# Comparison of Performance in Binaural Sound Source Localisation using Convolutional Neural Networks for differing Feature Representations

Jago T. Reed-Jones[1], Karl O. Jones[1], Paul Fergus[1], John Marsland[1], and David L. Ellis[1]

[1]*Liverpool John Moores University*

Correspondence should be addressed to Jago T. Reed-Jones (j.t.reedjones@2019.ljmu.ac.uk)

**ABSTRACT**

Binaural Sound Source Localisation is increasingly being achieved by means of the Convolutional Neural Network (CNN). These networks take in a Time-Frequency representation of audio as an input, and use this to estimate the direction of arrival of a sound. In previous works, different Time-Frequency representations have been used, but never only using solely magnitude spectra, leading to a lack of understanding in the importance of this in full azimuthal binaural sound source localisation.

This work aims to address that gap by testing the performance of a CNN trained and tested on four different Time-Frequency representations: Mel-Spectrogram, Gammatonegram, Mel-Frequency Cepstrum, and Gammatone-Frequency Cepstrum.

From this test, it was found that Spectrograms are suitable for the task of full azimuthal binaural sound source localisation.

## 1 Introduction

Binaural Sound Source Localisation (BSSL) refers to the estimation of the direction of arrival (DOA) of a sound from the measurement of a sound field using a binaural array, either through estimation of any or all of either spherical or Cartesian coordinates. This paper focuses solely on azimuth estimation.

In conventional sound source localisation (SSL) techniques measurement of the time difference of arrival between the sensors is often relied upon, however in the case of only two sensors this leads to multiple solutions for one measurement leading to front-back confusion in the azimuthal plane. To overcome this BSSL utilises not just binaural cues, comparisons of phase and level differences between the two ears, but also monaural cues; comparisons of differences in level and phase and different frequencies dependent on the DOA.

These monaural cues are caused by the unique transfer functions at different DOAs of a binaural array. The different transfer functions are caused by the filtering effects of mostly the pinna, but to some extent the head and torso also. These are referred to as the head related transfer functions (HRTFs).

The aforementioned binaural array refers not just to a microphone array with two points, but must also inflect DOA-dependent filtering on the acoustic wave, such as a Head and Torso Simulator (HATS) which places microphones in the ear canals of a mannequin head and torso.

As the binaural cues are understood to be prominent in biological localisation, it is unsurprising that in machine sound localisation comparison of time-delay and level differences are also used [1, 2]. Use of binaural cues alone, however, leads to ambiguity when trying to localise in the full azimuthal plane due to the possibility of front-back confusions, and so many of these works are restricted to the front hemi-field.

To combat this, spectral cues must also be interpreted. This may be possible through hand-crafted features, but Convolutional Neural Networks (CNNs) have also been employed for this task [3, 4], as these networks' ability to craft feature maps from audio spectra allows for the ability to interpret these cues.

This paper investigates the CNN's ability to extract these cues for localise in the full azimuthal plane, only using magnitude representations of sound. This is an unconventional approach, and other uses of CNN have combined spectral cues with phase or time delay related features [5, 6, 7], but by doing as such the aim is uncover the utility of using time-frequnecy spectra in full azimuthal BSSL.

To this end, four different time-frequency representations of sounds are investigated. Two spectra: Mel-Spectrograms and Gammatonegrams, and two Cepstrums: the Mel-Frequency Cepstrum (MFC) and Gammatone Frequency Cepstrum (GFC).

## 2 Audio Datasets

### 2.1 Audio Training Data

Prior to processing, a dataset of binaural recordings was created through through the convolution of Binaural Room Impulse Responses (BRIRs) with speech.
Speech samples were taken from the Librispeech corpus [8], a collection of English speech sampled at 16kHz. A total of 2000 samples were created by cutting 200 different audio files from the corpus into 10 100mS files.
BRIRs are head related impulse responses but in a diffuse field. These were simulated using an implementation of the image-source method [9], using HRTFs of a KEMAR head taken from the CIPIC Dataset [10].
The BRIRs represented the azimuthal positions available in the original HRTF Dataset, consisting of sources spaced by $5^\circ$ in the ranges $-45^\circ \leq \varphi \leq 45^\circ$ on a $360^\circ$ scale where $0^\circ$ is facing forward, with fewer sources in the remaining range. This can be seen in Figure 1.

