AudioConv

- A new metric for blind source separation -

Masters Thesis

Daniel Woodward

Aalborg University
Electronics and IT

Here you can write something about which tools and software you have used for typesetting the document, running simulations and creating figures. If you do not know what to write, either leave this page blank or have a look at the colophon in some of your books.

Architecture, Design and Media Technology
Aalborg University
http://www.aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Title:
AudioConv: A new metric for blind source separation

Theme:
Master's Thesis
Machine Learning

Project Period:
Autumn Semester 2022

Project Group:
XXX

Participant(s):
Daniel Woodward

Supervisor(s):
Cumhur Erkut

Copies: 1

Page Numbers: 35

Date of Completion:
December 21, 2022

Abstract:

Blind Source Separation for musical signals is an active research area and is currently evaluation often using only the Signal to Distortion Ratio. However, he metric has been critised in literature for not correlating with listeners rating scores for separation models. Hence, this thesis aims to document the creation of a new metric called audioConv, a deep learning perceptually inspired metric. AudioConv is Convolutional Neural Network using additional features based upon well established models. Analysis of audioConv is through the correlation with listener rating score and shows that while there is a potential, further work is needed improve the metric. The work focuses on machine learning techniques for audio and the need for quality data for these models.

Contents

Chapter 1

Introduction

The evaluation of similar of musical signals is important in several fields. For example, in on going research fields such as musical source separation, copyright music analysis and musical identification algorithms. The aim is identifying from two musical signals how well one is said to resemble the other. In other words, if a person listens to two musical signals, how would they rank the likeness of one signal to another. Improving a similarity metric has many benefits, in an ideal case it would eliminate the use of listening tests for assessing the quality of signal altering algorithms against a reference signal. It would also improve the reproducability and consistency of results across these areas. It is certainly possible to make significant statistical claims using listening tests but there is an opportunity to reduce the introduction of biases in these tests and the associated time and cost of listening tests are often prohibitive. This thesis will focus on the development of a tool for replacing listening tests for musical source separation.

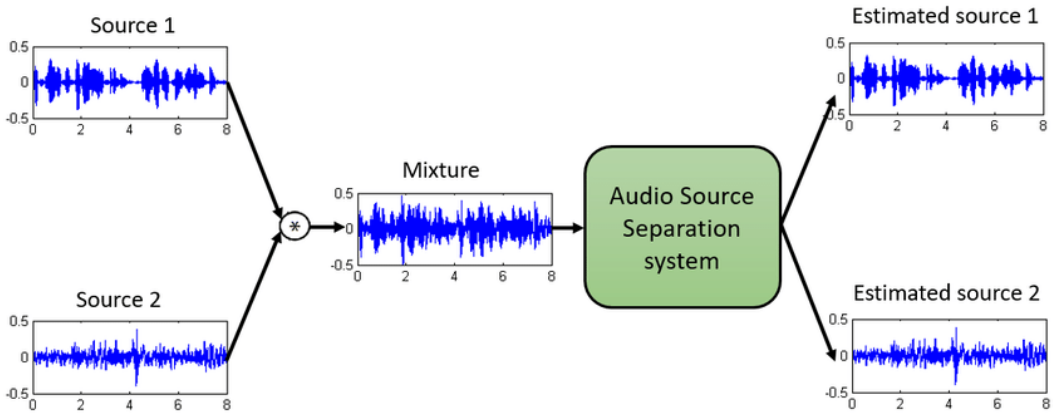

Figure 1.1: Source Separation Example

Firstly to understand the replacement for listening tests, source separation for musical signals must be understood. It is described as separating a mixed signal into it's constituent parts. For example, applied to a pop song, source separation algorithms may render two signals such as the singing and accompaniment similar to to Fig 1.1. Here the sources 1 and 2 refer to the vocals and accompaniment of the pop song, the mixture is the combined audio signals the source separation system aims to separate this mixture back into it's constituent parts. There exist other algorithms that separate signals in several signals too, and as such would separate a pop song into singing, drums, guitar, other. The interest in source separation lies in it's many uses. Music remastering relies upon the original separated audio signals (also called stems) which are recombined through the use modern music software. However, if these stems are missed, source separation provides an opportunity to remaster material that previously would have proved difficult. This has been shown by Mark Linett in his remastering of Beach Boys albums [1].

The development of source separation algorithms is an active research area. The evaluation of these algorithms is not standardised but there are several common methods as discussed in chapter 2. There is significant criticism of these metrics, however they form the basis of evaluation for literature in the research area [13]. In an effort to address the issues brought up by Cano et al [5], a perceptually motivated metric for musical similarity is proposed to model human hearing. The advent of new machine learning techniques, access to more data than ever before and increase in computation power has drastically improved the results neural networks in recent years and comparisons between neural networks and how a human brain works motivates the strong use of such a machine learning technique. By correlating the results of our metric with several datasets the aim of this masters thesis is to show this new metric can accurately predict a similarity score as given by a human listener.

[1] https://www.soundonsound.com/people/mark-linett-remixing-beach-boys

Chapter 2

Literature Review

2.1 Blind Source Separation

Source Separation has it's origins in the 'cocktail party effect' whereby individuals are able to distinguish different conversations when multiple people are speaking at the same time and what it would take to build a machine that would accomplish this task [6]. In 1990 Bregman introduced Auditory Scene Analysis (ASA), a model for how the brain understands its surroundings through sound [2]. I.e a model for how humans can decompose mixed audio signals into the constituent components of that mixture. ASA was then extended to Computational models in an attempt to emulate the results of signal processing similar to the brain [3]. These signals do not have to be people in conversation however. One major application is in Music Information Retrieval where a musical signal such as a song is deconstructed into the singing voice and accompaniment tracks. The first literature on blind source separation comes from Jeanny Herault and Christian Jutten's H-J algorithm for separating two independent source signals [41]. This opened a new research area and several categories of techniques have since been used when approaching the problem of source separation.

2.2 State of the art in Source Separation

2.2.1 Machine Learning

Machine learning is the set of techniques that result in computer programs that learn to imitate intelligent behavior. These techniques include but are not limited to K-means clustering, logistic regression, decision trees and deep neural networks (DNNs). DNNs have become extremely powerful and prominent in recent years for several reasons. They directly benefit from the increase in available computing power, the quality and quantity of data available and have shown to have applica-

tions in many research areas such as healthcare, education, chip design and in this case music information retrieval [42]. In blind source separation machine learning, and more specifically DNNs represents the state of the art [9]. Defossez et al shows a hybrid demucs model which includes both spectral and waveform domain Unet models as in figure 2.1. It builds upon the previous work by combining popular techniques in both domains.

