2006-01-01

Shifted 2D Non-negative Tensor Factorisation

Derry Fitzgerald
Cork Institute of Technology

Matt Cranitch
Technological University Dublin, matt.cranitch@cit.ie

Eugene Coyle
Technological University Dublin, Eugene.Coyle@tudublin.ie

Shifted 2D Non-negative Tensor Factorisation

Derry FitzGerald, Matt Cranitch[ϕ] and Eugene Coyle*

[ϕ]Dept. of Electronic Engineering,
Cork Institute of Technology,
IRELAND
E-mail:[ϕ]derry.fitzgerald@cit.ie

** School of Control Systems and Electrical Engineering,*
Dublin Institute of Technology,
IRELAND
*E-mail: *eugene.coyle@dit.ie*

Recently, Non-negative Matrix Factor 2D Deconvolution was developed as a means of separating harmonic instruments from single channel mixtures. This technique uses a model which is convolutive in both time and frequency, and so can capture instruments which have both time-varying spectra and time-varying fundamental frequencies simultaneously. However, in many cases two or more channels are available, in which case it would be advantageous to have a multi-channel version of the algorithm. To this end, a shifted 2D Non-negative Tensor Factorisation algorithm is derived, which extends Non-negative Matrix Factor 2D Deconvolution to the multi-channel case. The use of this algorithm for multi-channel sound source separation of pitched instruments is demonstrated.

Keywords – Non-negative tensor factorisation, sound source separation

I INTRODUCTION

In recent years, matrix factorisation techniques such as non-negative matrix factorisation (NMF) [1] have been used to attempt single channel sound source separation [2]. These techniques attempt to approximate a magnitude spectrogram **X** as the product of low rank matrices **A** and **S**, i.e. $\mathbf{X} \approx \mathbf{AS}$. The columns of **A** contain frequency basis functions, while the associated rows of **S** contain corresponding amplitude envelopes for the frequency basis functions. Individual elements of **A** and **S** can then be used to attempt resynthesis of individual components or sources in the input data. Using a cost function which encourages sparseness in **A** and **S** results in a factorisation where the basis functions in **A** and **S** correspond to perceptually meaningful features, such as the frequency spectra of individual notes and their associated amplitude envelopes. Ensuring non-negativity is useful in obtaining the factorisation as a magnitude spectrogram is by definition non-negative, and it also reflects the intuition that sound sources add together.

However, these techniques are limited in that the decomposition is linear, and so each basis function pair typically corresponds to a single note played by a given pitched instrument. Therefore, to use these techniques for sound source separation, the basis functions must be grouped by instrument or source. While grouping techniques have been developed, it has proved difficult to obtain good clustering in many situations [3]. To overcome this problem shifted non-negative matrix factorisation was developed, which models notes played by an instrument as translations of a single instrument basis function [4]. This necessitates the use of a time-frequency resolution with log-frequency resolution, such as the Constant Q transform [5]. If the centre frequencies are set so that $f_i = f_{i-1}2^{1/12}$ where f_i is the centre frequency of band i, then the spacing between centre frequencies matches that of the even-tempered tuning system. Therefore, translating a frequency basis function of a note up by one bin is equivalent to a rise in pitch of one semitone.

A further problem with matrix factorisation techniques is that they return fixed spectra for each note, whereas the spectra of real instruments evolve over time. To this end, convolutive forms of NMF have been developed which model sources as a sequence of successive spectra and a corresponding amplitude envelope which is translated across time to activate each successive spectrum [6].

Recently, these methods have been combined in an attempt to model a source or pitched instrument as translations of successive spectra in both frequency and time, thereby allowing time-varying spectra and fundamental frequencies. This leads to a more realistic model of the sources present. This technique, called Non-negative Matrix Factor 2D Deconvolution (NM2D) has been used to separate mixtures of single channel instruments [7].

