Master thesis on Music and Sound Computing

Universitat Pompeu Fabra

Exploring Detection and Localization of Overlapping Sound Sources with Deep Learning

Francesca Ronchini

Supervisor: Andrés Perez Lopez

Co-Supervisor: Daniel Arteaga

August 2020

Master thesis on Music and Sound Computing

Universitat Pompeu Fabra

Exploring Detection and Localization of Overlapping Sound Sources with Deep Learning

Francesca Ronchini

Supervisor: Andrés Perez Lopez

Co-Supervisor: Daniel Arteaga

August 2020

Contents

Abstract

Sound event localization and detection (SELD) refers to the problem of identifying the presence of independent or temporally-overlapped sound sources, correctly determining to which sound class they belong, and estimating their spatial directions while they are active. Until recently, SELD has been considered and studied as two standalone tasks: sound event detection and sound event localization. Only in the last years, they started to be conjointly considered. Neural networks have become one of the prevailing method to approach the SELD task, with convolutional recurrent neural networks being among the most used systems.

The main scope of this project is to contribute to the SELD field, exploring the field of research of sound event detection and localization with deep learning. The algorithm presented in this work consists of a convolutional recurrent neural network using rectangular filters, specialized in recognizing significant spectral features related to the task. In order to further improve the evaluation metrics and to generalize the system performance to unseen data, the training dataset size has been increased using data augmentation. The technique used to create new samples, increasing the dataset size, is based on channel rotations and reflection on the xy plane in the First Order Ambisonic domain. This approach allows to improve Direction of Arrival labels keeping the physical relationships between channels.

In order to reach the described method, the study has been mainly divided into two experiments. During the first experiment, different rectangular filter shapes have been studied in order to understand the filter's size which gives the best performance and helps the network to properly learn frequency features with the aim to accordingly detect and classify events. The second experiment has been focused on reducing overfitting and further improve the evaluation metrics using data augmentation. In order to do so, the research has been principally concentrated on three data augmentation techniques: time stretching, pitch shifting, and channel rotations. Each technique has been independently explored. While time stretching and pitch shifting did not help to improve the results, channel rotation substantially

enhances the evaluation metrics.

The system presented in this project has also been submitted to the Detection and Classification of Acoustic Scenes and Events (DCASE) 2020 Challenge, which main purpose is to encourage the development of computational acoustic scene and event analysis methods, comparing different proposals using a common publicly available dataset. This year challenge consisted of 6 tasks, each centered on a particular aspect of detection and classification of acoustic scene. This project has been submitted as possible solution to the Task 3, focused on sound event localization and detection.

The system has been evaluated using the same dataset provided for the DCASE 2020 Challenge Task 3: TAU-NIGENS Spatial Sound Events 2020. The network predictions have been evaluated considering the joint nature of localization and detection, proposed as evaluation criteria for the corresponding task of the DCASE 2020 challenge. In particular, the task of sound event detection has been evaluated considering location-dependent F-score and Error Rate, considering a prediction as true positive only if under a distance threshold of 20° from the ground truth. The sound localization task has been evaluated considering classification-dependent Localization Error and Localization Recall, which are computed only if the prediction has been correctly classified. Evaluation results on the 6 splits development dataset show that the proposed system outperforms the baseline results, considerably improving Error Rate and F-score for location-aware detection.

Keywords: Sound Event Detection, Sound Event Localization, Direction of Arrival estimation, CRNN, First Order Ambisonic, Data Augmentation, SELD.

Chapter 1

Introduction

1.1 Sound event localization and detection

Sound event localization and detection **(SELD)** refers to the combined task of sound event detection **(SED)** and sound event localization **(SEL)**, whose aim is the recognition of sound sources and their spatial location. In particular, SED requires to identify, instantaneously, the onset and offset of sound events and their correct classification, labeling the event according to the sound class that they belong to, as illustrated in Figure 1. The figure shows a sound wave with different categories of sounds such as car, speech and footsteps. The aim of the SED task is twofold. The first intention is to understand where each sound starts and disappears, while the second goal is to correctly label the sound that has been detected, associating it with the correct belonging class. SEL is defined as the estimation of the sound event direction in space with respect to a microphone when an event is active, referred to as Direction-of-Arrival (DOA) estimation. Figure 2 is a representation of the polar coordinate system, which is used in this study to evaluate the DOA of the sound sources.

