SHIFTED NON-NEGATIVE MATRIX FACTORISATION FOR SOUND SOURCE SEPARATION

Derry FitzGerald, Matt Cranitch

Dept. of Electronic Engineering
Cork Institute of Technology
Rossa Avenue, Bishopstown, Cork, Ireland

Eugene Coyle

Dept. of Electronic Engineering
Dublin Institute of Technology
Kevin St. Dublin Ireland

ABSTRACT

A shifted non-negative matrix factorisation algorithm is derived, which offers advantages over previous matrix factorisation techniques for the purposes of single channel source separation. It represents a sound source as translations of a single frequency basis function. These translations approximately correspond to notes played by an instrument. Results are presented for a set of synthetic data, and on a single channel recording of piano and clarinet. Though the system is aimed at musical recordings, the technique can be applied to any data which contains shifted versions of an underlying factor, and so the algorithm could possibly be used in other applications such as image processing.

1. INTRODUCTION

In recent years, a number of systems have been proposed that attempt matrix factorisation into sets of outer product basis functions. These include Non-negative Sparse Coding (NNSC) [5] and Non-negative Matrix Factorisation (NNMF) [6]. These systems have found use in single channel sound source separation [7, 8]. Further, both NNSC and NNMF have been used to attempt the transcription of polyphonic music of a single instrument [9, 10].

All these methods attempt to factorise a data matrix $\mathbf{X}$ into matrix factors $\mathbf{A}$ and $\mathbf{S}$ such that $\mathbf{X} \approx \mathbf{AS}$, where $\mathbf{X}$ is an n x m matrix, $\mathbf{A}$ is an n x r matrix, and $\mathbf{S}$ is an r x m matrix, with r smaller than n or m. This results in a compressed version of the original data matrix. The main difference between the systems lies in how the factorisation into outer products is achieved.

The Independent Subspace Analysis (ISA) algorithm proposed by Casey uses Principal Component Analysis (PCA) for dimensional reduction, followed by Independent Component Analysis (ICA) to achieve independence of the basis functions. NNSC uses a cost function that balances the reconstruction of the data matrix with the sparsity of the recovered components, along with additional constraints to ensure the non-negativity of the sources. NNMF uses a generalised Kullback-Liebler divergence between the spectrogram and the reconstruction of the spectrogram, and uses multiplicative updates to ensure the basis functions are non-negative.

In the context of single channel sound source separation, the single channel is typically transformed into a time-frequency representation such as a magnitude or power spectrogram. When factorisation takes place on the spectrogram, the columns of $\mathbf{A}$ contain frequency basis functions, and $\mathbf{S}$ contains a set of amplitude envelopes associated with the frequency basis functions. Ensuring that the factorisation is non-negative is particularly useful when decomposing a spectrogram, as a spectrogram contains only non-negative data, and so any decomposition which reflects this is more likely to give meaningful results. As a result, NNMF and NNSC based techniques are currently finding favour over ISA, which was the first of these decomposition methods used for single channel source separation.

A problem with using the above methods for single channel sound source separation is that the factorisation is linear. In practice, this means that each frequency basis function can only describe a single note, or group of notes such as a chord. As a result, these methods were particularly suited to sounds which do not change in pitch from occurrence to occurrence, such as drum sounds, and so these methods have found use in the automatic transcription of percussion instruments [1, 2, 3]. As most musical signals involve changes in pitch, this restriction means that some form of grouping must be carried out after using the above methods to obtain separated sources which change in pitch. Grouping methods have been proposed in [7] and [8]. Despite the existence of these grouping methods,this has been a serious limitation on the usefulness of these algorithms to-date.

It can therefore be seen that an extended model is needed to deal with the situation where various notes from the same instrument occur over the course of the mixture spectrogram. Previous work attempting to do this includes the non-linear ISA technique proposed by Vincent et al [11]. The remainder of this paper presents an alternative method for attempting to solve this problem which does not involve the learning of source priors before attempting separation.