All of the above techniques work on single channel mixtures, however, most recordings of popular music from the past 40 years are stereo or two channel recordings. These two channel recordings are typically created by linear mixing of single channel recordings of individual instruments, with the only difference between each channel for a given instrument lying in the gain of the instrument in each channel. Therefore, the same model or set of basis functions can be used to describe a given instrument in either channel. Further, this gain difference is a source of extra information which can be used to aid the separation process. It can be seen then that extending NMF-based techniques to the multi-channel case would be advantageous, firstly due to the widespread use of 2 channel recordings, and secondly, as it provides extra information to aid the sound source separation process. To this end, techniques such as non-negative tensor factorisation (NTF) and shifted NTF have been proposed to deal with the multi-channel case [8],[9]. For sound source separation of multi-channel recordings, the NTF model can be written as:

$$\mathcal{X} \approx \hat{\mathcal{X}} = \sum_{k=1}^{K} \mathbf{G}_{:k} \circ \mathbf{A}_{:k} \circ \mathbf{S}_{:k} \qquad (1)$$

where $\mathcal{X}$ is a r x n x m tensor containing the spectrograms of the r channels, containing n frequency bins and m time frames. $\hat{\mathcal{X}}$ is an approximation to $\mathcal{X}$, $\mathbf{G}$ is a r x K matrix containing the gains of each factor in each channel, $\mathbf{A}$ is an n x K matrix containing the frequency basis functions, and $\mathbf{S}$ is an m x K matrix containing the amplitude basis functions, $\circ$ denotes outer product multiplication, and $:k$ denotes the kth column of a given matrix. The tensor factorisation is obtained by minimising the generalised Kullback-Liebler divergence between $\mathcal{X}$ and $\hat{\mathcal{X}}$. This is defined as:

$$D\left(\mathcal{X}\middle\|\hat{\mathcal{X}}\right) = \sum_{i,j,l} \mathcal{X} \log \frac{\mathcal{X}}{\hat{\mathcal{X}}} - \mathcal{X} + \hat{\mathcal{X}} \qquad (2)$$

where l, i, and j index over channel, frequency bin and time frame respectively.

For the rest of the paper, the following conventions, in line with those adapted by Bader and Kolda in [10] are used. Tensors are denoted by upper case letters such as $\mathcal{X}$, and contracted tensor product multiplication is defined as follows. If $\mathcal{W}$ is a tensor of size I_1 x ··· x I_N x J_1 x ··· x J_M and $\mathcal{Y}$ is a tensor of size I_1 x ··· x I_N x K_1 x ··· x K_P then contracted tensor multiplication along the first N modes is given as:

$$\begin{aligned}\left\langle \mathcal{W}\mathcal{Y} \right\rangle_{\{1:N,1:N\}}&\left(j_1,\ldots,j_M,k_1,\ldots,k_P\right) \\ &= \sum_{i_1=1}^{I_1} \cdots \sum_{i_N=1}^{I_N} \mathcal{W}\left(i_1,\ldots,i_N,j_1,\ldots,j_M\right) \qquad (3) \\ &\qquad \mathcal{Y}\left(i_1,\ldots,i_N,k_1,\ldots,k_P\right)\end{aligned}$$

where element indexing occurs within ( ) brackets, and where the modes to be multiplied are specified in the subscripts within the angle brackets.

II SHIFTED 2D NON-NEGATIVE TENSOR FACTORISATION

This research aims to extend NM2D to the multi-channel case. To do so requires the extension of the basic NTF model to allow for translations of the underlying sources in both frequency, and time. Where NM2D carries out translations by means of a shift operator, in line with our previous work, we carry out shifting by means of translation tensors. For an n x 1 vector, an n by n translation matrix is required. This translation matrix can be easily obtained by permuting the columns of the identity matrix. For example, in order to shift a vector by one position, the required matrix is $\mathbf{I}(:,[n, 1:n-1])$, where $\mathbf{I}$ is the identity matrix and the ordering of the columns is defined in the square brackets. For z translations, each of the z translation matrices can then be grouped into a translation tensor, $\mathcal{Q}$ of size n x z x n.

