Evaluation of the Convolutional NMF for Supervised Polyphonic Music Transcription and Note Isolation

Stanislaw Gorlow* and Jordi Janer

Universitat Pompeu Fabra — Music Technology Group†
Roc Boronat 138, 08018 Barcelona, Spain

Abstract. We evaluate the convolutive nonnegative matrix factorization in the context of automatic music transcription of polyphonic piano recordings and the associated problem of note isolation. Our intention is to find out whether the temporal continuity of piano notes is truthfully captured by the convolutional kernels and how the performance scales with complexity. Systematic studies of this kind are lacking in existing literature. We make use of established measures of accuracy and similarity. NMF dictionaries covering the piano's pitch range are learned from a given sample bank of isolated notes. The kernel alias patch size is varied. By using a measure of performance advantage, we show up that the improvements due to convolved bases do not justify the extra computational effort as compared to the standard NMF. In particular, this is true for the more realistic case, in which the dictionary does not fully correspond to the mixture signal. Further pertinent conclusions are drawn as well.

Keywords: Nonnegative matrix factorization, convolution, supervised learning, polyphony, automatic music transcription, note separation

1 Introduction

Nonnegative matrix factorization (NMF) [1] meanwhile is an established tool in music processing, and music transcription has emerged as its main area of application, see [2, 3]. Since a complete transcription would also include a note's velocity, this information, together with the learned bases, can be used to isolate notes from the mixture by Wiener filtering. This can be done either in a supervised or in an unsupervised task.

In this study, we seek to compare the performance of the convolutional NMF [4, 5] with the standard NMF in regard to supervised learning, i.e. where the bases are held fixed and their activations are updated until the modeled spectrogram is in the shortest distance from the observed spectrogram. In our evaluation, we resort to more frequently used measures, such as the root-mean-square deviation (RMSD). We also provide perception-related ratings. Above, we are interested in seeing how the superior modeling accuracy that convolutional bases are expected to bring about relates to the extra computational effort. As the convolutional NMF was designed for capturing the

* S. Gorlow is now with Sony Computer Science Laboratory (CSL) in Paris, France.
† This work was funded in part by the Yamaha Corporation.

temporal evolution of sound patterns, we expect it to track the temporal decay of notes more faithfully than the standard NMF. For the transcription of polyphonic recordings and the related task of note isolation, temporal continuity of notes is a crucial factor. Thus, the convolutional NMF looks promising and seems to be a reasonable alternative to other variants that favor temporal continuity through additional penalty terms in the cost function. More generally speaking, our interest is in evaluating the aptitude of the convolutional NMF for musical applications.

2 Convolutional NMF

The basic idea behind the convolutional or convolutive NMF is to treat sequences of single-column bases, or multi-column bases, in the exact same manner that single-column bases are treated by the standard NMF. This is meant to better capture the temporal evolution of repeating patterns of the dominant or principal components in the mixture as compared with the standard, i.e. non-convolutional, NMF. In our case, we mean sequences of magnitude and/or power spectra when speaking of patterns and the principal components are piano notes. We will further refer to the length of such a sequence of spectra as the "patch size". The rank of the factorization is given by the number of distinct piano notes.

Now consider a Bregman distance $D_F^{\mathbf{X}}$ formally given in the form of the Kullback–Leibler (KL) divergence with $\mathbf{X}$ of size $K \times N$, $x_{kn} \in \mathbb{R}_0^+$, being approximated as

$$\mathbf{X} \approx \mathbf{Y} = \sum_{m=0}^{M-1} \mathbf{S}(m) \cdot \mathbf{A} \,\mathrm{rshift}\, m \ , \tag{1}$$

where $\mathbf{A}$ is the activations matrix, rshift is the zero-fill right-shift operator applied to the rows of $\mathbf{A}$, $\mathbf{S}$ is the bases matrix or the spectral imprint, m is the patch index and M the patch size, respectively. To show that (1) is indeed a convolution in n, we need to consider the following term which is applied to every element of $\mathbf{Y}$,