Developing a system able to efficiently perform SED and SEL tasks combined together might allow humans and machines to automatically interact with the acoustic scene around them, beyond having huge impact in fields such as robotic, smart

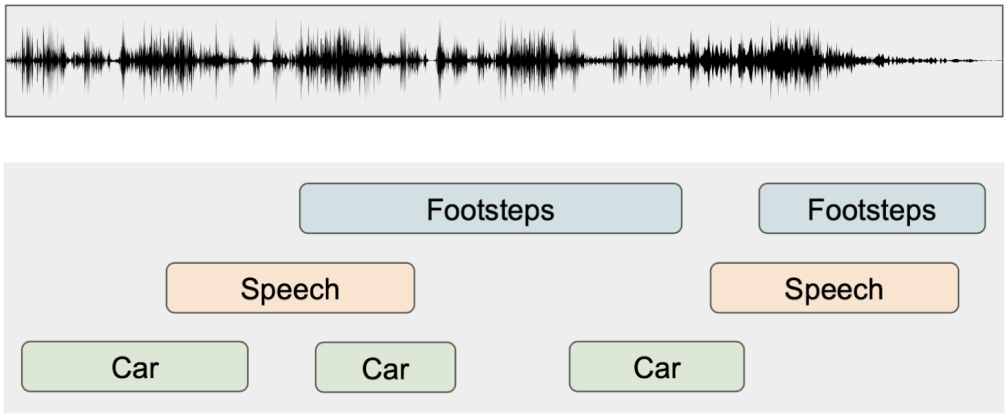

Figure 1: Representation of sound event detection task.

cities for audio surveillance [1] or teleconferencing systems [2]. Other relevant fields of SELD application are virtual reality, smart meeting rooms, where speakers can be identified and localized [3], and bio-diversity monitoring, where animal population density can be estimated using passive acoustic [4].

Formerly, SED and SEL have been explored as two standalone tasks. Only in recent times they started to be considered jointly. In fact, until 2018, the proposed systems considering SELD as a single task were scarce, with only one method based on deep neural network [5]. In this method, sound events are localized exclusively at a predefined grid of directions, and a large number of output classes were required for an higher number of sound event labels and increased spatial resolution. In 2018, Advanne et al. introduced SELDnet [6], a convolutional recurrent neural network (CRNN) which simultaneously recognizes, localizes and tracks sound event sources, being the first method to address the localization and recognition of more than two concurrent overlapping sound events. Moreover, the system is able to localize sources at any continuous spatial position, and it is robust to new spatial locations, reverberation, and background sound, beside being generic to any input array configuration. The system was proposed as baseline for the Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge 2019 Task 3 [7] and for the same task of DCASE Challenge 2020 [8]. This year's baseline system includes some modifications inspired by the highest ranked architectures of last year's challenge submissions, among which can be highlighted [9], [10] and [11], which are

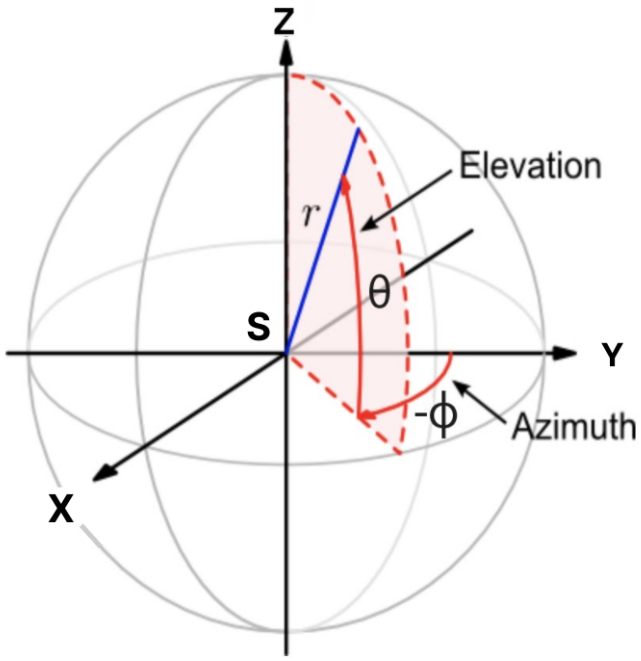

Figure 2: Representation of the polar coordinate system used to evaluate the DOA of sound sources.

