NON-NEGATIVE MATRIX DIVISION FOR THE AUTOMATIC TRANSCRIPTION OF POLYPHONIC MUSIC

Bernhard Niedermayer
Department of Computational Perception
Johannes Kepler University Linz, Austria

ABSTRACT

In this paper we present a new method in the style of non-negative matrix factorization for automatic transcription of polyphonic music played by a single instrument (e.g., a piano). We suggest using a fixed repository of base vectors corresponding to tone models of single pitches played on a certain instrument. This assumption turns the blind factorization into a kind of non-negative matrix division for which an algorithm is presented. The same algorithm can be applied for learning the model dictionary from sample tones as well. This method is biased towards the instrument used during the training phase. But this is admissible in applications like performance analysis of solo music. The proposed approach is tested on a Mozart sonata where a symbolic representation is available as well as the recording on a computer controlled grand piano.

1 INTRODUCTION

Transcription of polyphonic music is a difficult task even for humans after several years of musical training. In the computational field people have been working on the problem of extracting single note events from complex music recordings for more than three decades. In special cases like monophonic music some systems have proven to be successful [1]. But pieces where more than one note is present at a time are much more challenging. Consonant tones in Western music often have frequency relations close to simple integer ratios and therefore cause overlapping harmonics. So when considering a power spectrum the mapping of found energies to certain fundamental frequencies is usually ambiguous.

A review of transcription methods is given in [4] and [5], clustering them into three main approaches. The first systems were built on pure bottom-up principles without considering any higher level knowledge. Although these algorithms used to fit very specific cases only, recent works like [6] or [13] show that bottom-up methods have overcome those early restrictions. A second group of transcription methods, like used by [12], is based on blackboard systems. Here low-level information gathered by digital signal processing and frame-wise description of the auditory scene as well as high-level prior knowledge is used to support or discard hypotheses at multiple levels.

The third major approach to music transcription is made up of model based algorithms. Similar to blackboard systems they also include high-level information as well as low-level signal based features. The difference is that prior knowledge is fed into the system by introducing a model of the analyzed data. The signal is then processed in order to estimate the model's parameters. The results of these methods can only be as good as the assumed model fits the actual data. Works like [14] or [2] are examples of this class.

During the last years the methods of non-negative matrix factorization (NMF) [10], independent component analysis (ICA) [9] and sparse coding [13] became of increasing interest in audio analysis. The basic idea is the introduction of hidden, not directly observable atoms. The above cited methods decompose an input matrix into two factors where one is a dictionary describing the individual atoms and the other gives the activation of these components as a function of time. The non-negativity constraint is derived from the areas of application where single observations linearly add up to the whole. For instance in audio analysis it would not make sense to consider notes with negative loudness. Since the only prior knowledge is the maximum number of independent components these algorithms are part of the group of bottom-up methods as described above.

In this paper we propose a new method where the ideas of matrix factorization and sparse, independent components are adapted to follow a model based approach. Studies on the human approach to music transcription [5] have shown that trained musicians use lots of background information like the style of the piece or the instruments playing. They expect certain timbres and therefore would for example never search for distorted guitar tones in a classical piano piece. Applying this principle to the non-negative matrix factorization, prior knowledge about the dictionary is incorporated. The activation matrix will then remain the only

unknown component which needs an operation like a non-negative matrix division in order to be calculated.

Section 2 of this paper focuses on the NMF and its shortcomings in the context of transcription of solo music as a motivation for our method that is then explained in detail. Section 3 describes how the tone models representing a certain instrument can be extracted from sample recordings. The post processing step transforming the activation patterns yielded by the matrix division into discrete note events is described in section 4. Sections on the experimentation results and our conclusions complete the paper.

2 PITCH DECOMPOSITION

2.1 Non-negative matrix factorization

Non-negative matrix factorization as introduced in [9] decomposes a non-negative input V of size $m \times n$ into two non-negative output matrices W and H of size $m \times r$ and $r \times n$ respectively, such that

$$V \approx W \cdot H \tag{1}$$

where by convention W is regarded as a set of basis vectors and H as the aggregation of their activation patterns.

Since perfect factorization is not possible in almost all cases, a solution to equation (1) with minimal error of reconstruction is achieved by minimizing a cost function over the difference between V and $W \cdot H$. Common such functions are the Euclidean distance $E(V, WH)$ or the Kullback-Leibler divergence $D(V \parallel WH)$.