2. SHIFTED NON-NEGATIVE MATRIX FACTORISATION

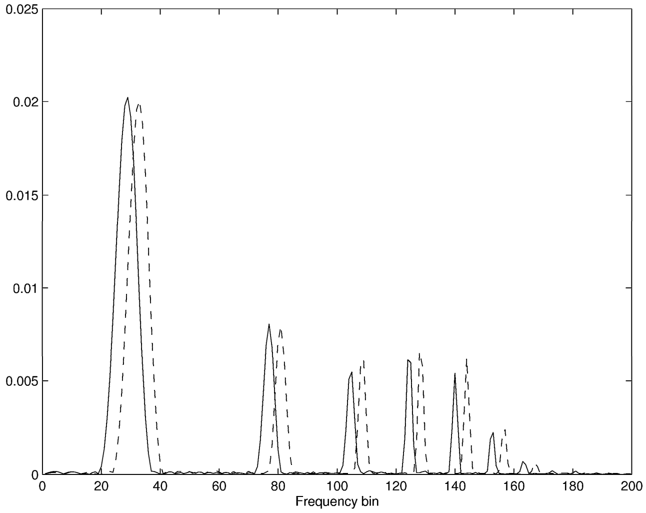

Fig. 1. Spectra of two notes played by French Horn

A potential way of overcoming the problem of dealing with multiple notes belonging to a single source is to assume that the notes belonging to a single source consist of translated versions of a single frequency basis function. This single frequency basis function is then taken to represent the typical frequency spectrum of any note played on the instrument in question. This is a simplified approximation of the real situation, where the frequency spectrum of the note does vary with pitch. Nevertheless, this assumption does represent a valid approximation over a limited pitch range. Indeed, a version of this assumption is used in commercial music samplers and synthesisers, where a recorded note of a given pitch is used to generate other notes in proximity to the original note. The use of this assumption also places a further restriction on the type of spectrogram being analysed, namely that the frequency resolution of the spectrogram must be logarithmic in scale.

A suitable method of obtaining such a frequency resolution would be the use of the Constant Q Transform (CQT) [4]. If the frequency resolution of the transform is set so that the center frequencies of each band are $2^{1/12}$ apart, then the spacing between frequencies will match that of the even-tempered tuning system used in western popular music. In this case, translating a frequency basis function of a note up or down by one position is equivalent to a pitch change of one semi-tone in the even-tempered scale. Figure 1 shows two notes played by a french horn. It can be seen that the frequency spectra of the two notes are quite similar, and hence either of the notes could be approximated by a translation of the other note. Further, it is assumed that no important information is contained in the extremities of the frequency basis function. It can be seen that assumption is valid for the frequency spectra in Figure 1, and can be ensured by setting suitable limits on the maximum and minimum frequencies in the CQT.

The following conventions are used in the remainder of this paper. Indexing of elements within a matrix or tensor, usually denoted by $\mathbf{X}_{i,j}$ is here denoted by $\mathbf{X}(i,j)$. Tensors are denoted by calligraphic uppercase letters, eg. $\mathcal{T}$, and contracted product multiplication of two tensors is defined as follows. If $\mathcal{W}$ is a tensor of size $I_1 \times \cdots \times I_N \times J_1 \times \cdots \times J_M$ and $\mathcal{Y}$ is a tensor of size $I_1 \times \cdots \times I_N \times K_1 \times \cdots \times K_P$ then contracted product multiplication of the two tensors along the first N modes is given by:

$$\langle \mathcal{W}\mathcal{Y} \rangle_{\{1:N,1:N\}}(j_1,\ldots,j_m,k_1,\ldots,k_p) = \sum_{i_1=1}^{I_1} \cdots \sum_{i_N=1}^{I_N} \mathcal{W}(i_1,\ldots,i_N,j_1,\ldots,j_M) \mathcal{Y}(i_1,\ldots,i_N,k_1,\ldots,k_P)$$

In this notation, the modes to be multiplied are specified in the subscripts that follow the angle brackets. These are the conventions adapted by Bader and Kolda in [12].

To translate a given n x 1 vector, an n x n translation matrix can be used. Such a translation matrix can be generated using the identity matrix and rearranging the columns. For example, to achieve a shift up of one, the translation matrix would be obtained from $\mathbf{I}(:,[n,1:n-1])$ where $\mathbf{I}$ denotes the identity matrix, and where the ordering of the columns is contained in the square brackets. An example of this is given below:

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \\ 4 \\ 5 \\ 1 \end{pmatrix}$$

For k possible translations, these translation matrices can be grouped into a translation tensor $\mathcal{T}$ of size n x k x n.

For r sources the frequency basis functions are contained in an n x r tensor denoted $\mathcal{A}$. The translated versions of these basis functions are then obtained from:

$$\mathcal{P} = \langle \mathcal{T}\mathcal{A} \rangle_{\{3,1\}}$$

where $\mathcal{P}$ is a tensor of size n x k x r

Then a spectrogram $\mathbf{X}$ of size n by m can be decomposed as follows:

$$\mathbf{X} \approx \langle \mathcal{P}\mathcal{S} \rangle_{\{2:3,1:2\}}$$

where $\mathcal{S}$ is a tensor of size k x r x m containing the amplitude envelopes associated with each translation of each source.