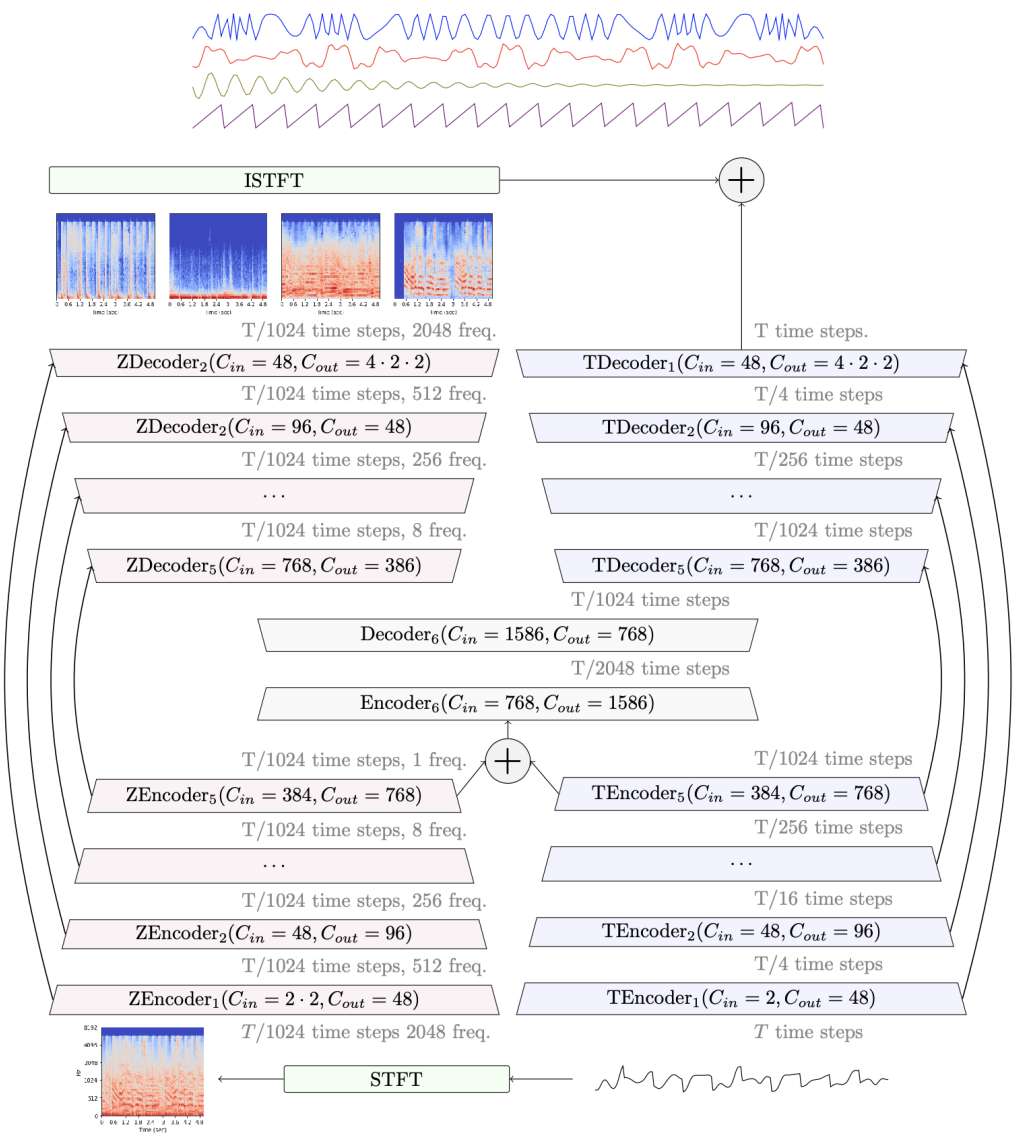

Figure 2.1: Hybrid Spectrogram and Waveform Source Separation Architecture [9]

2.2.2 Waveform Domain

Wave Domain DNN models have only been explored recently with a Wavenet for speech denoising model [26], Conv-Tasnet [21] (both adapted for musical source separation) and Wave-U-Net being presented by Lluis and Pons [20]. There are several advantages to source separation in the waveform domain. In spectral models, the phase information is often discarded when the data is represented as power or magnitude spectrograms. As a result, there is a loss of information and may cause phase related problems. In contrast, waveform domain models do not have this issue. Other machine learning techniques used for source separation involving matrix factorisation may benefit from a non-negative constraint which are easily implemented with magnitude and power spectrograms (such as NMF). Waveforms are typically in the range of [-1,1]. [20]. Demucs when first published showed a big improvement in source separation in the waveform domain and at the time showed a higher signal to distortion ratio on the MUSDB dataset than any other published work including spectral based models [10] [25].

2.2.3 Spectral Domain

The Spectral Domain has been more thoroughly explored for Source Separation of musical signals. It is a representation of the variation of the frequencies within the signal over time as shown in figure 2.2. The power or magnitude spectrogram is a very common transform before separation. The Signal Separation Evaluation Campaign (SiSEC) 2015 proved a significant improvement through the use of a neural networks for source separation [22]. The model which used a spectrogram input presents a relatively simple feed forward neural network [35]. SiSEC 2016 continues this trend with an improvement blending the previous winner with a recurrent neural network and data augmentation [19] [36]. The best model of SiSEC in 2018 was a Deep Neural Network using multiscale multiband densenet with a spectrogram as an input again [33] [34]. The vast majority of the best submissions from the SiSEC literature show how ubiquitous spectrograms and the spectral domain are in source separation. Hybrid Demucs [9] partly uses as Unet [27] architecture with a spectrogram as an input. There have been different techniques for the ouput including a mask of the input, modulation [16] or a Complex as channels (where the input real and complex parts are concatenated) spectrogram.

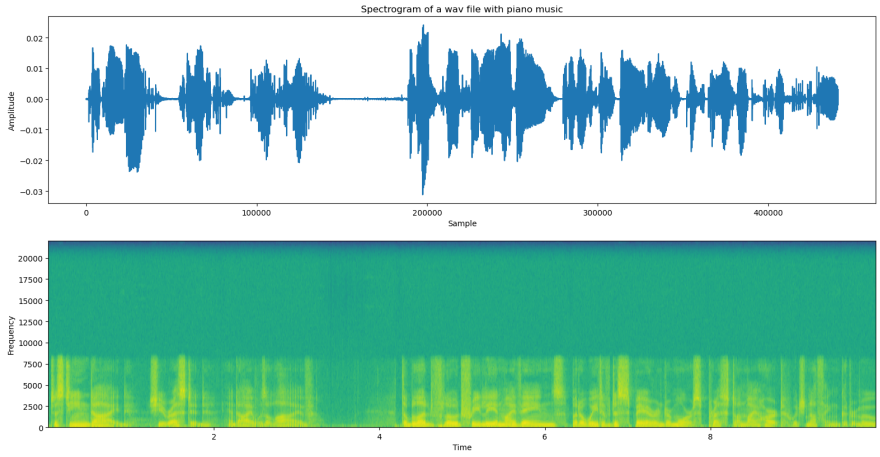

Figure 2.2: Spectrogram Example from the SASSEC dataset [39]

2.3 Evaluation

Typically Source separation metrics rely upon a reference source signal from an original mixed source such as the Signal Distortion Ratio [37]. This is then compared to an algorithms resulting estimated signal. This workflow is shown in figure 2.3. For musical source separation, the original mixture is often a song and the estimated sources are the vocals and accompaniment. The metric scores are then calculated with respect to reference signal I.e. isolated vocals and accompaniment which then informs the quality of the source separation.

Some metrics such as PEASS IPS scores [11] also require the original mixed signal as shown in the metric fig 2.4

2.4 BSS EVAL

In source separation the aim is to decompose the mixed source into its constituents. There is a target signal ($S_{target}(t)$) which we aim to produce but in reality it is unlikely a method will work perfectly. The estimated target signal ($\hat{S}_{target}(t)$) can be said to approximate S_{target} but with some additional associated error $e_{tot}(t)$. $e_{tot}(t)$ can be further decomposed into three error types:

- $s_{interf}(t)$: Interference from unwanted sources. E.g unwanted audio from the original source which was not removed.

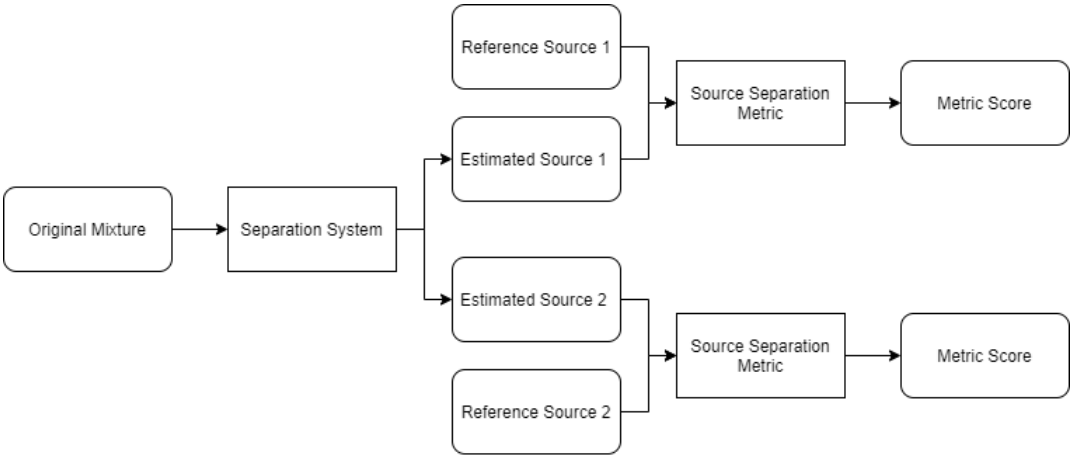

Figure 2.3: Typical Metric Workflow

- $s_{noise}(t)$: Perturbation noise not from the source. E.g. noise that is contained in the source pre-separation.
- $s_{artif}(t)$: Artifacts introduced by the separation algorithm. E.g. if any values in the padded section of the sample are non-zero.

2.4.1 Signal Distortion Ratio

The signal to noise ratio (SDR) is an almost ubiquitous metric for source separation. It's described by equation 2.1.

$$SDR = \frac{\left\|s_{target}(t)\right\|^2}{\left\|s_{interf}(t) + s_{noise}(t) + s_{artif}(t)\right\|^2} \tag{2.1}$$

The signal distortion ratio is a measure of the quality of the final output relative to the unwanted sources of distortion in the output. For calculation the numerator in equation 2.1 is the ground truth of the model, the samples which we aim to recover. The denominator in equation the difference between the ground truth (the target) and the neural network output.

2.4.2 Signal Interference Ratio

The Signal Interference Ratio (SIR) is a measure of the signal to interference from unwanted signals. As seen in equation 2.4 it is similar to SDR but omits the artifacts and noise associated with the SDR. It can help give a better idea of the leakage of other sources into the estimated signal relative to the reference signal.