Applied to a power spectrum, as obtained by the short time Fourier transform, the basis components in W are weighted frequency groups that are found to sound together. Ideally they belong either to a single pitch played on a certain instrument or a group of pitches that are normally played together like the notes of a chord. If the number r of basis components is smaller than the number of different pitches played, some of the pitches have to be either omitted or grouped within one atom. In the reverse case where r is sufficiently large there can be atoms representing noise, or there is more than one atom per one single pitch. Here it is very likely that a component represents the sustained part of a note whereas another maps to the note onset with much richer harmonics. A detailed investigation on these effects can be found in [13].

The component activation data in H contains the strength of each atom at a certain time frame. Due to the non-negativity constraint the combination is additive only. This gives consideration to the fact that there is nothing like negative loudness of notes.

Effective algorithms for the calculation of the NNMF have been introduced in [10]. Multiplicative update rules are used in order to find local minima starting form randomly initialized matrices W and H. Using the Kullback-Leibler divergence $D(V \parallel WH)$ as cost function these update rules are

$$H_{a\mu} \leftarrow H_{a\mu} \frac{\sum_i W_{ia} V_{i\mu} / (WH)_{i\mu}}{\sum_k W_{ka}} \tag{2}$$

$$W_{ia} \leftarrow W_{ia} \frac{\sum_\mu H_{a\mu} V_{i\mu} / (WH)_{i\mu}}{\sum_\nu H_{a\nu}} \tag{3}$$

In [10] a proof for the convergence of this algorithm towards a local minimum is given. It is shown that the divergence is (i) non-increasing under above update rules and (ii) invariant if and only if W as well as H are at stationary points.

2.2 Drawbacks of NNMF

Several works like [13] or [15] have concentrated on applying matrix factorization using non-negativity and sparseness constraints to automatic music transcription. Although there are numerous advantages, an inherent problem of NMF based approaches is the determination of an appropriate number r of base components. This parameter has to be guessed since the number of different pitches present in a piece of music is not known in advance. An r that is too small cannot represent each pitch individually whereas too large values cause increased computational expenses as well as difficulties when mapping base vectors in W to transcribed pitches.

Another drawback is that there is no guarantee that each played pitch is represented at all. [13] reports that although using a more than sufficiently large number of base vectors in an NMF as well as in two sparse coding approaches a few notes are not represented. This is the case when chords or certain residual noise patterns become more significant than single tones that are played only very rarely.

Thirdly learning a dictionary of independent components while transcribing music played by a single instrument does not seem to be a natural way of approaching the problem. As shown in [5] humans start by detecting the genre and style of a piece, which allows them to limit the number of possible instruments and timbres to be expected. Learning the dictionary of independent components along with their activation is, as pointed out above, likely to model noise as well and therefore prone to overfitting. Restricting dictionary vectors to feasible values in advance is a reasonable means of preventing overfitting as well as unnecessary computational costs.

2.3 Non-negative matrix division

To overcome the above drawbacks we propose fixing the number r of independent components to the number of actual possibilities regarding the pitch range of the instrument in focus, and the single atoms of the dictionary W to stereotypical tone models of the corresponding pitches. One approach would be to use multiplicative updates on a random initialization of H applying the rule in (2) while omitting (3) or defining constraints on its computation like done in [16]. But exploiting the fact that W is fixed and therefore single vectors of V can be processed independently, equation (1) resolves to

$$v \approx \overline{W} \cdot h \tag{4}$$

where $\overline{W}$ is the fixed dictionary. v and h are column vectors representing one time frame of the spectrogram and the pitch activation respectively. In order to measure the quality of an approximation in (4) again a cost function is needed. A convenient measure is the mean square criterion where

$$f = \frac{1}{2} \parallel \overline{W}h - v \parallel^2 \tag{5}$$

has to be minimized while regarding the constraint of non-negativity. According to [8] this problem is solved by an iterative algorithm as follows.

1. Initialize all elements of h to zero and introduce two sets P and Z where P is empty and Z contains all indices within h.
2. Compute the gradient $\bigtriangledown_h f = \overline{W}^T \cdot (v - \overline{W}h)$ where f is the cost function as defined in (5).
3. If $Z = \{\}$ or $\forall i : i \in Z \Rightarrow (\bigtriangledown_h f)_i \leq 0$ then terminate.
4. Find the maximum element of $\bigtriangledown_h f$ and move its index from Z to P.
5. Solve the unconstrained linear least squares problem $\overline{W}_{sub} \cdot z = v$ where $\overline{W}_{sub}$ is a copy of $\overline{W}$ where all columns corresponding to indices in Z are set to zeros. Within the result z only those elements with indices contained in P are significant. The others are set to zero.
6. If $\forall i : i \in P \Rightarrow z_i \geq 0$ then z is a feasible solution to the subproblem, h is set to z and the main loop is continued at step 2.
7. If the above condition does not hold z can only contribute to the new temporary solution up to a certain amount. Therefore a factor α (a learning rate) is calculated as $\alpha = argmin_i(h_i/(h_i - z_i))$ where only the indices of negative elements in z are allowed as i.
8. Calculate the new temporary solution using α from the above step as $h = h + \alpha(z - h)$
9. Move all indices for which the corresponding element in h is zero from P to Z. Continue working on the subproblem at step 5.