$$y_{kn} = \sum_{r=1}^{R} \sum_{m=0}^{M-1} s_{kr}(m) \cdot a_{r,n-m} = \sum_{r=1}^{R} s_{kr}(n) * a_{rn} \ , \tag{2}$$

where $*$ denotes convolution and R is the rank of $\mathbf{Y}$, $R \ll \min(K, N)$. The generalized KL divergence w.r.t. $\mathbf{X}$,

$$D_{\mathrm{KL}}^{\mathbf{X}}(\mathbf{Y}, \mathbf{X}) = \sum_{k,n} y_{kn} \log \frac{y_{kn}}{x_{kn}} - \sum_{k,n} y_{kn} + \sum_{k,n} x_{kn} \ , \tag{3}$$

is generated from the convex function

$$F(\mathbf{X}) = \sum_{k,n} x_{kn} \log x_{kn} - \sum_{k,n} x_{kn} \ . \tag{4}$$

Alternatively, the KL divergence from (3) can be replaced by [2]

$$D_{\mathrm{KL}'}^{\mathbf{X}}(\mathbf{Y}, \mathbf{X}) = \left\| \mathbf{Y} \odot \log\left(\mathbf{Y} \oslash \mathbf{X}\right) - \mathbf{Y} + \mathbf{X} \right\|_{\mathrm{F}} \ , \tag{5}$$

where $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm, $\oslash$ stands for element-wise division and $\odot$ for element-wise multiplication, respectively. Note that for $M = 1$, (1) turns into the standard NMF. So, in supervised learning, the problem at hand can be stated as follows. Given $\mathbf{X}$ and $\mathbf{S}$, $s_{kr} \in \mathbb{R}_0^+$, $R \ll \min(K, N)$, $M \in \mathbb{N}$, find

$$\mathbf{A}_{\mathrm{opt}} = \arg\min_{\mathbf{A}} D_{\mathrm{KL}'}^{\mathbf{X}}(\mathbf{Y}, \mathbf{X}) \qquad \text{s.t. } a_{rn} \in \mathbb{R}_0^+ \ . \tag{6}$$

2.1 Multiplicative Update Rule

$\mathbf{A}_{\mathrm{opt}}$ in (6) can be found using the convolutional update rule given in [5], which is

$$\mathbf{A} \leftarrow \mathbf{A} \odot \left[\mathbf{S}^{\mathsf{T}}(m) \cdot (\mathbf{X} \oslash \mathbf{Y}) \,\mathrm{lshift}\, m\right] \oslash \left[\mathbf{S}^{\mathsf{T}}(m) \cdot \mathbf{1}\right] \ , \tag{7}$$

where $\mathbf{1}$ is a $K \times N$ all-ones matrix and lshift stands for the row-wise zero-fill left-shift operator. In [5], it is further suggested that for each $\mathbf{S}(m)$ a different $\mathbf{A}_m$ should be learned and that the final $\mathbf{A}$ should be computed as $\mathbf{A} = \langle \mathbf{A}_m \rangle$, where $\langle \cdot \rangle$ denotes the time average operator,

$$\langle \mathbf{A}_m \rangle = \frac{1}{M} \sum_{m=0}^{M-1} \mathbf{A}_m \ . \tag{8}$$

2.2 Dictionary Learning and Normalization

To construct an instrument's dictionary, one requires a dataset of separate note recordings. A typical piano range, e.g., would consist of $I = 88$ notes, starting with A_0 and ending with C_8, at a distance of a semitone. For every ith note, one computes the spectrogram $\mathbf{X}_i$ and learns the corresponding patch of M bases using [5]

$$\mathbf{S}_i(m) \leftarrow \mathbf{S}_i(m) \odot \left[(\mathbf{X}_i \oslash \mathbf{Y}_i) \cdot (\mathbf{A}_i \,\mathrm{rshift}\, m)^{\mathsf{T}}\right] \oslash \left[\mathbf{1} \cdot (\mathbf{A}_i \,\mathrm{rshift}\, m)^{\mathsf{T}}\right] \ , \tag{9}$$

while alternating with (7). In a final step, the M $\mathbf{A}_i$ matrices are discarded and the convolutional bases $\mathbf{S}_i(m)$ are kept. The overcomplete dictionary, which is held stiff in (6), is obtained by stringing the note patches together to

$$\mathbf{S}(m) = \left[\mathbf{S}_1(m)\ \mathbf{S}_2(m) \cdots \mathbf{S}_I(m)\right] \qquad \in \mathbb{R}^{K \times MI} \ . \tag{10}$$