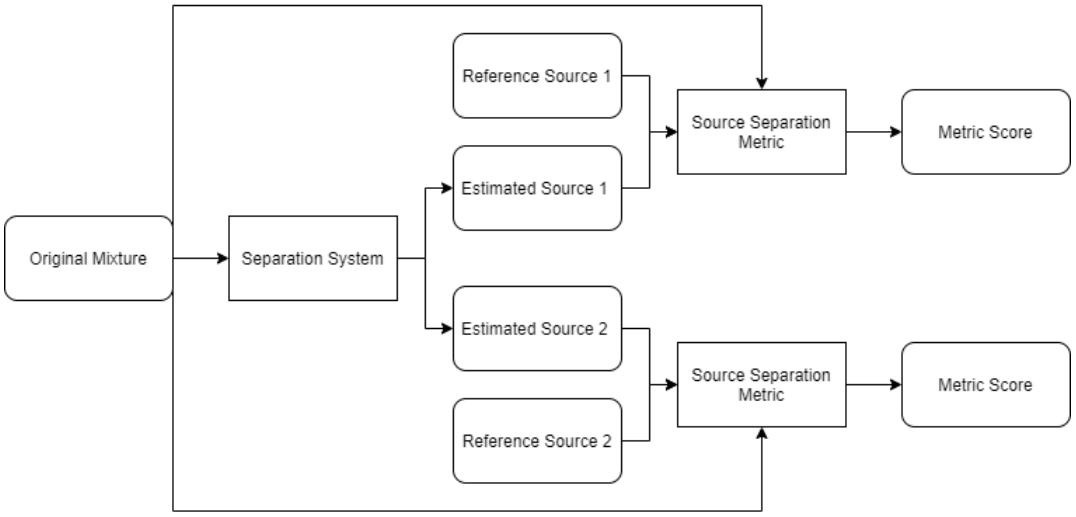

Figure 2.4: Extended Metric Workflow

$$SIR = \frac{\left\|s_{target}(t)\right\|^2}{\left\|s_{interf}(t)\right\|^2} \tag{2.2}$$

2.4.3 Signal Artifact Ratio

The Signal Artifact Ratio (SAR) is a measure of the ratio of the estimated signal to the artifacts found in that signal. I.e artifacts caused by the separation method. Often these artifacts are reduced in recent algorithms by making the initial reference signal a linear combination of the estimated separated signals.

$$SIR = \frac{\left\|s_{target}(t) + s_{interf}(t) + s_{noise}(t)\right\|^2}{\left\|s_{artif}(t)\right\|^2} \tag{2.3}$$

2.4.4 Signal Noise Ratio

The Signal Artifact Ratio (SNR) is a measure of the estimated signal to the difference between the reference and estimated signal. It is less commonly used in source separation.

$$SIR = \frac{\left\|s_{target}(t) + s_{interf}(t)\right\|^2}{\left\|s_{noise}(t)\right\|^2} \tag{2.4}$$

2.5 PEASS

2.5.1 Interference-related perceptual score

The interference perceptual score (IPS) is a method of transforming a set of Signal to Distortion ratios into a score that is more representative of the human perception. This evaluation is done using the PEASS evaluation toolkit and using a non-linear mapping using a Single-output two-layer Perceptron network with one hidden layer [11]. The Perceptron was first introduced in 1958 by Rosenblatt [28]. It is a supervised learning algorithm that aims to model the neurons of the human brain and while initially used in binary classifiers, it can be used with non-linear 'smooth' activation functions to learn non-linear mappings [12]. In the mapping to give more context to the SDR over a set of examples. The IPS score is sometimes referred to as more important than the SDR scores individually [8].

A correlation study for the PEASS and BSS Eval metrics was performed by Cano et al [5]. The PEASS is considered a better toolkit for analysis of musical signals but while PEASS may work very well on the kinds of algorithms it was designed with, it does not generalize well to other types of algorithms as the ones used in this work. Often BSS Eval and PEASS scores were not a good predictor of the MOS scores obtained using listening tests. This shows a clear discrepancy between the analytical metrics used for evaluation and human listening scores. As a result, it is unclear whether PEASS should be used to indicate how a human would interpret the result of a seperation algorithm.

Furthermore, Ward et al [40] notes that further work is needed to reduce the error in line with tolerable limits.

2.6 Listening Tests

Less common than Signal ratios, listening tests provide direct feedback from human listeners to evaluate separation algorithms. Advantages of listening tests result from the human interaction, and the applications for many separation algorithms involve human listeners anyway. Hence, it is intuitive to perform listening tests in the evaluation of newly developed system. There are clear disadvantages however. It takes longer to collect data, it can be expensive and individuals or groups can have biases which can be difficult to account for. The most common listen test formats for separation algorithms are Mean Opinion Scores (MOS) and MUlti-Stimulus test with Hidden Reference and Anchor (MUSHRA)

2.6.1 Mean Opinion Scores

MOS was originally developed as an ITU standard for transmission of audio [1] but has been co-opted for use in singing voice separation [17] [9]. The method involved is a five point category judgement scale as shown in 2.1. The scale has become more common in recent individual source separation research but remains useful as a tool for comparing algorithms [5].

Listening Quality Scale	
Quality	Score
Excellent	5
Good	4
Fair	3
Poor	2
Bad	1

Table 2.1: MOS scale

MOS is often used as an extra metric to provide further analysis of source separation systems. It is mostly used in literature for comparative studies on the same dataset (typically MusDB) [10] [9] [20]. However, in the meta-analysis, SiSEC campaigns there are no perceptual evaluations.

2.6.2 MUSHRA

MUSHRA is a methodology for conducting listening tests. It is based on a 0-100 scale that is defined in ITU-R recommendation BS.1534-3 [4]. The MUSHRA method defines the method of collection with the aim of reliable and repeatable results. The assessor should evaluate the audio on a scale of 1-100 continuous scale which are derived from the adjectives as seen in table 2.5

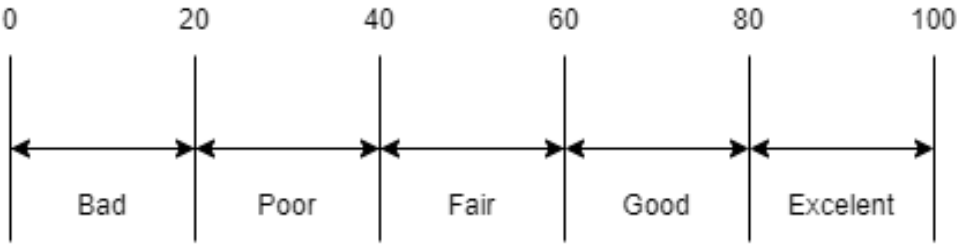

Figure 2.5: MUSHRA scale

MUSHRA has been used in evaluating singing voice separation and in investigations into different source separation metrics [40]. MUSHRA to the knowledge

of the author has not been used as a metric in any source separation literature. It is included in many datasets however.

2.7 Datasets

2.7.1 SEBASS

A Consolidated Public Data Base of Listening Test Results for Perceptual Evaluation of BSS Quality Measures is a collection of 5 musical datasets that have been used widely in source separation literature [15]. Each subset of data contain the signal mixtures, reference signals, separated signals, a label of the human listener, the separation algorithm and also human perception scores collected using the MUSHRA methodology as discussed in the previous section 2.6.2. The data is available online [1] and provides over a gigayte of data in total.

In table 2.2, the breakdown of the subsets within SEABASS is available. The datasets with parenthesis are split to prevent unwieldy excel datasheets. The list below references where each dataset was first presented or used in literature.

SASSEC Stereo Audio Source Separation Evaluation Campaign [39]

SiSEC08 The 2008 Signal Separation Evaluation Campaign [38]

SAOC First Stereo Audio Source Separation Evaluation Campaign [39]

PEASS-DB Subjective and objective quality assessment of audio source separation [11]

SiSEC18 The MUSDB18 corpus for music separation[38] [25]

Dataset	Algorithms	Separations	Listeners	Size	Number of Rating Scores
SASSEC	11	14	6	333Mb	2730
SiSEC08	9	14	14	282Mb	2156
PEASS-DB	6	10	7	83Mb	560
SiSEC18 (1)	21	6	19	359Mb	912
SiSEC18 (2)	27	9	14		1008
SiSEC18 (3)	26	9	11		792
SAOC (1)	7	14	12	615Mb	1512
SAOC (2)	6	14	9		1008
SAOC (3)	6	14	8		896

Table 2.2: Subset Datasets available in SEABASS

[1] https://www.audiolabs-erlangen.de/resources/2019-WASPAA-SEBASS

2.7.2 MUSDB-HQ

An extension of the MUSDB dataset that has been widely used in recently literautre is the MUSDB-HQ dataset [24]. This is a source separation dataset containing the reference mixtures and separate tracks for the same 150 songs contained in the MUSDB [25] but instead of being stored as compressed .mp4 files they are stored in raw wav files at a sample rate of 16kHz.At this time there does not appear to be any papers for any algorithms with rating scores for these files. The dataset is publicly available online [2]

[2]https://zenodo.org/record/3338373

Chapter 3

Design

To model how a human would score the similarity of two musical signals, two important design aspects arise. What is the best technique for a machine to learn such a process and what data should be presented to the model to learn from. The state of the art for many forms of artificial intelligence relies upon Neural Networks. A Neural Network is a machine learning technique vaguely inspired by the human brain through the use of layers of 'neurons' [29]. These layers are then connected together in different model architectures to make a Deep Neural Network. A deep neural network can be thought of as a universal function approximator which can produce a arbitrarily complex function. The universal approximation theorem shows that any function can be modelled [7]. However in reality, neural networks will not achieve this function perfectly for a number of reasons including poor or limited data, a finite amount of a computing time and complexity of the network required.