Although the result of one frame is a useful hint for the computation of the next frame, single time frames can now be independently processed. This makes the method suitable for parallelization as well as online processing.

Reassembling the results of individual frames gives a complete activation matrix like H from equation (1) as an optimal non-negative quotient of an input power spectrogram V and a given tone model dictionary W. The method can therefore be seen as a non-negative matrix division in contrast to the uninformed matrix factorization.

3 TONE MODEL LEARNING

The method as pointed out so far requires a given dictionary of tone models that has to be learned in advance. In this work an approach is explained where the same algorithm as for the pitch decomposition is used. The necessary training data consists of recordings of single pitches played on the particular instrument. Starting from equation (1) again instead of fixing W to a given dictionary, H is chosen to have a number of components $r = 1$. This does justice to the facts that there shall be exactly one basis vector per midi pitch and in a single training instance there is only one tone present. The values of H are set to the corresponding values of the amplitude envelope expressing the current loudness of the sound. Then the tone model is calculated as described in section 2.3 but having a fixed H instead of a fixed W.

In cases where only recordings of some notes and not the whole pitch range are available interpolation is applied. Given a fundamental frequency f_0 for which the tone model is not known the starting point are the two nearest frequencies f_l and f_h with given energy distribution such that $f_l < f_0 < f_h$. The interpolation is then done in two steps. At first for each bin's center frequency f within the spectrum of f_0 the corresponding frequency within the known models is calculated. In the second step the energy of f' is estimated using parabolic interpolation. Finally the approximations using the lower and the upper frequency model are combined by taking the average.

4 FRAME-WISE CLASSIFICATION AND TRANSCRIPTION

The steps described in sections 2.3 and 3 lead to a piano-roll-like representation of the activation of individual pitches given by the matrix H. In order to close the gap between this representation and a symbolic transcription in midi-format a final step of post-processing is still needed since the activation data (i) contains low energy noise as well as higher level outliers and (ii) is not separated into discrete note events.

All the above calculations have been at the level of individual frames. Since musical notes are in most cases stable over a certain amount of time it makes sense to use the information given within a local neighborhood. We do so by applying a median filter in order to smooth the activation curves as well es to eliminate very short fragments. In our experiments window lengths of around 100 ms have been found to yield good results.

Since the smoothed activation data is very sparse, discrete note events can easily be extracted by identifying segments with above zero activation while filling out very small gaps (up to 30 ms) that might be there even after smoothing.

5 EXPERIMENTAL RESULTS

5.1 Tone model learning

As our prime test environment we have chosen solo piano music. In order to learn the dictionary for our experiments we took recordings of single tones played on a computer controlled Bösendorfer SE290 grand piano like they were also used by Goebl [3]. Recordings were available for every fourth midi pitch at different velocity levels. The power spectrum was computed using a short time Fourier transform with a window length of 4096 samples and a hop size of 441 frames or 10 ms when having input data sampled at a rate of 44.1 kHz. Additional zero padding by a factor of 2 was used in order to get narrower frequency bins. The window used was a Hamming window.

The resulting spectrum was then preprocessed in two steps. At first a silence detector sets all frames which have a total energy below a certain threshold to zeros. Then in order to further suppress noise as well as to remove the bias due to the tone model learned from one specific instrument, magnitude warping and spectral average subtraction as described in [7] is performed. The spectrum $X(k)$ of the signal is assumed to be a combination of $S(k)$ representing the sound that is originally excited, $H(k)$ being the frequency response of the environment like the instrument body and an additional noise component $N(k)$, giving the decomposition

$$X(k) = H(k)S(k) + N(k) \tag{6}$$

In order to equalize the factor $H(k)$ the power spectrum $X(k)$ is magnitude warped by applying

$$Y(k) = \ln\left(1 + \frac{1}{g}X(k)\right) \tag{7}$$

The purpose of the term g is to normalize the spectrum such that the level of the noise $N(k)$ is close to one whereas the spectral peaks are much larger. Assuming that the major part of $X(k)$ is just noise floor and the peaks of $H(k)S(k)$ are quite sparse any outlier resistible average measure can be taken in order to find a feasible g. In our tests just using the minimum of $X(k)$ gave satisfying results as well. Due to this warping the influence of $H(k)$ is reduced.