Separate translation tensors are defined to deal with shifts in frequency and across time respectively, resulting in the following model for shifted 2D Non-negative Tensor Factorisation (SNTF):

$$\hat{\mathcal{X}} = \sum_{k=1}^{K} \left\langle \mathcal{G}_{:k} \left\langle \left\langle \mathcal{T}\mathcal{A}_{:k} \right\rangle_{\{3,1\}} \left\langle \mathcal{S}_{:k}\mathcal{P} \right\rangle_{\{3,1\}} \right\rangle_{\{2:4,1:3\}} \right\rangle_{\{2.2\}} \qquad (4)$$

where $\hat{\mathcal{X}}$ is a tensor of size r x n x m which is an approximation to $\mathcal{X}$, and $\mathcal{G}$ is a tensor of size r x K, containing the gains of each of the k sources in each of the r channels. $\mathcal{T}$ is an n x z x n translation tensor, which translates the frequency basis functions in $\mathcal{A}$ up or down in frequency, thereby approximating different notes played by a given source. $\mathcal{A}$ is a tensor of size n x K x p, where p is the number of translations across time. $\mathcal{S}$ is a tensor of size z x K x m and $\mathcal{P}$ is a translation tensor of size m x p x m, which translates the amplitude envelopes contained in $\mathcal{S}$ across time, thereby allowing time-varying source spectra. For simplicity of notation, we adopt the convention that $:k$ denotes the tensor slice associated with the k^{th} source, with the singleton dimension included in the size of the slice.

$\mathcal{A}(:, d)_{:k}$ can then be interpreted as the frequency spectrum associated with the d^{th} translation in time of the k^{th} source. Similarly $\mathcal{S}(e , :)_{:k}$ can be interpreted as the amplitude envelope or time basis function associated with the e^{th} translation in frequency of the k^{th} source.

The required tensor factorisation is obtained again using Eqn. (2) as a cost function. By eliminating terms in $\mathcal{X}$ which are constant, this reduces to:

$$D\left(\mathcal{X}\middle\|\hat{\mathcal{X}}\right)=\sum_{i,j,l} -\mathcal{X}\log\hat{\mathcal{X}}+\hat{\mathcal{X}} \qquad (5)$$

Substituting for Eqn. (4) and taking the gradient with respect to $\mathcal{G}_{:k}$ yields the following update equation:

$$\mathcal{G}_{:k}=\mathcal{G}_{:k}+\lambda *$$

$$\left[\left\langle\left\langle\left\langle \mathcal{D}\mathcal{T}\right\rangle_{\{2,1\}}\mathcal{A}_{:k}\right\rangle_{\{4,1\}}\mathcal{S}_{:k}\right\rangle_{\{3:4,1:2\}}\mathcal{P}\right\rangle_{\{2:4,3:1\}} - \left\langle\left\langle\left\langle \mathcal{O}\mathcal{T}\right\rangle_{\{2,1\}}\mathcal{A}_{:k}\right\rangle_{\{4,1\}}\mathcal{S}_{:k}\right\rangle_{\{3:4,1:2\}}\mathcal{P}\right\rangle_{\{2:4,3:1\}}\right] \qquad (6)$$

where $\mathcal{D}=\mathcal{X}./\hat{\mathcal{X}}$, $\mathcal{O}$ is an all-ones tensor of size equal to $\mathcal{X}$, and ./ denotes elementwise division. This can be converted to a multiplicative update rule by setting λ equal to:

$$\lambda=\mathcal{G}_{:k}./\left\langle\left\langle\left\langle\left\langle \mathcal{O}\mathcal{T}\right\rangle_{\{2,1\}}\mathcal{A}_{:k}\right\rangle_{\{4,1\}}\mathcal{S}_{:k}\right\rangle_{\{3:4,1:2\}}\mathcal{P}\right\rangle_{\{2:4,3:1\}}$$