After each update (9), it is very common to normalize the columns of $\mathbf{S}_i(m)$ by their lengths in the Euclidean space. A reason for doing this is numerical stability. Another way of normalizing is by patch, i.e. either by relating each matrix element to the largest singular value of $\mathbf{S}_i(m)$, by taking the Euclidean matrix norm, or as an alternative by dividing each matrix element by the Frobenius norm. In this wise, the temporal decay of the notes' spectral envelopes can be tracked.

2.3 Gaussian-Additive Mixture Model

Consider the short-time Fourier transform (STFT) domain. In reference to the central limit theorem, the Fourier coefficients are approximately complex-normally distributed. We further assume that they are circularly-symmetric, i.e. that they have zero mean and zero covariance matrix. Now, if we stipulate that the note components are mutually independent, the mixture's PSD can be decomposed into a sum of notes' PSDs. In other words, the NMF can be performed on the mixture's PSD. Yet note that this model does not hold for the magnitude spectra, as the square root of a sum of squares is not equal to the sum of magnitudes.

2.4 Note Separation

With the signal model from Section 2.3, Wiener filtering can be used to separate the note components from the mixture. In a first step, one computes the learned spectrograms

$$\widehat{\mathbf{Y}}_i = \sum_{m=0}^{M-1} \mathbf{S}_i(m) \cdot \hat{\mathbf{A}}_i \,\mathrm{rshift}\, m \ , \tag{11}$$

$i = 1, 2, \ldots, I$, and applies Wiener filtering to every element separately:

$$\hat{z}_{ikn} = \frac{\hat{y}_{ikn}}{\sum_{j=1}^{I} \hat{y}_{jkn}} \cdot x_{kn} e^{\jmath \phi_{kn}} \qquad \forall\, i, k, n \ , \tag{12}$$

where ϕ is the phase of x in time-frequency (TF) point (n, k) and $\jmath$ is the imaginary unit. The corresponding time-domain signal is obtained by the inverse STFT on $\widehat{\mathbf{Z}}_i$.

3 Evaluation

For the purpose of evaluation, we design various dictionaries using the RWC Music Database, while each dictionary is trained for Yamaha's Pianoforte, normal playing style, and "mezzo" level of dynamics.[1] For the STFT, we apply a 4-term Blackman–Harris window of the size of the transform and overlap succeeding blocks by 87.5 %.

As for the mixture signal, we generate it from a MIDI file taken from the Saarland Music Data (SMD) using Kontakt 5 by Native Instruments.[2] The 32-s excerpt is part of Chopin's Opus 10.[3] We generate two mixtures: one synthetic using the RWC samples and one realistic for the Berlin Concert Grand. We perform the NMF on the mixture using the NMFlib for a fixed number of 30 iterations.[4][5] The critical testing parameter is the patch size M which is increased from 1 onwards. Also, we evaluate the NMF performance for the transform lengths of 2048 and 4096 points for two different non-negative TF representations: the magnitude spectrum and the power spectrum. Overall, we train 24 dictionaries, one for each set of configuration parameters. We normalize the basis spectra by patch using the Euclidean matrix norm.

[1] https://staff.aist.go.jp/m.goto/RWC-MDB/

[2] http://www.mpi-inf.mpg.de/resources/SMD/SMD_MIDI-Audio-Piano-Music.html

[3] The results shown are representative of what we experienced for different piano recordings.

[4] https://code.google.com/p/nmflib/

[5] The number was chosen empirically. Above it, no significant improvement was observed.