3.1 Dense Layers

A dense layer is comprised of many neurons which receive an input from previous layers or inputs from data. They compute this data and then feed the output forward to the next layer. The formula for each dense neuron can be seen in equation 3.1 where x is the input to the neuron, w is a learned weight associated with the neuron and b is the bias, another learned parameter. f is an activation function which calculates the output y based upon the input parameters. It is non-linear and there are several widely used activation functions today.

$$y_1 = f(w_1 * x + b_1) \tag{3.1}$$

A layer of a DNN may be made up of hundred or thousands of these neurons (typically a power of two). Each Dense node in a layer is connected to every node

in the previous and next layer. The equation for this next node after being passed the output of the first layer is shown in equation 3.2.

$$y_2 = f(w_2 * f(w_1 * x + b_1) + b_2) \tag{3.2}$$

For a deeper neural network a more complex function can be learned. This corresponds to better performance on a given task.

3.2 Convolutional Layers

A Convolutional layer is another form of popular layer in machine learning. These layers contain a set of learned filters (also called kernels). This form of layer is particularly useful when deal with image data as the layer is shift invariant [23]. The kernel is usually much smaller than the image and convolved with it to create an activation map. The weights of kernel are learned parameter. An activation map represents which part of the input to the layer is most salient. For example, given a convolutional filter of size 3x3 as in figure 3.1 the kernel is tuned for vertical lines as a feature. When passing over a vertical line the activation will be at it's most salient. Hence, the activation map is effectively showing us where the feature, in this case a veritcal line, is located in the image.

-1	9	-1
-1	9	-1
-1	9	-1

Table 3.1: Vertical Edge Detection Filter

This activation map is then used as the input to a non-linear activation function like a dense layer. This allows the convolutional layer to learn non-linear features. Typically, multiple convolutional layers are used where the output from one is passed to the next. This allows for more complex features to be learned the deeper the network. Convolutional layers are made up of multiples of these filters, again usually as a power of two.

3.2.1 Activation Functions

Activation functions are used to give layers in neural networks non-linearity [18]. There are many popular functions some of which are shown in figure 3.1. The choice of activation function is dependant on the application and data. The Rectified Linear Unit function known as Relu is extremely popular. However, for negative values the gradient of Relu is 0. During training of a neural network this

means if the input are negative the network will not learn from the new information. If there are negative values tanh may be appriote but only if the values of the inputs are bounded between -1 and 1. For the data in the SEABASS dataset, there are positive and negative values greater than 1 and less than -1. The exponential linear unit (elu) would be appropriate in this case then and is defined in equation 3.3.

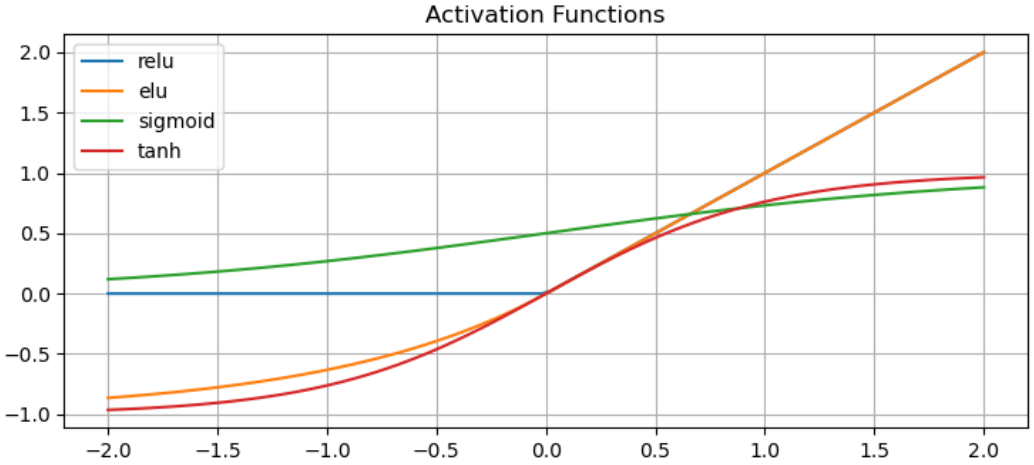

Figure 3.1: Activation functions

$$Elu(x) = \begin{Bmatrix} x & x > 0 \\ \alpha.(x^{\alpha} - 1)) & x <= 0 \end{Bmatrix} \tag{3.3}$$

3.3 Preprocessing and Features

3.3.1 Average Rating Scores

The audio and rating scores form the raw data for the inputs to the model. However there is a huge variation in the rating score between listeners. A result of this is that when trained on raw data, the model finds it extremely tough to learn anything significant. A possible solution to this is to average the rating scores between the listeners. This increases the quality of the data by showing what the average person would rate the separation. In the SASSEC dataset there are 10 listeners which results in a 90% reduction in data point. The averaging of rating score significantly reduces the amount of data however. This trade-off is necessary as models trained with the raw data consistently produced a constant rating score irrespective of the input audio. This shows that the raw features are not meaning-

ful. There is too much variance in the rating scores per listener for the network to learn to predict rating scores. A model was trained with the raw data and during inference on the SAOC dataset the predicted rating score would always output 34.051 as shown in figure **??**. The average rating score within the total dataset was 36.470 showing this averaging and the failure of the model to learn.

3.3.2 Spectrograms and Museval features

Convolutional data works best on image data. There is a history of convolutional neural networks for audio as discussed in 2.2.3. For the input tot he convolutional layers the raw audio is converted into spectrograms. For each channels in an audio track a spectrogram is calculated over a time step of 4096 samples. There 4096 faster transform bins per time step and the hop length is 1024. This results in a spectrogrm of with dimensions 1024 x 485 for each channel. An example of which can be seen in figure 3.2.

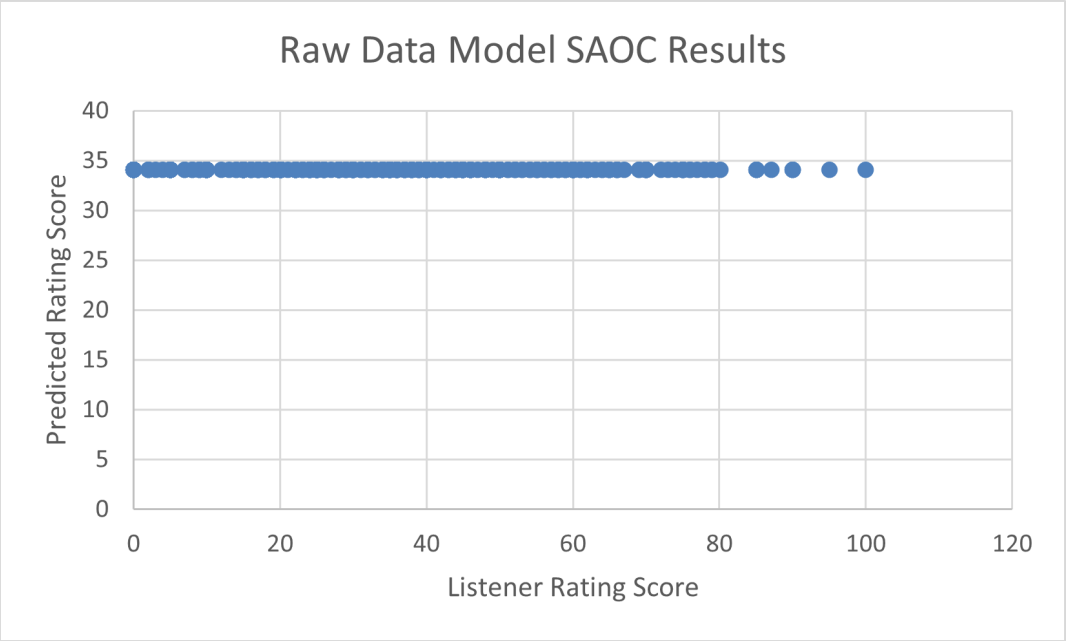

Figure 3.2: Constant, average output example

Features are also calculated explicitly from the raw audio. These features are the SDR, SIR, SAR and SNR. They are calculated using the museval python library [32].