This yields:

$$\mathcal{G}_{:k}=\mathcal{G}_{:k}.*\left[\left\langle\left\langle\left\langle \mathcal{D}\mathcal{T}\right\rangle_{\{2,1\}}\mathcal{A}_{:k}\right\rangle_{\{4,1\}}\mathcal{S}_{:k}\right\rangle_{\{3:4,1:2\}}\mathcal{P}\right\rangle_{\{2:4,3:1\}}./\left\langle\left\langle\left\langle \mathcal{O}\mathcal{T}\right\rangle_{\{2,1\}}\mathcal{A}_{:k}\right\rangle_{\{4,1\}}\mathcal{S}_{:k}\right\rangle_{\{3:4,1:2\}}\mathcal{P}\right\rangle_{\{2:4,3:1\}}\right]$$

Similarly, update rules can be derived for $\mathcal{A}_{:k}$ and $\mathcal{S}_{:k}$. These are given by:

$$\mathcal{A}_{:k}=\mathcal{A}_{:k}.*\left[\left\langle\left\langle(\mathcal{G}_{:k}\circ\mathcal{T})\mathcal{D}\right\rangle_{\{[1,3],1:2\}}\left\langle\mathcal{S}_{:k}\mathcal{P}\right\rangle_{\{3,1\}}\right\rangle_{\{[1,2,4],[2,1,4]\}}./\left\langle\left\langle(\mathcal{G}_{:k}\circ\mathcal{T})\mathcal{O}\right\rangle_{\{[1,3],1:2\}}\left\langle\mathcal{S}_{:k}\mathcal{P}\right\rangle_{\{3,1\}}\right\rangle_{\{[1,2,4],[2,1,4]\}}\right]$$

$$\mathcal{S}_{:k}=\mathcal{S}_{:k}.*\left[\left\langle\left\langle\left\langle(\mathcal{G}_{:k}\circ\mathcal{T})\mathcal{A}_{:k}\right\rangle_{\{[2,5],[2,1]\}}\mathcal{D}\right\rangle_{\{1:2,1:2\}}\mathcal{P}\right\rangle_{\{2:3,[2,1]\}}./\left\langle\left\langle\left\langle(\mathcal{G}_{:k}\circ\mathcal{T})\mathcal{A}_{:k}\right\rangle_{\{[2,5],[2,1]\}}\mathcal{O}\right\rangle_{\{1:2,1:2\}}\mathcal{P}\right\rangle_{\{2:3,[2,1]\}}\right]$$

Once $\mathcal{G}$,$\mathcal{A}$, and $\mathcal{S}$ are randomly initialised to positive values, non-negativity of the factorisation is guaranteed through the use of multiplicative updates. The algorithm was implemented in Matlab, using the tensor classes for Matlab available at [11].

III SOUND SOURCE SEPARATION USING SNTF

To demonstrate the use of SNTF for the purposes of sound source separation, a 2 channel mixture of flute, piano and trumpet was made using sampled instruments. A separate sample was used for each note of each instrument to make the test as realistic as possible. The flute was panned to mid-left, the piano to the centre and the trumpet to mid-right. Figures 1 to 3 show the Constant Q spectrograms of the flute, piano and trumpet signals respectively, while Figures 4 and 5 show the left channel and right channel mixtures respectively.