The additive noise is suppressed by subtraction of a moving average $\overline{N}(k)$ within the logarithmic scale. The size of the sliding window has a width of 100 Hz but increases at higher frequency bands such that it always covers at least a range of ± 4 semitones with regard to the currently processed coefficient k. The moving average $\overline{N}(k)$ representing the noise floor is then subtracted from $Y(k)$ leaving the preprocessed spectrum $Z(k)$ as

$$Z(k) = \max\left\{0, Y(k) - \overline{N}(k)\right\} \tag{8}$$

According to [7] using the logarithmic scale gives clearly better results than the linear scale.

The preprocessed recordings of the single tones are then passed to the model learner as described in section 3. Recordings were available for every fourth pitch using velocities of 30, 50, 70, 90 and 110. The influence of loudness of the training data is weakened by its explicit consideration during the learning algorithm. But since individual harmonics fade out unequally the initial loudness still influences the resulting model. We overcome this effect by taking into account the models learned from all different velocities and calculating a final one by taking the average spectral power at each frequency bin. To complete the dictionary containing all midi pitches from 21 to 108, which constitutes the whole pitch range of the piano, the missing tone models were interpolated.

5.2 Transcription

As test data we use the recording of Mozart's sonata KV279 played by a professional pianist on a computer monitored Bösendorfer SE290 grand piano, giving us a precise ground truth of played notes in a midi-like format. The test set of 10.000 time frames contains 1087 keystrokes and continuous pedal events that are known as well. Although this covers only less than the

first two minutes of the piece it is a respectable basis for cross validation experiments as will be described in this section.

The data was converted to be monaural and transformed into the frequency domain using the same parameters and preprocessing as used for the dictionary learning. The only difference was that the upper half of the frequency bands were dropped in order to reduce computational costs on further operations. Tests have shown that this reduction of data causes hardly any loss of quality. The power spectrum was then processed using the non-negative matrix division approach including smoothing of the result as well as the extraction of discrete note events. First tests have shown that the resulting set of identified notes includes 1077 out of the 1087 present notes, missing less than 1%. However the amount of spurious events being more than three times as high as the number of correct notes was unacceptable and made further post-processing necessary.

The data showed that the higher the pitches, the higher are the activation levels of occurring spurious events. A reason might be that due to the disregard of the higher half of the spectrum the higher notes within the dictionary are only represented by the fundamental frequency and the first one or two harmonics. Such a base vector is more likely to match a noise pattern than one containing the whole range of harmonics as is the case with models of lower pitches.

For this reason it is not adequate to use a single magnitude threshold on the activation data in order to distinguish correct note events from spurious ones. Instead we applied a very simple rule based classifier (RB) that has one magnitude threshold per frequency band as the only rules.

We compared this classifier to a second, instance based, one. We decided to use a nearest neighbor algorithm (IB) having a broader basis of decision-making. The features used were pitch, length, maximum energy and the sum of energy of a note. A 10-fold cross validation on our data set was done to test the ability of these two classifiers to separate correct notes from spurious ones. The results are listed in table 1 showing that the instance based classifier clearly outperforms the rule based one on all measures defined as

$$precision = \frac{TP}{TP + FP} \tag{9}$$

$$recall = \frac{TP}{TP + FN} \tag{10}$$

$$f = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{11}$$

where TP are the correctly found notes, FN are the missed and FP the spurious notes. A note event is counted as correct if the transcribed and the real note do overlap. Cases where the sustain pedal is used are handled in the way that notes are allowed to last as long as the pedal is pressed but they do not need to since the actual length of the note cannot be told exactly.

	precision	recall	f
raw data	21.8%	99.1%	0.357
classifier (RB, notes)	85.9%	85.4%	0.856
classifier (IB, notes)	95.6%	88.1%	0.917
classifier (frames)	42.3%	68.8%	0.524

Table 1. Classification results on solo piano music

Determining note onsets and offsets by just considering if the activation is above zero is simple. However this usually leads to transcriptions into notes that are longer than the original ones. To overcome this drawback we applied another classifier to the raw activation data in order to decide whether a pitch is played or not for each frame individually. In our test set of 10.000 frames containing 88 pitches each, this leaves us with 880.000 instances with more than 20.000 instances belonging to an actually played note. The features given to the, again instance based, classifier were the ones used for note-wise processing with addition of the current activation of each instance.