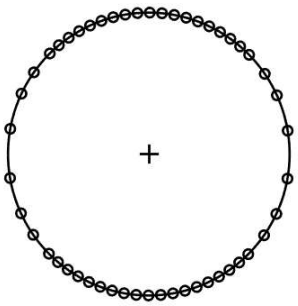

**Fig. 1:** Source directions available in dataset

This is represented as:

$$x_{L,R}[n,\varphi_T] = s[n] * BRIR_{L,R}[n,\varphi_T] \tag{1}$$

Where $s[n]$ is the speech sample, and $BRIR[n,\varphi_T]$ is the BRIR for the target azimuth $_T$, and $L,R$ refer to the left an right channels.

The BRIRs were simulated for a variety of rooms, altering in dimensions, as well as in level of reverberation. Five different sets of room dimensions were used, where each dimension was randomly generated on a scale of 1-10m. For each of the five room dimensions, the absorption coefficients of the boundaries were altered so as to yield target reverb times of:

$$RT_{60} = [0.5, 1, 1.5] seconds \tag{2}$$

The BRIR of a different room was used for each unique audio file from the speech corpus, therefore each 10 speech samples, with a uniform distribution. Additionally, the original HRTFs were used as to represent $RT_{60} = 0$. In this case there are no room dimensions for the files to be distributed over, but the total number of files representing $RT_{60}$ was the same as at other reverb times.
Additionally, noise was added to the audio files. The noise was made up of a mixture of pink noise sources combined with BRIRs of a random azimuth. For each audio file from the corpus, a random number of sources between 1-10 was used, and for each source pink noise would be convolved with a BRIR from the same room as used for the speech for that audio file, but a random azimuth. Each source was scaled by a random amplitude, but the entire mixture was scaled to yield the following Signal-to-Noise Ratios (SNRs):

$$dB(SNR) = [0, 12, 24, 36] \tag{3}$$

This approach to creating noise was done to more closely simulate a realistic noise environment, rather than the entirely diffuse nature that pink noise with no HRTF applied own would have, which is less challenging for BSSL systems. The noise mixture $m[n]$ is described then as:

$$\begin{gathered} m_{L,R}[n] = A \sum_{n=1}^{N} (A_R . p[n] * BRIR_{L,R}[n,\varphi_R]) \\ A = (1/A_{SNR}) . RMS\{s[n]\} \end{gathered} \tag{4}$$

where $p[n]$ is pink noise, $A_R$ is a random scalar between 0 and $1/N$, $\varphi_R$ is a random azimuth, and $A_{SNR}$ represents the magnitude conversion of the target SNR levels in Equation 3. The final audio file is the sum of the source and noise mixture:

$$y_{L,R}[n,\varphi_T] = x_{L,R}[n,\varphi_T] + m_{L,R}[n] \tag{5}$$

The final distribution of the audio training files can be seen in Table 1. The total number of files was then 32,000 per source direction: 1,600,000 in total.

| | Signal to Noise Ratio (dB) | | | |
|---|---|---|---|---|
| RT60 (S) | 0 | 12 | 24 | 36 |
| 0 | 6.25% | 6.25% | 6.25% | 6.25% |
| 0.5 | 6.25% | 6.25% | 6.25% | 6.25% |
| 1 | 6.25% | 6.25% | 6.25% | 6.25% |
| 1.5 | 6.25% | 6.25% | 6.25% | 6.25% |

**Table 1:** Distribution of training files per each source direction

### 2.2 Audio Testing Data

The method of preparing the testing data was similar to that of the training data, however with some changes made so that the degree to which the systems are generalising is better tested.
Different audio files from the same corpus were used, with one 10mS sample being taken from 100 audio files in the corpus for this purpose.
While the same target reverb times were used, new BRIRs for 10 new unique room dimensions were used instead. This is as at each azimuth the BRIR has unique reverberation, which can be learned by the system to identify the source direction rather than the HRTF.
The noise source was changed from pink noise to a random sample of room noise.
All other factors remained consistent, and as such the distribution between noise levels and reverberation seen in Table 1 is also applied.

