and Plumbley, M.D. "Polyphonic music transcription by non-negative sparse coding of power spectra," *Proc. 5th International Conference on Music Information Retrieval*, pp. 318–325, Barcelona, Spain, 2004.

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

[3] Dhillon, I.S. and Sra, S. "Generalized Nonnegative Matrix Approximations with Bregman Divergences," *Proc. Neural Information Processing Systems,* Vancouver, USA 2005.

[4] Kameoka, H., Nishimoto, T., Sagayama, S. "Multi-pitch Analyzer Based on Harmonic Temporal Structured Clustering," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 15, no. 3, pp. 982–994, Mar, 2007.

[5] Karvanen, J. and Cichocki, A. "Measuring sparseness of noisy signals," *Proc. 4th International Symposium on Independent Component Analysis and Blind Signal Separation*, 2003.

[6] Klapuri, A.P. "Automatic Music Transcription as We Know it Today," *Journal of New Music Research*, vol. 33, no. 3, pp. 269–282, 2004.

[7] Lee, D.D. and Seung, H.S. "Algorithms for Non-negative Matrix Factorization," *Advances in Neural Information Processing Systems*, vol. 13, pp. 556–562, 2001.

[8] Lee, D.D. and Seung, H.S. "Learning the parts of objects by nonnegative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[9] Lepain, P. "Polyphonic Pitch Extraction from Musical Signals," *Journal of New Music Research*, vol. 28, no. 4, pp. 296–309, 1999.

[10] Li, S.Z. and Hou, X.W. and Zhang, H.J. and Cheng, Q.S. "Learning spatially localized, parts-based representation," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1–6, 2001.

[11] Schmidt, M.N. and Mørup M. "Sparse Non-negative Matrix Factor 2-D Deconvolution for Automatic Transcription of Polyphonic Music," *Proc. 6th International Symposium on Independent Component Analysis and Blind Signal Separation*, Charleston, USA, 2006.

[12] Sha, F. and Saul, L.K. "Real-Time Pitch Determination of One or More Voices by Nonnegative Matrix Factorization," *Advances in Neural Information Processing Systems*, vol. 17, 2005.

[13] Smaragdis, P. and Brown. J.C. "Non-Negative Matrix Factorization for Polyphonic Music Transcription," *Proc. 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New York, USA, 2003.

[14] Sra, S. and Dhillon, I.S. "Nonnegative Matrix Approximations: Algorithms and Application," *Technical Report Tr-06-27,* Computer Sciences, University of Texas, Austin, USA 2006.