The generalised Kullback-Liebler divergence from NNMF is used as a cost function. This divergence is given by:

$$D(F\|G) = \sum_{ij}(F(i,j)log\frac{F(i,j)}{G(i,j)} - F(i,j) + G(i,j))$$

In the case of the decomposition described above, this divergence becomes:

$$D(\mathbf{X}\|\langle\mathcal{PS}\rangle_{\{2:3,1:2\}}) =$$

$$\sum_{ij}\left(X(i,j)log\frac{X(i,j)}{\langle\mathcal{PS}\rangle_{\{2:3,1:2\}}} - X(i,j) + \langle\mathcal{PS}\rangle_{\{2:3,1:2\}}\right)$$

Using this divergence the following multiplicative update equations can be derived.

$$\mathcal{S} = \mathcal{S}. * \langle\mathcal{PD}\rangle_{\{1,1\}}./\langle\mathcal{PO}\rangle_{\{1,1\}}$$

where $.*$ denotes elementwise multiplication and $./$ denotes elementwise division, and where $\mathcal{O}$ is an all ones tensor of size n by m. $\mathcal{D}$ is defined as:

$$\mathcal{D} = \mathbf{X}./\langle\mathcal{PS}\rangle_{\{2:3,1:2\}}$$

The update equation for $\mathcal{A}$ is then given by:

$$\mathcal{A} = \mathcal{A}. * \langle\mathcal{WS}\rangle_{\{[1,3],[1,3]\}}./\langle\mathcal{QS}\rangle_{\{[1,3],[1,3]\}}$$

where $\mathcal{W} = \langle\mathcal{TD}\rangle_{\{1,1\}}$, and $\mathcal{Q} = \langle\mathcal{TO}\rangle_{\{1,1\}}$. Once the initial estimates of $\mathcal{A}$ and $\mathcal{S}$ are set to positive values, the multiplicative updates ensure that the factorisation is non-negative. Although the convergence proofs given in [6] do not apply, it has been found that, in practice, the algorithm converges reliably.

3. RESULTS

The above algorithm was implemented in Matlab using the Matlab Tensor Classes written by Bader and Korda, available at [13]. Initially the algorithm was tested on synthetic data consisting of a single harmonic spectrum shifted up and down, and multiplied by a set of binary amplitude envelopes. In this case, the data fits the assumptions inherent in the algorithm perfectly and so the algorithm should be able to recover the underlying harmonic spectrum, subject to shifting and scaling factors. Both $\mathcal{A}$ and $\mathcal{S}$ were randomly intialised to positive values, and the number of sources set to 1. The algorithm converged after 50 iterations. Figure 2 shows the data set used to test the algorithm, while Figure 3 shows the normalised harmonic spectra of the input, shown as a solid line, and the harmonic spectrum recovered using the algorithm, shown as a dotted line. It can be seen that the algorithm has indeed recovered the underlying harmonic spectrum used to generate the test set. This demonstrates that the algorithm works as designed.

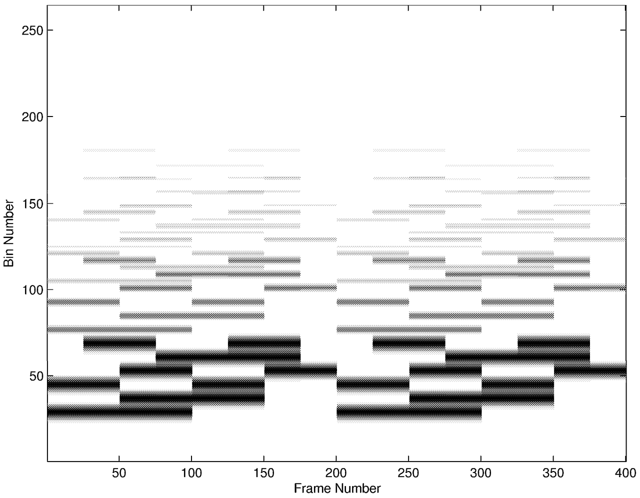

Fig. 2. Synthetic Test Data generated using a single shifted harmonic spectrum

Further to this, the algorithm was tested using a single channel signal generated from sampled notes of both piano and clarinet. Four samples used per octave in order to ensure that the same sample was not used for all notes played, thus making the test more realistic. The signal was transformed to a time-frequency representation using the Constant Q Transform [4] and a magnitude spectrogram obtained. Both $\mathcal{A}$ and $\mathcal{S}$ were randomly initialised to positive values and the number of sources r set to two. The number of allowable translations, k, was set to 15. The algorithm converged after 300 iterations.