further detailed in the Section 2.3.3 of Chapter 2. Further information regarding the differences between the SELDNet of this year and the previous year challenge are given in Chapter 3.1. As an overview, the network proposed as baseline this year receives as input features the normalized log-mel spectral coefficients, together with generalized cross-correlation (GCC) for the microphone array (MIC) format, and the acoustic intensity vector in the First Order Ambisonic (FOA) domain. Regarding the network architecture, the model is initially trained with a SED loss only, and then continued with a joint SELD loss. The localization part of the joint loss is masked with the ground truth activations of each class, so that if an event is not active, it does not contribute to the training of the network. The SELDNet system proposed for the DCASE Challenge 2020 has been considered as baseline of this study.

1.2 Objectives

The main scope of this project, beside participating and submitting the system to the DCASE Challenge 2020 Task 3, is to improve assisted listening system approaches,

exploring the sound event detection and localization field with deep learning. In fact, several neural network architecture configurations, filter shapes and dimensions, and data augmentation techniques have been implemented and explored before reaching the network architecture described in Chapter 3.

The methodology presented in this work is based on the SELDNet proposed by Adavanne et al. [6], including some of the adopted additions in the baseline algorithm of the DCASE 2020 Task 3, such as the use of log-mel spectral coefficients and acoustic intensity vector for the FOA format. However, the system proposed in this study differs from the baseline system presented for the DCASE 2020 Challenge, especially regarding the following three main points:

1. Network architecture
2. Training loss functions
3. Data augmentation

Regarding the network architecture, 2 convolutional layers have been added, increasing the convolutional layers number from 3 to 5. Furthermore, the receptive field along the frequency dimension has been expanded using rectangular filters (instead of squared ones) in order to make the network able to recognize spectral features relevant for the task. With regard to training loss functions, this study uses the same functions of the baseline system of the last year challenge [6]. Instead of masked mean square error (used in the baseline system of this year), binary cross-entropy loss is used for the SED prediction task and mean square error (MSE) loss for DOA estimation. With respect to data augmentation, different methods has been explored: time strecthing, pitch shifting and channel rotations. Among those, -90°, +90° and +180° channel rotations of the azimuth angle ϕ and its position reflection on the xy and xz planes are used as final data augmentation method, implementing the *16 patterns* spatial augmentation as proposed by Mazzon et al. for the same task of last year's challenge [12]. The 16 patterns technique allows to augment DOA labels maintaining the physical relationships between channels.

Results on the development dataset are evaluated considering the metrics proposed by Mesaros et. al. [13], which take into account the joint nature of localization and detection. In particular, a prediction is considered as true positive only if under a distance threshold from the ground truth, in this project set to 20°, and only if correctly classified. This is considered in order to avoid ambiguous situations where the prediction would have been correctly classified but not localized and vice versa.

1.3 Detection and Classification of Acoustic Scenes and Events Challenge and Workshop

Detection and Classification of Acoustic Scenes and Events (DCASE) is a rapidly growing field of research, with always more companies and researchers working on the field. Starting from 2010, the DCASE Community started to grow more and more with the first DCASE Challenge being organized in the 2013 by the Audio and Acoustic Signal Processing (AASP) technical committee of the IEEE Signal Processing Society. Every year, a DCASE Challenge and Workshop are organized with the main purpose of bringing together student, researchers and industries working on the field.

The DCASE Challenge aims to encourage the development of computational acoustic scene and event analysis methods, comparing different proposals using a common publicly available dataset. This year challenge consisted of 6 tasks, each centered on a particular aspect of detection and classification of acoustic scene. This project has been submitted as possible solution to the Task 3, focused on sound event localization and detection. The project has been submitted together with a technical report explaining the system, which can be find at [14].

The DCASE Workshop main purpose is to make possible for researchers working on computational analysis of sound events to present and discuss their systems, idea, opinions and results. A paper describing the system and the results reached with the implementation of it has been submitted for the DCASE Workshop 2020. The paper has been accepted and will be presented at the DCASE 2020 Workshop.

1.4 Structure of the Report

The report is structured as follows: Chapter 2 summarizes the state of the art regarding sound event localization and detection. Chapter 3 presents the methodology and architecture of the proposed system. Chapter 4 describes the experiment setup. Chapter 5 reports the development results compared with the baseline method. Conclusions and future work are presented in Chapter 6. In order to promote reproducibility, the code is available under an open-source license on GitHub [1].