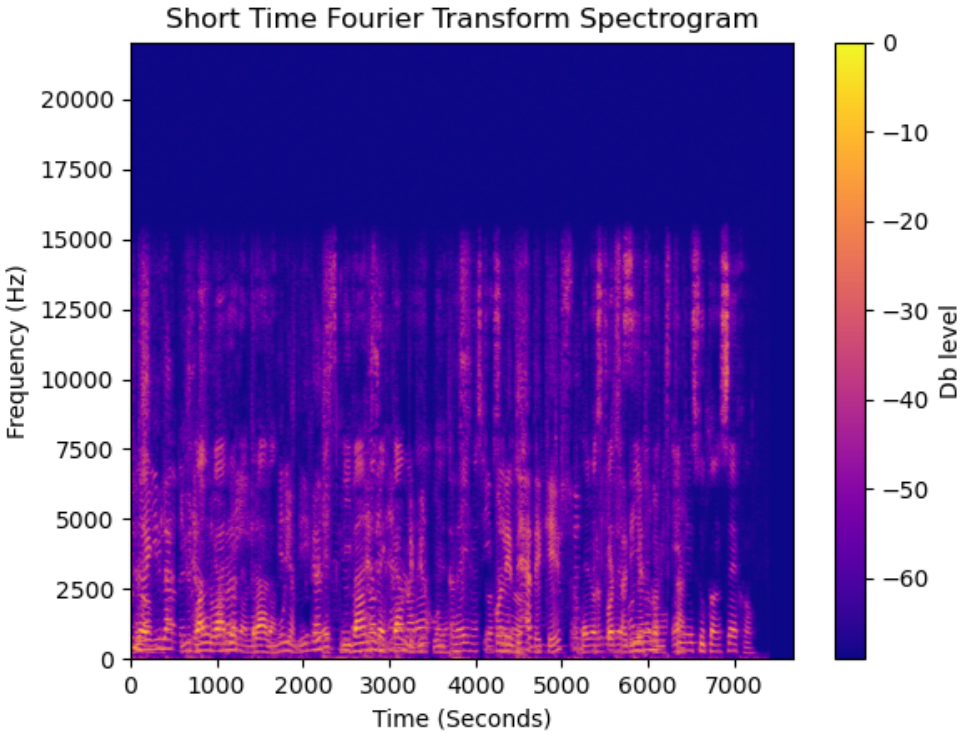

Figure 3.3: Spectrogram Example

3.3.3 Model Architecture

One of the most popular types of model for image classifacation is the VGG model. It has a series of convolutional layers followed by dense layers. The dense layers compute features of the spectrogram and the dense layers are used in VGG net to classify images. For this application, the network is used to predict a rating score and is therefore a regression model. The model employs series of convolutional layers with increasing filter sizes but using the Elu activation function. The max pooling is removed and batch normlisation is added after each convolutional layer. Due to constraints on the computing power available the layers in this model are smaller than are seen in VGG [30].

Aswell as the features generated by the convolutional layers, the SDR, SAR, SIR and SNR are calculated. These features are then concatenated at the beginning of the dense layers. This combined feature generation produced the best results compared using only one set of features. The model architecture is visualised in figure 3.4. For hyperparameters, the batch size is set to 8, this is constrained by the computing power available. The optimizer Adam with a learning rate of 0.01 and

a mean squared error loss function is utilised.

After each dense layer there is a dropout layer. Dropout is a regularisation technique which attempts to reduce the likelihood of overfitting and co-adaptation for a model. During each time step during training, a random set of neurons are turned off. When data is passed through the model, this means the neurons contribution to the next layer is not used and the weight of that neuron is not updated in that timestep. It prevents co-adaptation whereby different neurons have highly correlated behavior I.e. they have the same weights. Overfitting occurs when the network is large enough to 'learn' all of the inputs and outputs, this results in poor generalisation during inference when previously unseen data is used as input [31].

After each convolution layer and before the activation function, batch normalisation is applied. Batch normalisation standardises input to a layer for each mini-batch [14]. A mini-batch is a set of inputs which is less than the total total data. E.g For the input to the first convolutional layer, a mini-batch is a set of 8 two channel spectrograms.

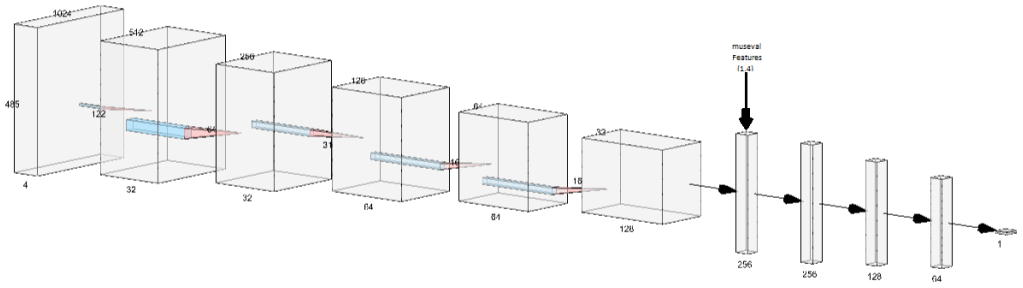

Figure 3.4: Model Architecture

The model is comprised of blocks of these convolutional and dense layers setup as shown in 3.5

3.3.4 Implementation

The model is implemented in Python 3 and Tensorflow 2. It is a free and open source library for machine learning. Dataset is loaded and the explicit features are calculated in the generate_dataset function. During development this allowed for greater flexibility with experiments for different models, features and hyper-parameters. however, this flexibility means there are no precalculated datasets. Preprocessed datasets such as those stored in hdf5 files are typically faster to load

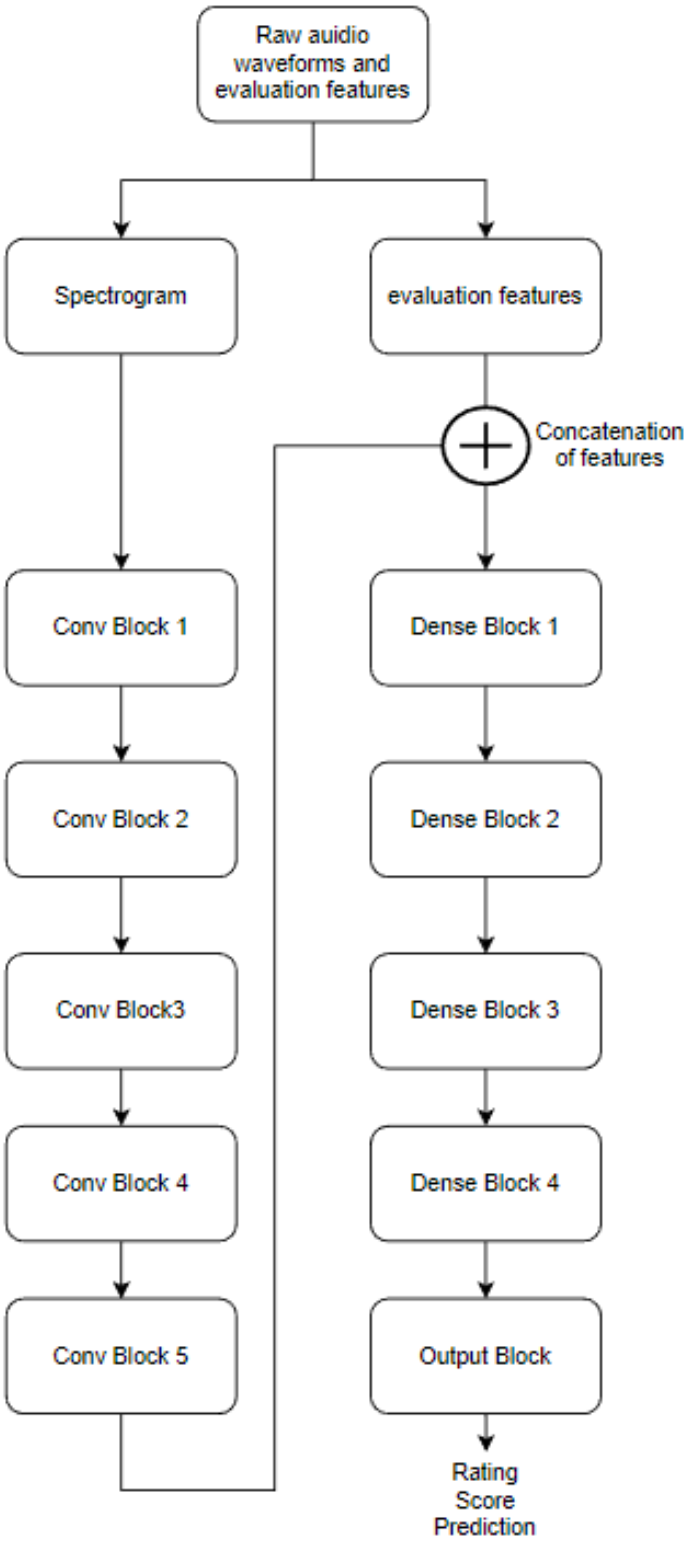

Figure 3.5: Block Architecture View

Block	Units	Kernel Size	Strides	Batch Normalisation	Activation Function	Padding
Conv Block 1	32	5x5	2,2	Yes	Elu	Same
Conv Block 2	32	3x3	2,2	Yes	Elu	Same
Conv Block 3	164	3x3	2,2	Yes	Elu	Same
Conv Block 4	64	2x2	2,2	Yes	Elu	Same
Conv block 5	128	2x2	2,2	Yes	Elu	Same

Table 3.2: Convolution Block Definitions

Block	Units	Activation	Dropout
Dense Block 1	256	Elu	0.2
Dense Block 2	256	Elu	0.2
Dense Block 3	128	Elu	0.2
Dense Block 4	64	Elu	0.2
Output Block 5	1	None	None

Table 3.3: Dense Block Definitions

and increase the speed of training. In training this model, the time to load the data proved a significant bottleneck.