## 3 Pre-Processing Methods

### 3.1 Mel-Spectrogram

The first method is creation of a Mel-Spectrogram. This refers to a Spectrogram create through the STFT, with mel-weighting applied. This is achieved by windowing the audio file, and applying a Discrete Fourier Transform to each window:

$$Y_{L,R}[k,\varphi_T] = \sum_{n=1}^{N} y_{L,R}[n,\varphi_T]w[n]e^{-j\omega kn/N} \quad (6)$$

where $w[n]$ is the window. In this case, the signal was windowed so that the window had a length of 465 samples, and the overlap length was 256 samples, leading to 6 windows.
Following this, the magnitude of each of the transformations is multiplied with the frequency response of a filter from a mel-filterbank.

$$\begin{bmatrix} M_1 \\ M_2 \\ \dots \\ M_N \end{bmatrix} = \sum \left( |Y_{L,R}[k,\varphi_T]|. \begin{bmatrix} MEL_1[k] \\ MEL_2[k] \\ \dots \\ MEL_N[k] \end{bmatrix} \right) \quad (7)$$

The sum of each of these resulting frequency domain representations, $G[\omega]$, then represents the energy in this band.
The mel-filterbank is made up of 300 triangular bandpass filters, scaled according to mel frequency rating in a range of 100Hz to 8Khz. This number is larger than typical in applications of CNN on audio, this is as full DoA estimation from spectra is a difficult task, and so smaller numbers of frequency bands leads to poor performance which is difficult to interpret.
The final product is a matrix of values, where one dimension with a length of 6 represents samples, and one dimension with a length of 300 represents frequency, and third dimension of 2 represents the left and right channels.

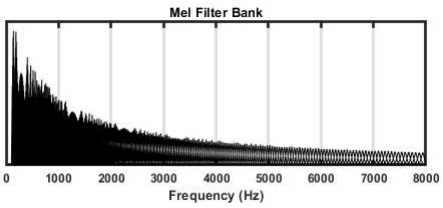


**Fig. 2:** Mel-Filterbank Frequency Curves

### 3.2 Gammatonegram

The Gammatonegram is similar to the Mel-Spectrogram, but differs in the filter bank used, instead using a gammatone filterbank. Accordingly, the same process of applying a DFT to windowed signals is applied, using the same window.
The Gammatonefilter is designed to more closely represent the cochlea. While the frequency scaling of the Mel-Spectrogram is already consistent with human perception, the Gammatone filter differs in that a much larger passband is represented in each filter, which also leads to significant crossover between each filter band, as seen in Figure 3.

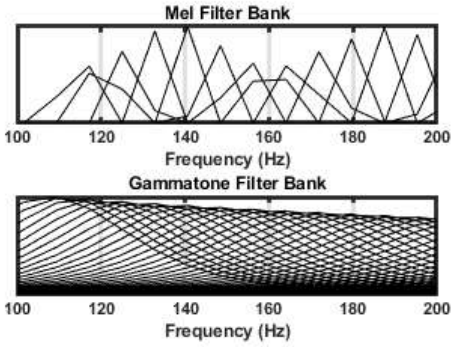


**Fig. 3:** Comparison of Frequency Curves of both Filterbanks

A filterbank was created with the same range of 100Hz-8kHz, with the same number of bands being 300. These filters were combined with the frequency domain audio in the same manner as for the mel-spectrogram.

$$\begin{bmatrix} G_1 \\ G_2 \\ \dots \\ G_N \end{bmatrix} = \sum \left( |Y_{L,R}[k,\varphi_T]|. \begin{bmatrix} GAM_1[k] \\ GAM_2[k] \\ \dots \\ GAM_N[k] \end{bmatrix} \right) \quad (8)$$

This again resulted in a matrix of the dimensions $[300,6,2]$ representing frequency, time and channel accordingly.

### 3.3 Mel-Frequency Cepstrum

The MFC, made of Mel-Frequency Cepstral Coefficients (MFCCs), use the Mel-Spectrogram but applies a Discrete Cosine Transform (DCT) to the spectrum, to create a cepstrum.
This was done by taking the log of each of the energies calculated in Eq. 7. For each window of values, the DCT was then taken yielding a new set of 300 values.

$$\begin{bmatrix} MFCC_1 \\ MFCC_2 \\ \dots \\ MFCC_N \end{bmatrix} = DCT \left\{ \begin{bmatrix} \log(M_1) \\ \log(M_2) \\ \dots \\ \log(M_N) \end{bmatrix} \right\} \quad (9)$$

This is done for each instance in timem and channel, again leading to a matrix of the dimensions $[300,6,2]$