3.1 Performance Measures

***F*-measure** In binary classification, the F-measure indicates the accuracy of a system under test and it is defined as the harmonic mean of precision and recall:

$$F \triangleq 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \ , \tag{13}$$

where TP is the number of true positives, FP is the number of false positives and FN is the number of false negatives. In the case of music transcription, true positives denote those TF points that have significant contributions according to (11) in the same spots as in the perfect transcription. False positives are activations in the wrong spots and false negatives denote missing activations, respectively. The F-score attains its best value at 1 and its worst value at 0.

Root-mean-square deviation The root-mean-quare deviation (RMSD) is a frequently used measure of accuracy for comparing errors of different models for a particular variable. With regard to notes:

$$\text{RMSD}_i \triangleq \sqrt{\frac{1}{N} \sum_{n=1}^{N} \left[\hat{z}_i(n) - s_i(n)\right]^2} \ , \tag{14}$$

where N is the length (in samples) of the time-domain signal $s_i(n)$ and $\hat{z}_i(n)$ is its estimate. Lower values are preferred.

Perceptual similarity measure "PEMO-Q" [6] is a method for the objective assessment of the perceptual quality of audio. It uses the model of auditory perception by Dau et al. to predict the audio quality of a test signal relative to a reference signal. PEMO-Q aligns the levels of both signals and transforms them into so-called "internal representations" of the auditory model. The cross-correlation coefficient between the two representations serves as a measure of the perceived similarity, PSM. And so, it can be used as a measure of the test signal's degradation.

Average performance and performance advantage The major goal of this evaluation is to relate the performance of the convolutional NMF to its computational complexity in a more formal manner. We state the average performance as

$$P_{\text{avg}} \triangleq \frac{P}{T} \ , \tag{15}$$

where P can be expressed as any of the above measures and T shall denote the execution time of the NMF. Moreover, we define the performance advantage of the convolutional NMF as the logarithm of the ratio between the performances of the convolutional and the standard NMF over time,

$$PA \triangleq \log \frac{P_M / T_M}{P_1 / T_1} \approx \log \frac{P_M}{M \cdot P_1} \tag{16}$$

with $T_M \approx M \cdot T_1$, i.e. on the assumption that it takes M times longer to compute the convolutional M-basis NMF as compared to the standard single-basis NMF [5]. A PA that is above zero indicates an advantage, a disadvantage if below zero, i.e. if it is negative, and a value of zero means equality.

3.2 Music Transcription and Note Isolation

In the first part of our evaluation, we compute the accuracy of the convolutional NMF as a function of the patch size M for different configurations using the F-measure. The perfect or reference transcription is computed from the score. For each note, we obtain a waveform signal from the respective MIDI track using the Kontakt 5 sampler. For all notes, we compute the time-pitch power spectra. We compare the powers with a threshold of -60 dB, and so we obtain a binary mask for the entire excerpt. The same thresholding procedure is applied to each note signal estimated according to (12). The two binary masks are then compared against each other in terms of (13). Errors are manifested in missing or superfluous positives that represent a mismatch between the signal and the model. Fig. 1 summarizes the results.

In the second part, we evaluate the quality of separated notes. For this, we use the RMSD and the PSM. All note signals are normalized to 0 dBFS RMS before computing the RMSD. In Fig. 2, the average over all isolated notes is shown.

3.3 Interpretation of Results and Observations

Looking at Fig. 1, one can observe a slight improvement that is due to a greater patch size in the case of the synthetic mix. For the realistic mix, the improvement is minor. The greater patch size seems rather counterproductive when the power is used as the nonnegative representation together with a lower frequency resolution. It looks like the magnitude spectra yield a better accuracy for both the mixtures. It is also evident that a higher frequency resolution improves the transcription. The fact that some curves are not monotonically increasing might be due to random initializations in the NMFlib.