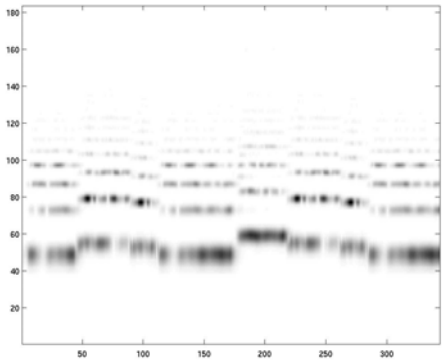

Figure 1: Spectrogram of flute signal

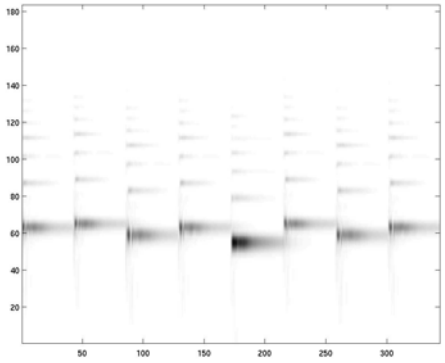

Figure 2: Spectrogram of piano signal

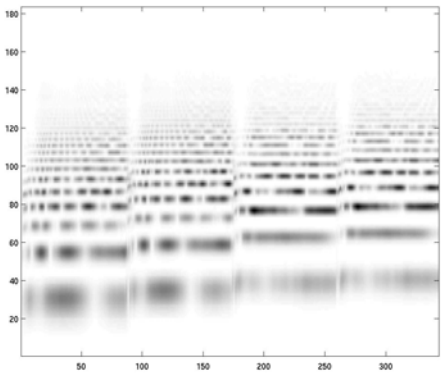

Figure 3: Spectrogram of trumpet signal

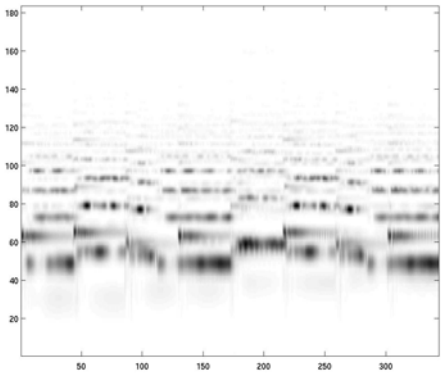

Figure 4: Left channel mixture spectrogram of flute, piano and trumpet.

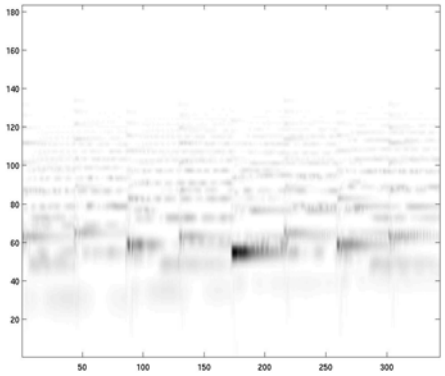

Figure 5: Right channel mixture spectrogram of flute, piano and trumpet.

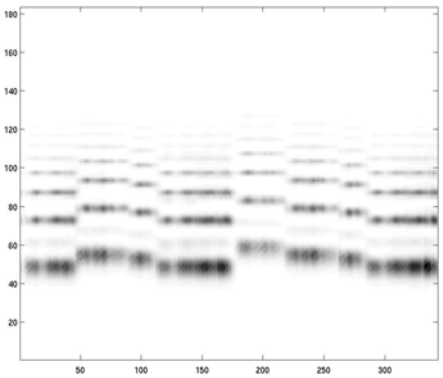

Figure 6: Separated spectrogram of flute.

Figures 6, 7 and 8 show the separated spectrograms of flute, piano and trumpet obtained by applying SNTF to the mixture spectrograms. The range of frequency translations was set to 7 frequency bins, corresponding to a pitch range of 7 semitones and the range of time translations set to 15 time frames, which corresponds to a maximum allowable shift in time of 0.18 seconds when using a hopsize of 512 samples at a sample rate of 44.1 kHz. The number of sources set to 3. Using these parameters, the SNTF algorithm succeeded in separating the instruments. This can be seen in the spectrograms of the separated instruments, with only the notes of each instrument visible in the separated spectrograms. The harmonic structure of the instruments has been successfully recovered, though some of the finer detail has been lost in the temporal evolution of the sources. This demonstrates that the SNTF algorithm is capable of separating mixtures of pitched instruments.