Figure 4 shows the spectrogram obtained from the mixture signal of clarinet and piano. The piano melody can be seen clearly by following the lowest harmonics visible in the spectrogram, while the clarinet melody can be followed from the straighter harmonics visible above those of the lowest piano harmonics. Figure 5 shows the separated piano spectrogram obtained from the shifted NNMF algorithm described above, while Figure 6 shows the separated clarinet spectrogram.

It can be seen that the sources have been separated quite well, though there are still errors in the separation, with some notes from the clarinet showing up in the piano spectrogram and vice-versa. The largest error occurs where the third piano note has in effect been taken as a clarinet note. Inspection of the input signal revealed that this was because the clarinet note and piano note playing simultaneously were an octave apart, and so all the harmonics of the piano note overlap with those of the clarinet. Further, it is the only time

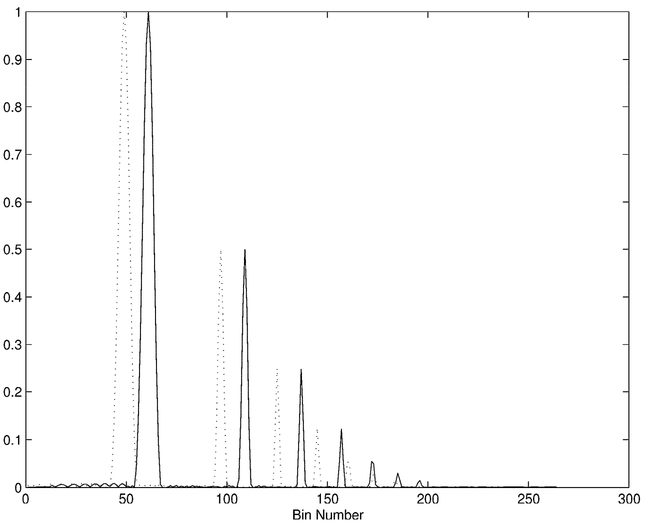

Fig. 3. Underlying harmonic spectrum of test data (solid line), and harmonic spectrum recovered using Shifted NNMF (dotted line)

these two notes occur in the signal, and so the algorithm does not have enough information to separate the two notes correctly. If another instance of either note was present, the algorithm would then have had sufficient information for separation to occur. This shows that the more instances of a note from a given instrument are present, the more likely the correct separation is to occur. Nevertheless, separation of the two sources has occurred using the shifted NNMF algorithm, demonstrating it's potential use as a means of single channel source separation. It also demonstrates that the assumption that an instrument or source can be viewed as translations of a single frequency basis function hold reasonably well over a limited pitch range.

The algorithm shows sensitivity to the choice of the number of allowable translations k. Too small a number results in the sources being amalgamated into a single source, while too large a number results in the recovery of an S which does not contain information which is not recognisable as being associated with a given source. Nonetheless, the results obtained with the algorithm are encouraging.

4. CONCLUSIONS

Having discussed the limitations of previous matrix factorisation techniques with regards to single channel sound source separation, it was proposed that to over come limitations in these techniques by assuming that a sound source or instrument could be represented as translations of a single frequency basis function. An algorithm based on NNMF which allowed for such translations was then derived. The algorithm was successfully tested on synthetic data which met the underlying assumptions of the data. It was further tested on a single channel recording of clarinet and piano generated from sampled notes of each instrument. Separation of the two sources was demonstrated, illustrating the utility of the approach. While the algorithm was derived with polyphonic music in mind, the algorithm could potentially have use when in any areas where the data can be represented as shifted versions of an underlying factor. As a result, the algorithm also has potential applications in areas such as image processing.

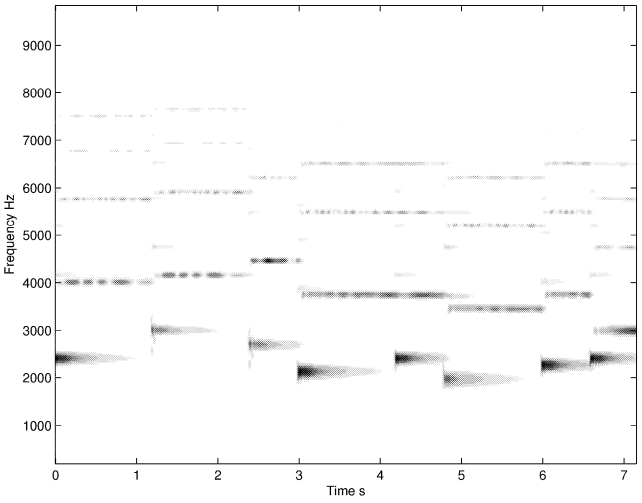

Fig. 4. Spectrogram of clarinet and piano signal

5. ACKNOWLEDGEMENTS

This research was supported by the Irish Research Council for Science, Engineering and Technology. Thanks also to David Flannery for his assistance with this research.