Fig. 2 confirms once more that a significantly better result can be expected if the dictionary fits the mixture. In regard to the RMSD, a gain of 3 dB can be stated. Here again, a higher frequency resolution has a stronger impact on the result than a larger patch. When listening to the note samples, we would further observe that for low-pitched notes a 2048-point STFT is insufficient to discriminate neighboring partials. For high-pitched notes, this issue is less critical. For the synthetic mix, the perceptual similarity between notes is higher in the case of magnitude spectra. Yet for the realistic mixture, the power spectral representation gives comparable if not better results.

Even though a performance improvement with respect to the F-measure and also the PSM is undeniable between 1 and 4 bases in particular, the PA-curves indicate that it comes at the expense of an almost M times higher effort. For a patch size greater than 4, the improvement looks negligible in most cases. Plus, irrespective of the chosen test case and measure, $PA \approx -\log M$, i.e. negative (disadvantageous) for all $M > 1$. And what is more, the improvement is scarcely audible. Another negative side effect of the convolution worth noting is that the attacks of low-pitched notes are smoothed out.

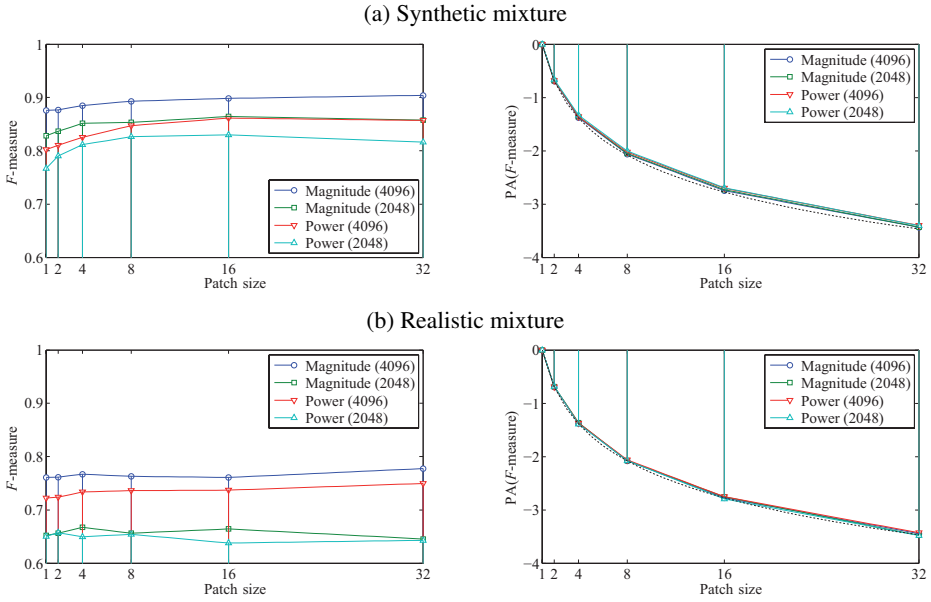

Fig. 1. F-measure values versus performance advantage of the convolutional NMF for music transcription

4 Conclusion

We conclude that because of the large pitch range of the piano, the STFT size should be no smaller than 4096 at a sampling rate of 44.1 kHz to separate low-pitched notes. As for the spectral representation, in most test cases the magnitude spectrum is more performant than the power spectrum. At this point, we do not have an explanation for this enigma that questions the validity of the Gaussian-additive mixture model. Finally, the study shows that is it senseful to favor a single-basis NMF over a computationally intensive convolutional NMF in musical applications, especially if the runtime plays an important role. No significant sound quality loss was established in our experiments.

References

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

[2] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *Proc. WASPAA 2003*, Oct. 2003, pp. 177–180.

[3] S. A. Abdallah and M. D. Plumbley, "Polyphonic music transcription by non-negative sparse coding of power spectra," in *Proc. ISMIR 2004*, Oct. 2004, pp. 318–325.

[4] P. Smaragdis, "Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs," in *Proc. ICA 2004*, Sep. 2004, pp. 494–499.

[5] ——, "Convolutive speech bases and their applications to supervised speech separation," *IEEE Audio, Speech, Language Process.*, vol. 15, no. 1, pp. 1–12, Jan. 2007.