### 3.4 Gammatone-Frequency Cepstrum

The GFC, made of Gammatone-Frequency Cepstral Coefficients (GFCCs), was created using the same procedure, but using the energies calculated using the Gammatonefilter bank in Eq. 8.

$$\begin{bmatrix} GFCC_1 \\ GFCC_2 \\ \dots \\ GFCC_N \end{bmatrix} = DCT \left\{ \begin{bmatrix} \log(G_1) \\ \log(G_2) \\ \dots \\ \log(G_N) \end{bmatrix} \right\} \quad (10)$$

## 4 CNN & Training

The design of a 3 layer convolutional neural network was used in all cases. The network features layers with filters of increasing size and number, as this was found to be suitable for this task in preliminary tests.

| | |
|---|---|
| Input Layer | [300, 6, 2] |
| Convolution Layer | ([2,2],8) |
| Batch Normalisation | |
| ReLu | |
| Max Pooling | (2,2) |
| Convolution Layer | ([8,8],16) |
| Batch Normalisation | |
| ReLu | |
| Max Pooling | (2,2) |
| Convolution Layer | ([16,16],32) |
| Batch Normalisation | |
| ReLu | |
| Dense Layer | [72] |

**Table 2:** CNN Architecture

Stochastic gradient descent training was employed for with an a learn rate of 0.001. A maximum number of 200 epochs was set, but training would be manually ended if required to prevent overfitting.

## 5 Results

### 5.1 Metrics

#### Classification Accuracy

The first metric recorded is the rate at which the system succesfully classified the testing set, so that $\varphi_P = \varphi_T$ where $\varphi_P$ is the predicted azimuth, and $\varphi_T$ is the true azimuth.

| | Classification Rate | RMSE | Front-Back Confusion Rate |
|---|---|---|---|
| **Mel-Spectrogram** | 40.4% | 62.9 | 2.5% |
| **Gammatonegram** | 38.5% | 63.7 | 2.8% |
| **MFC** | 20.5% | 78.8 | 5.3% |
| **GFC** | 19.3% | 79.2 | 5.7% |

**Table 3:** Overall performance metrics for each system

#### RMSE

The second metric recorded is the Root Mean Square Error (RMSE)

$$RMSE = \sqrt{\sum_{i=1}^{N} \frac{(\varphi_{P,i} - \varphi_{T,i})^2}{N}} \tag{11}$$

#### Front-Back Confusion Rate

The last metric is the confusion rate, these refers to the rate at which back-to-front confusions occur, which are a when the system mis-classifies a source as occurring from the mirror position of the true azimuth. This can occur frequently as when this condition is met the binaural cues are identical.
This is calculated as being the rate at which the system classifies to within $\pm 10^{\circ}$ of the mirror of $\varphi_T$, except for the cases where $\varphi_T$ is within $10\circ$ of it's own mirror position, for example at $80^{\circ}$.

These metrics for each system can be found in Table 3.

### 5.2 Response to Changing Reverberation SNR

Finally, it is also shown how the system responds with respect to increasing reverberation. This is shown with the classification accuracy being plotted against an x-axis of reverb time, and a line of best fit was created through polynomial regression, as seen in Fig 4.

The same approach is adopted to plot the response of the system to an increasing Signal-to-Noise Ratio, as seen in Fig 5

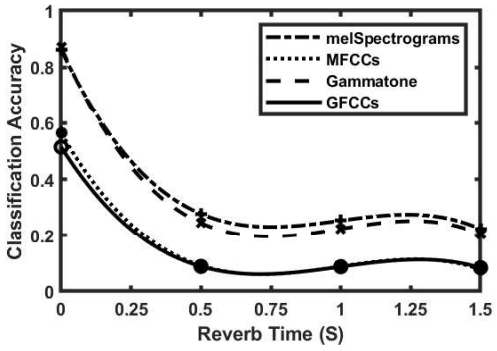


**Fig. 4:** Classification Accuracy with respect to Reverberation for all systems

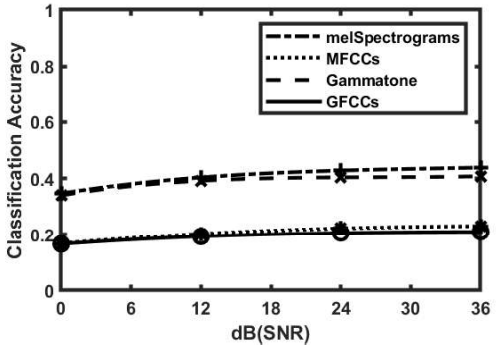


**Fig. 5:** Classification Accuracy with respect to SNR for all systems

## 6 Discussion

In all cases, a low level of performance can be seen. This is as this system is particularly vulnerable to overfit, specifically overfitting to BRIRs on which it has been trained, and so performing poorly on unknown BRIRs. Proof of this can be seen in Figure 4, where the classification accuracy is very high when $RT_{60} = 0$ but low at other times, this is as this is only time the test-

ing and training datasets have used the same impulse response.