Training for this model was done using the University of Aalborg machine learning workstation and took approximately 72 hours on a Nvidia TITAN X Pascal graphics card. The code for this project is available at:

https://github.com/dwoodw/audio-perception-loss.

Chapter 4

Evaluation and Results

The aim of audioConv is to replace or supplement the evaluation of blind source separation systems. The ideal model should predict the rating scores as accurately as possible. In order to do so, there should be a strong correlation between the predicted rating score and the listener rating scores in the datasets. For each data point we compute the predicted rating score and compare that with the user rating scores.As the ubiquitous measure of blind source source separation, as a bench mark a comparison should be made to the SDR's correlation with the user rating scores.

4.1 SASSEC

The SASSEC dataset is a subset of the SEASBASS. From 2008, it is oldest dataset used. It contains 2731 different trials and shows a correlation of 0.497 between the SDR and the listener rating scores. The audioConv metric shows a correlation of 0.814. A significant improvement in correlation over just the SDR as seen in figure 4.1 & 4.2

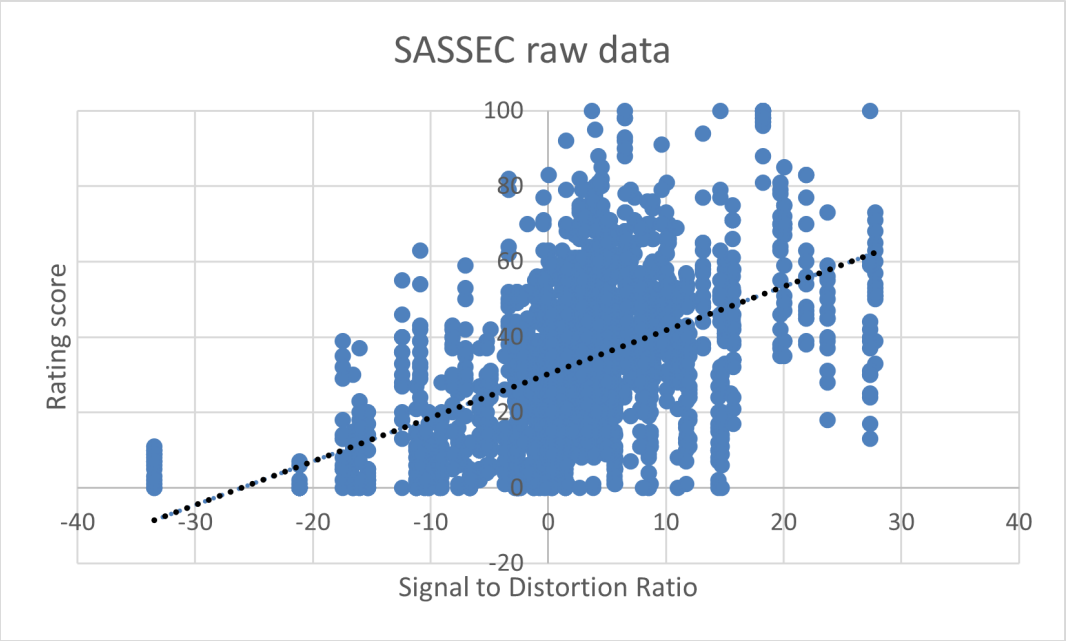

Figure 4.1: SASSEC SDR v Listener scores

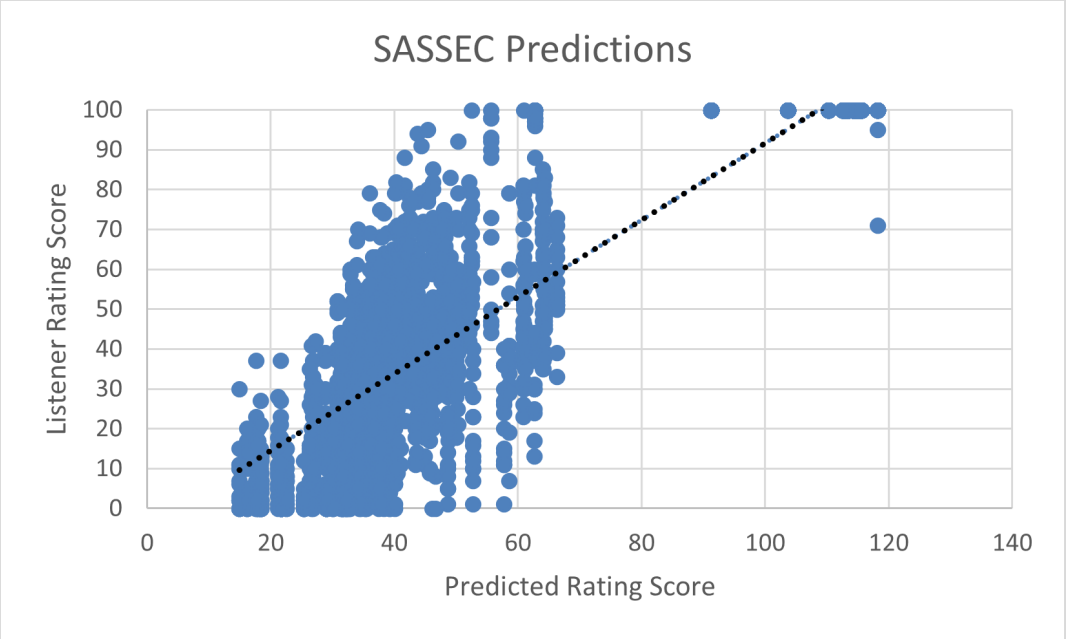

Figure 4.2: SASSEC audioConv v Listener scores

4.2 PEASS-DB

The PEASS-DB contains 490 datapoints and shows a very weak correlation between the listeners rating scores and the SDR of 0.153 in 4.3. The audioConv model shows a very weak correlation of 0.044 in 4.4. The model also fails to predict any rating scores between [0,14] and [66.100].