[1] https://github.com/RonFrancesca/dcase2020-fp

Chapter 2

State of the art

The sound event localization and detection (SELD) is nothing but the combination of two independent minor tasks: *localization* and *detection*. Each of them have been studied for long time, especially for the critical importance of their applications [15]. In literature each of the sub-task has been approached in different ways. Section 2.3 presents and discusses the main techniques used to explore each task. In order to better understand the context of this thesis, a brief review of First-Order Ambisonic (FOA), in Section 2.1, Microphone Array recording format (MIC), in Section 2.1.1, Deep learning Neural Network (with emphasis on Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN)), in Section 2.2, are also given.

2.1 Ambisonics

Ambisonics has been introduced in the 1973 by the British engineer Michael Gerzon [16] to overcome the limits of traditional multichannel audio techniques. Unlike traditional multichannel systems (stereo, 5.1 or 7.1 channels), where each channel transmits signal to a particular loudspeaker, Ambisonics encodes physical properties of the sound field for each channel, such as the pressure or acoustic velocity, making possible to have a speaker-independent representation of the sound field which can be decoded according to the listener speaker setup [17]. In this way, it is possible to encode different portions of the sound field originating from many different di-

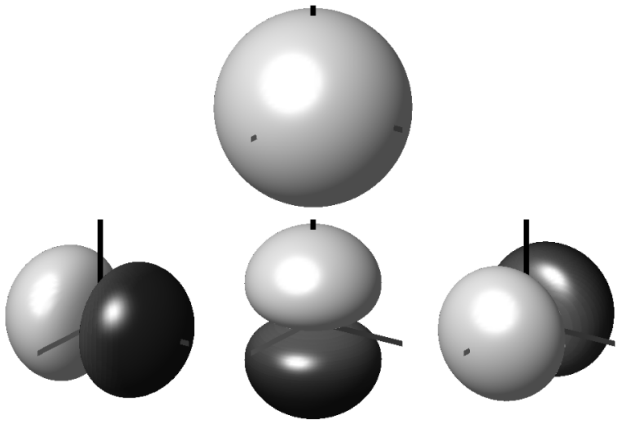

Figure 3: Visual representation of the First Order Ambisonic B-format. Adapted from figure in [17].

rections around a listener's position, considering not only the horizontal plane but also sources coming from above and below the listener, allowing people to think in terms of source directions instead of loudspeakers layout [18]. Therefore, Ambisonics is compatible with any configuration of any number of loudspeakers, with not limitation to the total transmission channels; the higher the number, the greater the directional resolution [19].

With the Ambisonics technology the sound field information can be stored, recorded, transmitted and processed using different formats and conventions. This project will consider the *B-format*, which is a synonym for First Order Ambisonics (FOA) and it is the main professionally and studio format used [19].

Ambisonic B-format uses four channels to encode information of a directional sound field denoted as *W, X, Y* and *Z*. In particular, W corresponds to an omnidirectional microphone recording the sound pressure. The signals X, Y and Z correspond to directional *figure-of-eight microphones* allocated along the components of the x, y and z axis respectively and measure the acoustic velocity of each directional component. In fact, if particular physical information such as sound pressure and particle velocity are known, the sound field can be uniquely determined [20]. Figure 3 reports a visual representation of the First Order Ambisonic B-format, with the omnidirectional microphone on the top and three figure-of-eight directional microphones in

each of the three direction at the bottom.

Considering a mono sound signal $s(t)$ and two parameters ϕ and θ representing respectively the azimuth angle and the elevation angle, Ambisonics B-format encodes pressure and acoustic velocity at the origin of the signal in the following way:

$$W(t) = s(t)/\sqrt{2} \tag{2.1}$$

$$X(t) = s(t)\cos\phi\cos\theta \tag{2.2}$$

$$Y(t) = s(t)\sin\phi\cos\theta \tag{2.3}$$

$$Z(t) = s(t)\ sin\theta \tag{2.4}$$

The channel W(t) represents the pressure field at the origin while the channels X(t), Y(t) and Z(t) express the acoustic velocity in each axis. In fact, being omnidirectional, the W channel always gets the same constant input signal, regardless of the angles. The term $\sqrt{2}$ attenuates the W signal in order to have the same average energy as the other channels [16]. This is a normalization scheme, known as the Furse-Malham normalization; other normalization schemes are also possible [21].