On listening to the separated waveforms, the separated instruments clearly predominate, with little or no traces of the other instruments to be heard in the separated signals. Indeed, the principal artefacts in the resynthesis is as a result of the approximate nature of mapping the log-frequency spectrograms back to linear-frequency spectrograms to allow for resynthesis. Methods of overcoming this problem are currently being researched [12]. A further shortcoming, inherent in all matrix factorisation and tensor factorisation algorithms, is that the algorithm is sensitive to the choices of shift ranges for both frequency and time, though we have observed that this is less of a problem for time shifts than for frequency shifts.

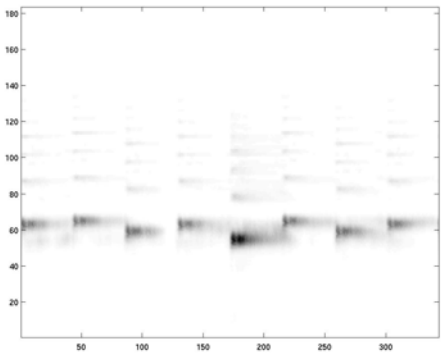

Figure 7: Separated spectrogram of piano.

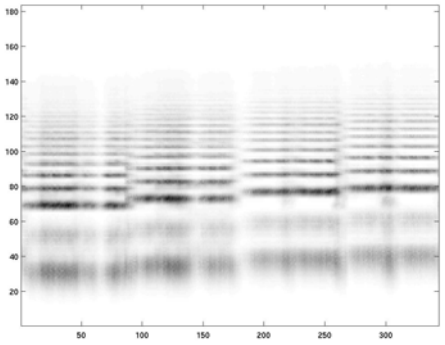

Figure 8: Separated spectrogram of trumpet.

IV CONCLUSIONS

A shifted 2D non-negative tensor factorisation algorithm has been derived, which extends non-negative matrix factor 2D deconvolution to the multi-channel case. The resultant algorithm is capable of modelling shifts in both time and frequency, thereby modelling sources as a group of successive spectra which can be shifted in frequency to approximate different notes played by the sources. The algorithm takes advantage of the spatial information available when dealing with multi-channel mixtures to improve source separation. The algorithm is demonstrated to be capable of separating simple mixtures of pitched instruments. Future work will concentrate on improving the separation capabilities of the algorithm by incorporating models of the instruments to be separated.

V ACKNOWLEDGEMENTS

This research was funded by the Irish Research Council for Science, Engineering and Technology.

VI REFERENCES

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

[2] P. Smaragdis, J.C. Brown, "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.

[3] D. FitzGerald, “Automatic Drum Transcription and Source Separation”, PhD. Thesis, Dublin Institute of Technology, 2004.

[4] D. FitzGerald, M. Cranitch, and E. Coyle, “Shifted Non-negative Matrix Factorisation for Sound Source Separation”, Proceedings of the IEEE Conference on Statistics in Signal Processing, Bordeaux, France, 2005.

[5] J. Brown, “Calculation of a Constant Q Spectral Transform”, J. Acoust. Soc. Am. 89 425-434, 1991

[6] T. Virtanen, “Separation of sound sources by convolutive sparse coding”, Proceedings of the ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Processing, 2004

[7] M. Schmidt and M. Morup, “Nonnegative Matrix Factor 2D Deconvolution for Blind Single Channel Source Separation”, Proceedings of the International Conference on Independent Component Analysis 2006.

[8] D. FitzGerald, M. Cranitch and E. Coyle, “Non-negative Tensor Factorisation for Sound Source Separation”, Proceedings of the Irish Signals and Systems Conference, Dublin 2005.

[9] D. FitzGerald, M. Cranitch and E. Coyle, “Sound Source Separation using Shifted Non-negative Tensor Factorisation”, Proceedings on the IEE Conference on Audio and Speech Signal Processing (ICASSP), Toulouse, 2006.

[10] B. Bader and T. Kolda, “MATLAB Tensor Classes for Fast Algorithm Prototyping”, Technical Report SAND2004-5187, Sandia National Laboratories, Livermore, California, 2004.

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

[12] D. FitzGerald, M. Cranitch, and M. Cychowski, “Towards an Inverse Constant Q Transform”, 120th AES Convention, Paris, 2006.