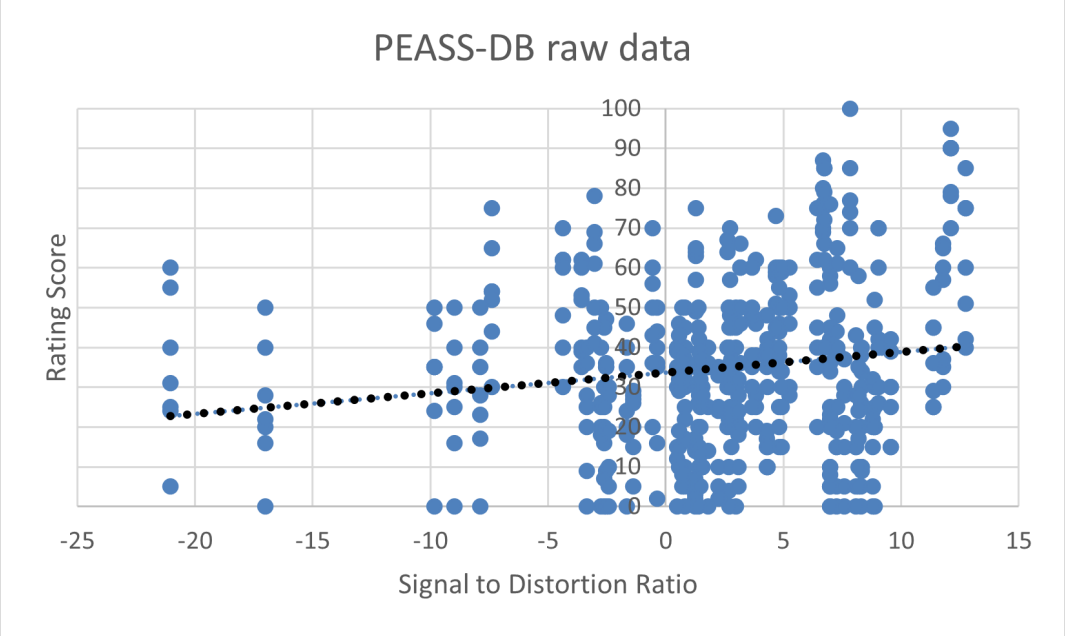

Figure 4.3: PEASS SDR v Listener Scores

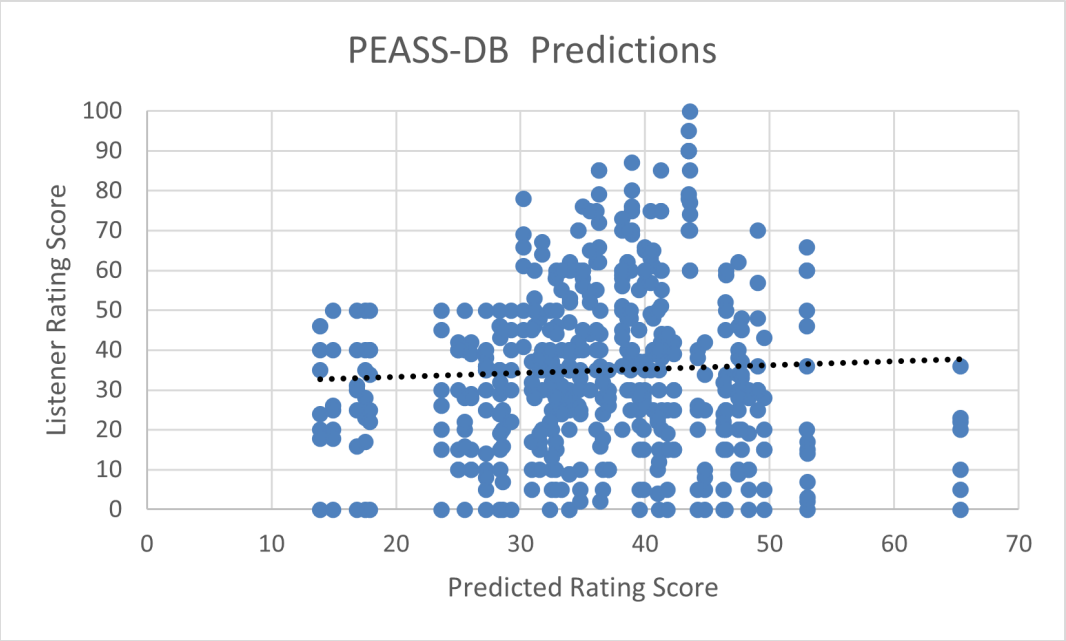

Figure 4.4: PEASS audioConv v Listener Scores

4.3 SiSEC08

The correlation is 0.538 for the SDR and 0.821 for the audioConv metric. Once again showing significant improvement over a large dataset.

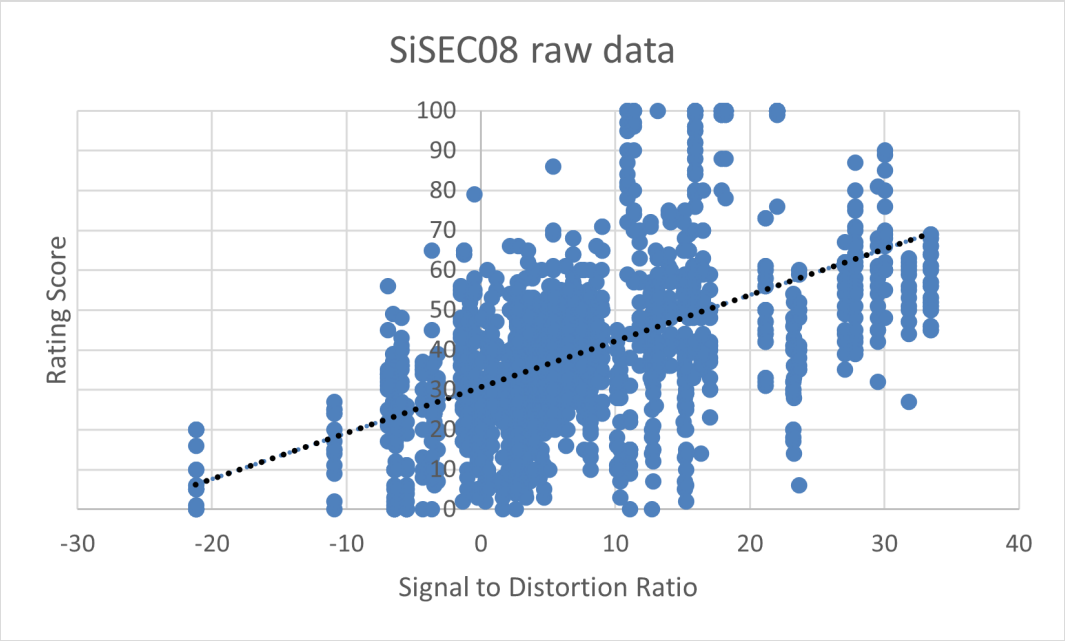

Figure 4.5: SiSEC08 SDR v Listener Scores

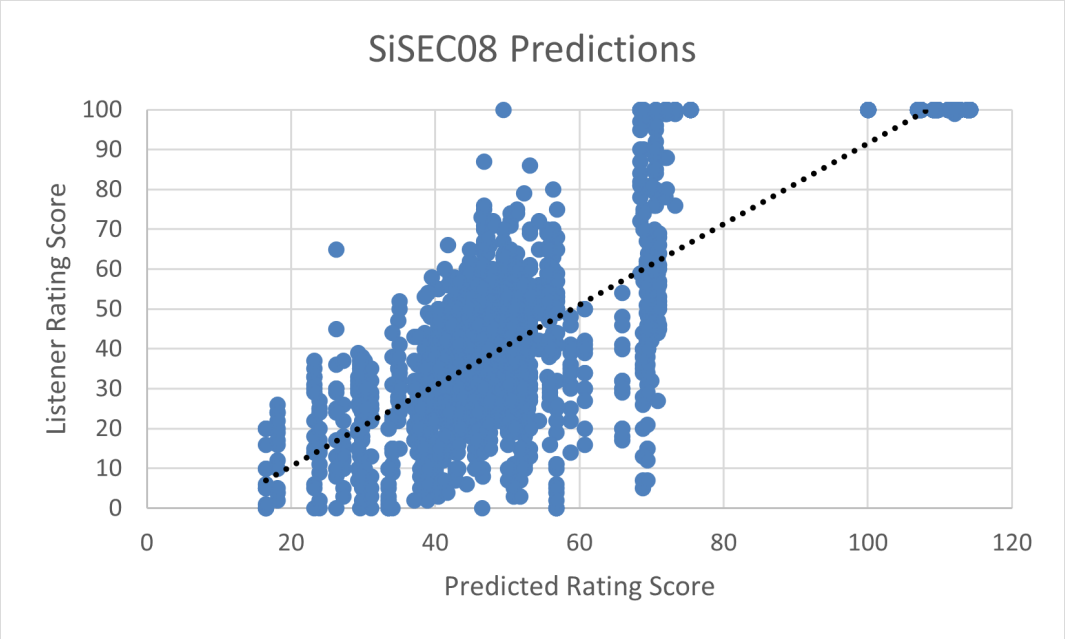

Figure 4.6: SiSEC08 v Listener Scores

Chapter 5

Further work

5.1 Datasets

In the duration of the project two different problems were presented as difficult to overcome. Firstly, the quality of data proved to be a significant issues. This is not due to the method of collection but because there are very few publicly available datasets. I recommend that a further dataset is collected using the most current source separation models. Neural networks benefit in general from more good data and I believe that the model would improve if a larger dataset was available. The methodology of collecting this dataset should also take into account some sort of training for the listeners to produce more consistent rating scores as this proved to be a problem throughout this project.

5.2 Model Architectures

The number of neural network architectures is always growing with larger and larger models proving to be the state of the art in many research areas. The nvidia titan that this model was trained on has 12Gb of memory which was the constraint for the size of the model. New hardware improvements could support bigger models that may perform better, for example convolutional layers the size of newer VGG style models.

Other types of network architecture could be employed to improve the model outcomes too. One further way of approaching this project that was not explored was treating it like a classification problem. Wavenet is an audio generator in the time domain and predicts the next point in the waveform by classifying what the most likely outcome would be. A similar approach could be adopted in model by aiming to classify the model into a rating score. This would also bound the output of the output of the model between 0 and 100.

Other features could also be produced for concatenation in the dense layers. Peaq is an audio evaluation metric for perceived audio quality. It could not be used in this case as there are separations in the dataset that do not contain voices and this is a requirement for Peaq. A model trained only for singing voice separation could used this feature.

Chapter 6

Conclusion

The goal of creating a new metric for blind source separation has not fully been realised but this thesis shows the potential of such a goal. Further work is needed to improve audioConv. The PEASS dataset shows only a very weak correlation and is definitely not ready to replace the SDR as the measure of blind source separation. However, for the large dataset the audioConv metric shows significant improvement over the SDR proving the potential of a Neural Network based model for evalua There are several suggestions in section 5 improve the model but the biggest issue is that the data available is extremely noisey and while data augmentation helps, the reduction in the size of the dataset results in too little data for the model to learn significantly. However, it is clear that there is meaningful information within the datasets and the correlations reflect that a future model could potentially imporve upon and replace the current practice of SDR only evluation.

Furthermore, the analysis of the dataset shows the drawback of using only SDR and in source separation models the use of only the SDR can now seen as potentially inaccurate in representing and comparing results.

Bibliography

[1] 1996. URL: https://www.itu.int/rec/T-REC-P.800-199608-I.

[2] Albert S Bregman. *Auditory scene analysis: The perceptual organization of sound*. MIT press, 1994.

[3] Guy J. Brown and Martin Cooke. "Computational auditory scene analysis". eng. In: *Computer speech & language* 8.4 (1994), pp. 297–336. ISSN: 0885-2308.