6. REFERENCES

[1] FitzGerald D (2004), Automatic Drum Transcription and Source Separation, Ph.D. Thesis, Dublin Institute of Technology, Dublin, Ireland.

[2] FitzGerald D, Coyle E and Lawlor B (2002) Sub-band Independent Subspace Analysis for Drum Transcription, 5th International Conference on Digital Audio Effects (DAFX02), pp. 65-69

[3] FitzGerald D, Coyle E and Lawlor B (2003) Prior Subspace Analysis for Drum Transcription, 114th AES Conference Amsterdam.

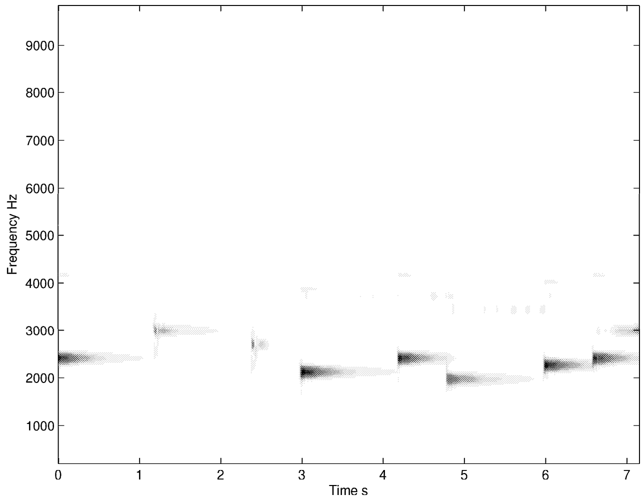

Fig. 5. Spectrogram of separated piano

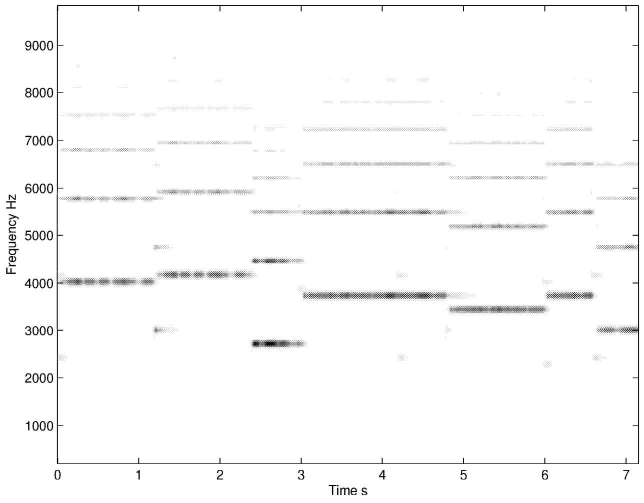

Fig. 6. Spectrogram of separated clarinet

[4] Brown, J.C., (1991), Calculation of a Constant Q spectral transform, Journal of the Acoustic Society of America, 90 60-66.

[5] Hoyer, P.O. (2002) Non-negative sparse coding Neural Networks for Signal Processing XII (Proc. IEEE Workshop on Neural Networks for Signal Processing), pp. 557-565, Martigny, Switzerland.

[6] Lee, D. and Seung H. (2001) Algorithms for non-negative matrix factorization. Adv. Neural Info. Proc. Syst. 13, 556-562.

[7] Casey M. and Westner A. (2000) Separation of Mixed Audio Sources By Independent Subspace Analysis in Proc. Of ICMC 2000, pp. 154-161, Berlin, Germany.

[8] Virtanen T, (2003) Sound Source Separation Using Sparse Coding with Temporal Continuity Objective, Proc. of International Computer Music Conference (ICMC2003), Singapore, 2003.

[9] S. A. Abdallah and M. D. Plumbley. Polyphonic transcription by non-negative sparse coding of power spectra. Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR 2004), Barcelona, Spain, October 10-14, 2004.

[10] Smaragdis, P., Brown, J.C., Non-negative Matrix Factorization for Polyphonic Music Transcription, IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), pp. 177-180, October 2003

[11] Vincent, E and Rodet, X., Music transcription with ISA and HMM. In Proc. ICA , 2004.

[12] B.W. Bader and T.G. Kolda, MATLAB Tensor Classes for Fast Algorithm Prototyping, Technical Report SAND2004-5187, Sandia National Laboratories, Livermore, California, Oct. 2004

[13] Tensor Classes for Matlab, available at http://csmr.ca.sandia.gov/ tgkolda/