The result is again shown in table 1. The recall means that almost 70% of the original sound is covered in the transcription. 42% of the play time within the result match with a real note whereas the remainder of 58% is made up by erroneous note elongations and spurious notes.

6 CONCLUSIONS

In this work we have proposed a modification to existing transcription approaches using non-negative matrix factorization. Instead of tackling the problem in an uninformed way the new method makes use of an a priori learned dictionary of base vectors or tone models. This transforms the problem from general factorization to a division problem. The methods for the calculation of activation magnitudes can also be applied to the initial model learning in order to yield an appropriate dictionary. An advantage over uninformed matrix factorization is that single time frames can be processed independently - a fact that can be utilized to reduce computational complexity.

Applied to solo piano music, the raw resulting activation patterns contained more than 99% of the original notes but more than three times as many spurious notes as well. A post processing step with quite simple classifiers achieved an overall f-value of about 90% for

note-wise detection of played notes. In comparison, the frame wise classification yields an f-value of about 0.5 which is significantly less accurate.

We believe that additional information like high-level musical knowledge could help to improve the final step of picking correct notes while neglecting spurious ones. Also the information from an additional onset detection could benefit the frame-wise detection accuracy. Steep slopes can be observed at the beginning of connected parts within the activation data. Yet their reliability for onset detection has not been investigated.

Another aspect that has not been considered yet is how to cope with situations where there is more than one instrument present, or pieces where the playing instrument is not known a priori. Although the preprocessing that was applied in the test environment does some spectral whitening and therefore reduces the influence of timbre we still expect the timbral correlation between the instrument used for the dictionary learning and the one that shall be transcribed to be essential.

7 ACKNOWLEDGMENTS

This research is supported by the Austrian Fonds zur Förderung der Wissenschaftlichen Forschung (FWF) under project number P19349-N15.

8 REFERENCES

[1] Brown, J. C. and Zhang, B. "Musical frequency tracking using the methods of conventional and narrowed autocorrelation", *Journal of the Acoustical Society of America 89 (5)*. pp. 2346–2354, 1991.

[2] Godsill, S. J. and Davy, M. "Bayesian harmonic models for musical signal analysis", *7th Valencia International meeting on Bayesian statistics*. Valencia, 2002.

[3] Goebl, W. "The Role of Timing and Intensity in the Production and Perception of Melody in Expressive Piano Performance", *Doctoral thesis*. Karl-Franzens-Universität Graz, Graz, 2003.

[4] Hainsworth, S. W. "Analysis of musical audio for polyphonic transcription", *1st Year PhD Report*. University of Cambridge, 2001.

[5] Hainsworth, S. W. and Macleod, M. D. "The Automated Music Transcription Problem", 2003.

[6] Klapuri, A. P. "Pitch estimation using multiple independent time-frequency windows", *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. New York, 1999.

[7] Klapuri, A. P. "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness", *IEEE Trans. Speech and Audio Processing 11(6)*. pp. 804–816, 2003.

[8] Lawson, C. L. and Hanson R. J. *Solving least squares problems*. Prentice Hall, Lebanon, Indiana, 1974.

[9] Lee, D. D. and Seung, H. S. "Learning the parts of objects by non-negative matrix factorization", *Nature 401*. pp. 788–791, 1999.

[10] Lee, D. D. and Seung, H. S. "Algorithms for Non-Negative Matrix Factorization", *Neural Information Processing Systems*. Denver, 2000.

[11] Lesser, V.; Nawab, H.; Gallastegi, I. and Klassner, F. "IPUS: An architecture for integrated signal processing and signal interpretation in complex environments", *Proceedings of the AAAI*. Washington, 1993.

[12] Martin, K. D. "A backboard system for automatic transcription of simple polyphonic music", *Technical report TR.385*. Media Laboratory, MIT, 1996.

[13] Plumbley, M. D.; Samer, A. A.; Blumensath, T. and Davies, M. E. "Sparse Representation of Polyphonic Music", *Signal Processing 86/3*. pp. 417–431, 2006.

[14] Ryynänen, M. P. and Klapuri, A. "Polyphonic Music Transcription using Note Event Modeling", *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. New York, 2005.

[15] Smaragdis, P. and Brown, J. "Non-negative matrix factorization for polyphonic music transcription", *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. New York, 2003.

[16] Vincent, E.; Bertin, N. and Badeau, R. "Harmonic and inharmonic nonnegative matrix factorization for polyphonic pitch transcription", *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*. Las Vegas, 2008.