The directionality of the sound field could be improved and expanded using Higher Order Ambisonics (HOA), adding more selective directional components, as showed in Figure 4. The HOA order is defined as the maximum order l at which the directionality is expanded. Each order requires $(l+1)^2$ channels. In case of horizontal-only reproduction (2D case) $2l + 1$ components are necessary.

Apart from synthetically producing an Ambisonic signal from a mono recording, ambisonic recordings can be captured using ambisonic microphones. Among those microphones, which are nothing but spherical microphone arrays, the simplest and most popular capsule arrangement is the tetrahedron, which records First Order Ambisonics. An example of soundfield microphones (commercial name of a specific FOA microphone) can be seen in Figure 5.

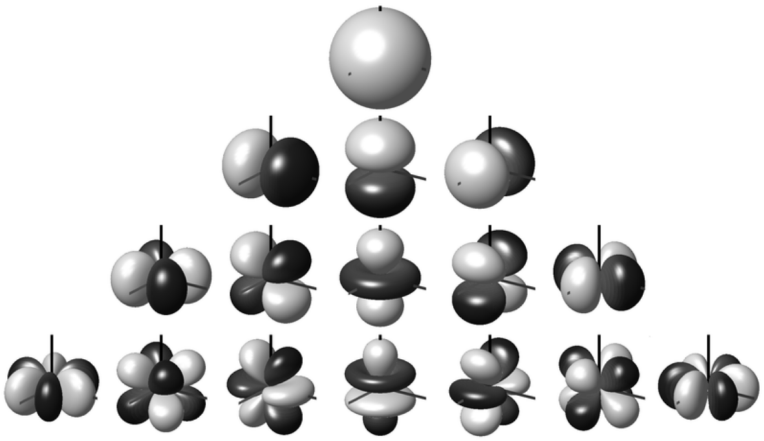

Figure 4: Visual representation of the Higher Order Ambisonic. Adapted from figure in [18].

Figure 5: RODE NT-SF1 soundfield microphone.

For more information regarding Ambisonic, the reader is referred to [17] and [22].

2.1.1 Microphone Array

A microphone array is defined as any number of microphones operating in tandem. The microphones composing the array could be omnidirectional, directional or a mix of both [23]. Usually, a beamformer combines the microphones allowing to record sound signals depending on their direction of propagation. Several geometry configurations are typically used, where one of the most common is the linear configuration. Anyway, this project considers the *spherical array geometry* which has the

advantage of leading the beampattern to any direction in 3-D space with no need of changing the shape of the pattern, allowing full 3-D control of it. The goal of this configuration is to design an array able to record spherical harmonics of higher order [20]. The prime design purpose is to record temporal and spatial information of the soundfield at the array position. As already seen for the Ambisonics, the soundfield can be uniquely determined knowing particular physical information such as sound pressure and particle velocity. Even if, theoretically, many other shapes are possible, the spherical geometry allows to keep the mathematics as simple as possible and, symmetrically, consider equal weight on all directions. For more information the reader is referred to [20].

In this project, the data has been recording using an *Eigenmike* spherical microphone array [1]. From its thirty-two capsules, only four have been used; specifically, the four of them forming a regular tetrahedron.

The following expansion gives the analytical expression [2], for the directional array response:

$$H_{\mathrm{m}}(\phi_{\mathrm{m}}, \theta_{\mathrm{m}}, \phi, \theta, \omega) = \frac{1}{(\omega R/c)^2} \sum_{n=0}^{30} \frac{i^{\mathrm{n\text{-}1}}}{h_{\mathrm{n}}{}'^{(2)}(\omega R/c)} (2n+1) P_{\mathrm{n}}(cos(\gamma_{\mathrm{m}}))$$

where m is the channel number, (ϕ m, θ m) are the specific microphone's azimuth and elevation position, $\omega = 2\pi f$ is the angular frequency, $R{=}0.042m$ is the array radius, $c{=}343m/s$ is the speed of sound, $cos(\gamma_m)$ is the cosine angle between the microphone and the Direction Of Arrival (DOA), P_n is the unnormalized Legendre polynomial of degree n [24], and $h_n{}'^{(2)}$ is the derivative with respect to the argument of a spherical Hankel function of the second kind [25]. The expansion is limited to 30 terms which provides negligible modeling error up to 20kHz.