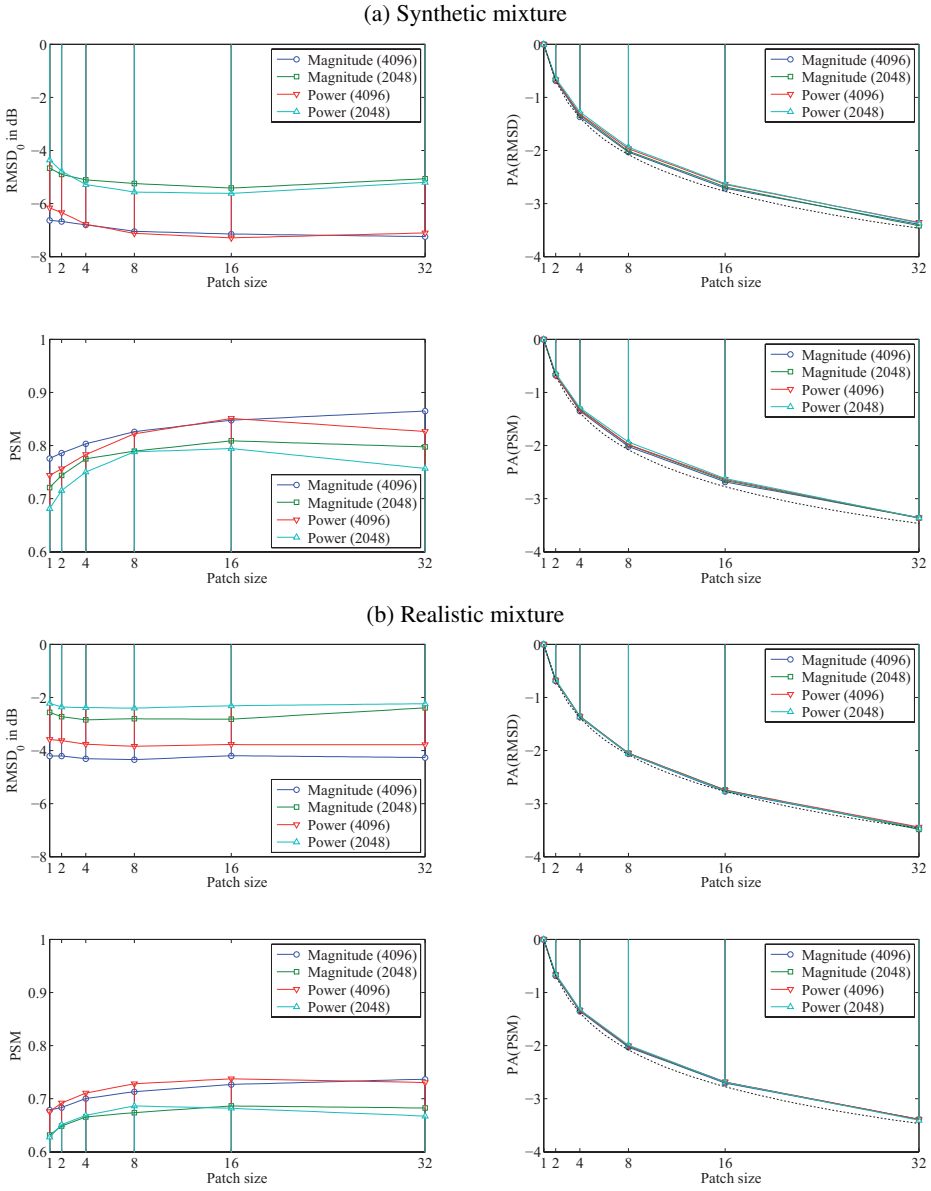

Fig. 2. RMSD and PSM values versus performance advantage of the convolutional NMF for note isolation

[6] R. Huber and B. Kollmeier, "PEMO-Q — a new method for objective audio quality assessment using a model of auditory perception," *IEEE Audio, Speech, Language Process.*, vol. 14, no. 6, pp. 1902–1911, Nov. 2006.