This performance can be improved by increasing the number of rooms used in the collection of BRIRs used in the training dataset, however doing this also requires increasing the total number of files or performance generally decreases, and due to limited computing resources this was not possible. As the aim of this work is only to compare the methods, this was nonetheless deemed sufficient.
From Table 3 the same pattern can be seen, Gammatonegram and Mel-Spectrogram outperform the GFC and MFC. There is also a very slight performance increase from the mel frequency decomposition, however this is small enough to be rendered statistically insignificant.
Back-to-front confusions are not a significant challenge for any of the systems, the errors that do occur are generally not correlated to the true azimuth, however it can be seen that the level of confusions which occur do rise above what naturally occur in a random distribution for the two cepstra, whereas they do not for the two spectrograms.
With regard to changing the SNR, it can be seen that even at very low SNRs the performance is not significantly altered. From the perspective of this test, it is impossible to determine if any are more robust to noise as all seem to alter at approximately the same rate.

## 7 Conclusion

From the results it has been seen that spectral cues can enable full azimuthal binaural localisation, however doing such with CNN can lead to issues with overfit. To tackle this, care must be taken in creation of training datasets which do not under-represent the range of possible acoustic conditions.
Spectrograms are more suitable than Cepstrums for this task, and the method of filtering the audio into frequency bands is not found to be of significance.

## References

[1] May, T., van de Par, S., and Kohlrausch, A., "A Probabilistic Model for Robust Localization Based on a Binaural Auditory Front-End," *IEEE Transactions on Audio, Speech, and Language Processing*, 19(1), pp. 1–13, 2011, doi:10.1109/TASL.2010.2042128.

[2] Ma, N., May, T., and Brown, G. J., "Exploiting Deep Neural Networks and Head Movements for Robust Binaural Localization of Multiple Sources in Reverberant Environments," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(12), pp. 2444–2453, 2017, doi:10.1109/TASLP.2017.2750760.

[3] Yang, Y., Xi, J., Zhang, W., and Zhang, L., "Full-Sphere Binaural Sound Source Localization Using Multi-task Neural Network," in *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 432–436, 2020.

[4] Jiang, S., Wu, L., Yuan, P., Sun, Y., and Liu, H., "Deep and CNN fusion method for binaural sound source localisation," *The Journal of Engineering*, 2020(13), pp. 511–516, 2020, doi:https://doi.org/10.1049/joe.2019.1207.

[5] Vecchiotti, P., Ma, N., Squartini, S., and Brown, G. J., "End-to-end Binaural Sound Localisation from the Raw Waveform," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 451–455, 2019, doi:10.1109/ICASSP.2019.8683732.

[6] Xu, Y., Afshar, S., Singh, R. K., Wang, R., van Schaik, A., and Hamilton, T. J., "A Binaural Sound Localization System using Deep Convolutional Neural Networks," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, 2019, doi:10.1109/ISCAS.2019.8702345.

[7] Pang, C., Liu, H., and Li, X., "Multitask Learning of Time-Frequency CNN for Sound Source Localization," *IEEE Access*, 7, pp. 40725–40737, 2019, doi:10.1109/ACCESS.2019.2905617.

[8] Panayotov, V., Chen, G., Povey, D., and Khudanpur, S., "Librispeech: An ASR corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 2015, doi:10.1109/ICASSP.2015.7178964.

[9] Mandel, M., "Matlab Recti-Linear Room Simulator," https://github.com/mim/rlrs, 2013.

[10] Algazi, V., Duda, R., Thompson, D., and Avendano, C., "The CIPIC HRTF database," in *Proceedings of the 2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics (Cat. No.01TH8575)*, pp. 99–102, 2001, doi: 10.1109/ASPAA.2001.969552.