[1] `https://mhacoustics.com/products`

[2] `http://dcase.community/challenge2020/task-sound-event-localization-and-detection`

2.2 Neural Network

2.2.1 Concepts

The Artificial Neural Networks (ANNs) are computing systems and machine learning methods inspired by the biological neural networks simulating human brains [26]. These systems, even if not being programmed with task-specific rules, aim to learn how to perform tasks considering examples.

An ANN is modeled as a collection of interconnected units or nodes called *artificial neurons*, which represent the human brain neurons. In particular, each ANN is defined by three principal components:

- *Node character*, which defines how signals are processed by the node;
- *Network topology*, which defines how nodes are organized and connected each other;
- *Learning rules*, which defines how weights are initialized and adjusted.

Each component is briefly described in the following paragraphs. Anyway, a complete survey of the topic is beyond the scope of this thesis so the reader is referred to [27] for a complete overview of Neural Networks and Deep Learning.

Neural networks mainly consist of connections where each of them provides the output of one neuron as input to another neuron. Figure 6 shows an example of a basic node in a neural network. Each node receives multiple inputs from connections with other nodes. Each connection has an associated weight, which represents its relative importance. The weighted sum of the inputs is combined with the internal state by an *activation function*, producing an output that will be the signal input of neighboring nodes. The output could be optionally thresholded. This process can be expressed with the following expression:

$$y = f \sum_{i=0}^{n} (w_{\mathrm{i}} x_{\mathrm{i}} - T)$$

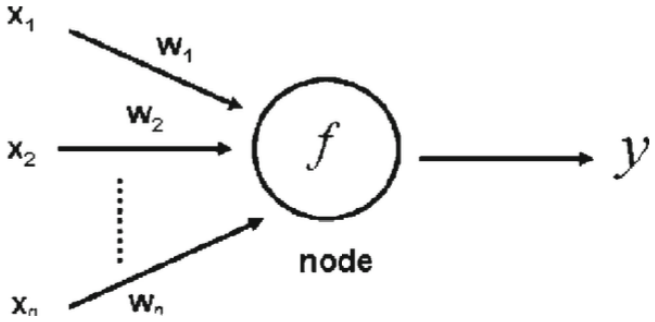

Figure 6: Example of a single node: x_i = input, w_i = weight, f = activation function and y = output. Adapted from figure in [26].

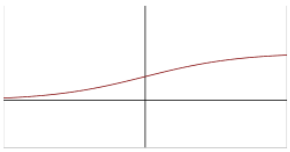

Figure 7: Logistic function.

where y is the output of the node, f is the activation function, w_i is the weight of input x_i, and T is the optional threshold value.

The activation function, also knows as *propagation function*, computes the input to a neuron from the outputs of its predecessor neurons and their connections as a weighted sum. A *bias term* can be added to the result of the propagation [28]. Usually the activation function is a non-linear function. In fact, they are considered to be more useful compared to linear functions since problems are not always linear separable [26]. Figure 7, Figure 8 and Figure 9 show the most used activation functions in literature.

Nodes are typically organized into multiple *layers*. A neural network is usually composed by an *input layer*, an *output layers* and *hidden layers*. The input layer takes external data as input, while the output layer produces the ultimate result. In between, there could be different hidden layers, which could vary from none to several [26]. Two layers could be connected in multiple ways. They can be *fully-connected* where every node of one layer is connected to every neuron of the next layer or, in case of *pooling*, a group of neurons in one layer is connected to only a

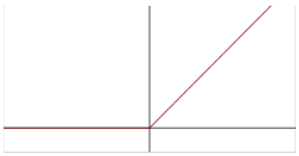

Figure 8: Rectified Linear Unit (ReLU).

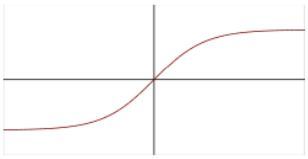

Figure 9: Hyperbolic tangent function.

single neuron in the next layer, reducing the number of neurons in that layer.

Neurons with only forward connections form a *directed acyclic graph* are known as *feed-forward networks*. Some networks allow connections between neurons in the same or previous layers, known as *recurrent networks* [28]. Figure 10 reports an example of a neural network with one hidden layer.