[4] Brweb. *Method for the subjective assessment of intermediate quality level of audio systems*. 2015. URL: https://www.itu.int/rec/R-REC-BS.1534-3-201510-I/en.

[5] Estefanía Cano, Derry FitzGerald, and Karlheinz Brandenburg. "Evaluation of quality of sound source separation algorithms: Human perception vs quantitative metrics". In: *2016 24th European Signal Processing Conference (EUSIPCO)*. 2016, pp. 1758–1762. DOI: 10.1109/EUSIPCO.2016.7760550.

[6] E. Colin Cherry. "Some experiments on the recognition of speech, with one and with two ears". In: *The Journal of the Acoustical Society of America* 25.5 (1953), 975–979. DOI: 10.1121/1.1907229.

[7] G. Cybenko. "Approximation by superpositions of a sigmoidal function". eng. In: *Mathematics of control, signals, and systems* 2.4 (1989), pp. 303–314. ISSN: 0932-4194.

[8] Christian Dittmar and Meinard Müller. "Reverse Engineering the Amen Break — Score-Informed Separation and Restoration Applied to Drum Recordings". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24.9 (2016), pp. 1535–1547. DOI: 10.1109/TASLP.2016.2567645.

[9] Alexandre Défossez. "Hybrid Spectrogram and Waveform Source Separation". eng. In: (2021).

[10] Alexandre Défossez et al. "Music Source Separation in the Waveform Domain". eng. In: (2019).

[11] V Emiya et al. "Subjective and Objective Quality Assessment of Audio Source Separation". eng. In: *IEEE Transactions on Audio, Speech, and Language Processing* 19.7 (2011), pp. 2046–2057. ISSN: 1558-7916.

[12] Daniel Graupe. "The Perceptron". eng. In: *Principles of Artificial Neural Networks*. World Scientific Publishing Co. Pte. Ltd., 2013, pp. 17–36. ISBN: 9789814522748.

[13] Udit Gupta, Elliot Moore, and Alexander Lerch. "On the perceptual relevance of objective source separation measures for singing voice separation". In: Oct. 2015, pp. 1–5. DOI: 10.1109/WASPAA.2015.7336923.

[14] Sergey Ioffe and Christian Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: *CoRR* abs/1502.03167 (2015). arXiv: 1502.03167. URL: http://arxiv.org/abs/1502.03167.

[15] Thorsten Kastner and Jürgen Herre. "The SEBASS-DB: A Consolidated Public Data Base of Listening Test Results for Perceptual Evaluation of BSS Quality Measures, in-press". In: *IEEE International Workshop on Acoustic Signal Enhancement (IWAENC'22)*. in-press. 2022.

[16] Qiuqiang Kong et al. "Decoupling Magnitude and Phase Estimation with Deep ResUNet for Music Source Separation". eng. In: (2021).

[17] Wen-Hsing Lai and Siou-Lin Wang. "RPCA-DRNN technique for monaural singing voice separation". In: *EURASIP Journal on Audio, Speech, and Music Processing* 2022.1 (2022). DOI: 10.1186/s13636-022-00236-9.

[18] Johannes Lederer. "Activation Functions in Artificial Neural Networks: A Systematic Overview". In: *CoRR* abs/2101.09957 (2021). arXiv: 2101.09957. URL: https://arxiv.org/abs/2101.09957.

[19] Antoine Liutkus et al. "The 2016 Signal Separation Evaluation Campaign". eng. In: *LATENT VARIABLE ANALYSIS AND SIGNAL SEPARATION (LVA/ICA 2017)*. Vol. 10169. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017, pp. 323–332. ISBN: 9783319535463.

[20] Francesc Lluís, Jordi Pons, and Xavier Serra. "End-to-end music source separation: is it possible in the waveform domain?" eng. In: (2019). ISSN: 1990-9772.

[21] Yi Luo and Nima Mesgarani. "Conv-TasNet: Surpassing Ideal Time-Frequency Magnitude Masking for Speech Separation". eng. In: *IEEE/ACM transactions on audio, speech, and language processing* 27.8 (2019), pp. 1256–1266. ISSN: 2329-9290.

[22] Nobutaka Ono et al. "The 2015 Signal Separation Evaluation Campaign". eng. In: *LATENT VARIABLE ANALYSIS AND SIGNAL SEPARATION, LVA/ICA 2015*. Vol. 9237. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2015, pp. 387–395. ISBN: 9783319224817.

[23] Keiron O'Shea and Ryan Nash. "An Introduction to Convolutional Neural Networks". In: *CoRR* abs/1511.08458 (2015). arXiv: 1511.08458. URL: http://arxiv.org/abs/1511.08458.

[24] Zafar Rafii et al. *MUSDB18-HQ - an uncompressed version of MUSDB18*. Aug. 2019. DOI: 10.5281/zenodo.3338373. URL: https://doi.org/10.5281/zenodo.3338373.

[25] Zafar Rafii et al. *The MUSDB18 corpus for music separation*. Dec. 2017. DOI: 10.5281/zenodo.1117372. URL: https://doi.org/10.5281/zenodo.1117372.

[26] Dario Rethage, Jordi Pons, and Xavier Serra. "A Wavenet for Speech Denoising". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2018, pp. 5069–5073. DOI: 10.1109/ICASSP.2018.8462417.

[27] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". eng. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 9351. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2015, pp. 234–241. ISBN: 9783319245737.

[28] F Rosenblatt. "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain." eng. In: *Psychological Review* 65 (1958). ISSN: 0033-295X. URL: http://search.proquest.com/docview/1290896970/.

[29] Jürgen Schmidhuber. "Deep Learning in Neural Networks: An Overview". In: *CoRR* abs/1404.7828 (2014). arXiv: 1404.7828. URL: http://arxiv.org/abs/1404.7828.

[30] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *International Conference on Learning Representations*. 2015.

[31] Nitish Srivastava et al. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". In: *Journal of Machine Learning Research* 15.56 (2014), pp. 1929–1958. URL: http://jmlr.org/papers/v15/srivastava14a.html.

[32] Fabian-Robert Stöter, Antoine Liutkus, and Nobutaka Ito. "The 2018 Signal Separation Evaluation Campaign". In: *Latent Variable Analysis and Signal Separation: 14th International Conference, LVA/ICA 2018, Surrey, UK*. 2018, pp. 293–305.

[33] Fabian-Robert Stöter, Antoine Liutkus, and Nobutaka Ito. "The 2018 Signal Separation Evaluation Campaign". eng. In: *14th International Conference on Latent Variable Analysis and Signal Separation*. Vol. 10891. Lecture Notes in Computer Science 10891. Cham: Springer International Publishing, 2018, pp. 293–305. ISBN: 9783319937632.

[34] Naoya Takahashi and Yuki Mitsufuji. "Multi-Scale multi-band densenets for audio source separation". eng. In: *2017 IEEE WORKSHOP ON APPLICATIONS OF SIGNAL PROCESSING TO AUDIO AND ACOUSTICS (WASPAA)*. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. IEEE. NEW YORK: IEEE, 2017, pp. 21–25. ISBN: 9781538616321.

[35] Stefan Uhlich, Franck Giron, and Yuki Mitsufuji. "Deep neural network based instrument extraction from music". In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2015, pp. 2135–2139. DOI: 10.1109/ICASSP.2015.7178348.

[36] Stefan Uhlich et al. "Improving music source separation based on deep neural networks through data augmentation and network blending". In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017, pp. 261–265. DOI: 10.1109/ICASSP.2017.7952158.

[37] E Vincent, R Gribonval, and C Fevotte. "Performance measurement in blind audio source separation". eng. In: *IEEE transactions on audio, speech, and language processing* 14.4 (2006), pp. 1462–1469. ISSN: 1558-7916.

[38] Emmanuel Vincent, Shoko Araki, and Pau Bofill. "The 2008 Signal Separation Evaluation Campaign: A Community-Based Approach to Large-Scale Evaluation". In: *8th International Conference on Independent Component Analysis and Signal Separation (ICA)*. Paraty, Brazil, 2009, pp. 734–741. DOI: 10.1007/978-3-642-00599-2_92.

[39] Emmanuel Vincent et al. "First Stereo Audio Source Separation Evaluation Campaign: Data, Algorithms and Results". In: *7th International Conference on Independent Component Analysis and Signal Separation (ICA07)*. London, United Kingdom, 2007, pp. 552–559. DOI: 10.1007/978-3-540-74494-8_69.

[40] Dominic Ward et al. "BSS Eval or Peass? Predicting the Perception of Singing-Voice Separation". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2018, pp. 596–600. DOI: 10.1109/ICASSP.2018.8462194.

[41] Xianchuan Yu, Dan Hu, and Jindong Xu. *Blind source separation: theory and applications*. eng. Somerset: Wiley, 2013, p. 4. ISBN: 9781118679869.

[42] Zhi-Hua Zhou. *Machine learning*. eng. Gateway East, Singapore: Springer, 2021, pp. 15–16. ISBN: 981-15-1967-6.

Appendix A

Appendix A name

Here is the first appendix