Learning is the adaptation of the network to better handle a task by considering sample observations. The learning procedure involves adjusting the weights and optional thresholds of the network to improve the accuracy of the result, minimizing the observed errors. The process is considered completed when the network keeps analysing new observations but the error rate is not usefully reduced. *Back propagation* mechanism is normally used for error correction [26]. The learning process can be classified into two main categories:

- *Supervised learning*: both training set (input examples) and the corresponding target output set are provided so to adjust the weights to minimize the error between the correct output and the network predicted output.

- *Unsupervised learning*: only the training set is provided. In this case the network tries to discover the underlying pattern or trend in the input data by itself.

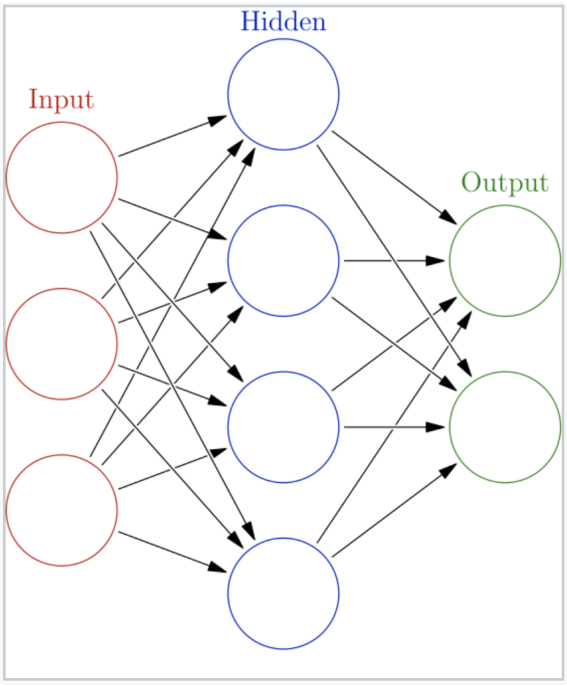

Figure 10: Example of artificial neural network with one hidden layers. Adapted from [28].

The learning process can be improved properly setting particular parameters, known as *hyperparameters* such as *learning rate* or number of hidden layers [27].

Furthermore, two or more networks can also be connected together (where the output of the first network is the input of the second network) or ensemble together [29]. This project implements the connection between *Convolutional Neural Network* (CNN) and *Recurrent Neural Network* (RNN).

Convolutional Neural Network (CNN) are specialized in processing data characterized by grid-like topology, such as images. The term "convolutional" indicates that the network implements a mathematical operation called convolution (a specialized kind of linear operation) in at least one of their layers [30]. They have been mainly motivated by three ideas which improve the machine learning system: *sparse interactions* (limiting the number of connections for each output using kernel filter of size n lower than input size), *parameter sharing* (using the same parameter for more than one function in a model) and *equivariant representations* (the output changes in the same way as the input changes) [27]. CNN are widely used in image

processing [31]. The reader is referred to [27] and [30] for further information about CNN.

Recurrent Neural Networks (RNNs) are specialized in processing temporal sequential data, such as a sequence of values $x^{(1)}$,..., $x^{(\tau)}$, thus providing context information for tasks based on sequential data, such as temporal context in audio tasks. In particular, RNNs allow to scale to longer sequences than would be practical for networks not specialized in handling sequences. Most recurrent networks can also process sequences of variable length [27]. The term "recurrent neural network" is usually refer to two categories of networks which general structure is quite similar: *finite impulse* and *infinite impulse*. Both structures can have additional stored states which could be directly controlled by the network. The storage can also be replaced by another network or graph. Such controlled states are referred to as *gated state* or *gated memory*, and they are part of *Long Short-Term Memory networks* (LSTMs) and *Gated Recurrent Units* (GRU) [32]. This work will mainly consider the GRU. For more information regarding RNN the reader is referred to [27] and [32].

2.2.2 Neural network and Deep Learning for audio

Neural network and Deep Learning techniques and methods started to be used and investigated in the audio field from 1988, when artificial neural network have been proposed for automatic music composition by Lewis et al. [33]. From 1989, other audio and music tasks, such as chord classification or classification of MIDI scores into music styles, started to be investigated by means of artificial neural networks [34] and [35].

In 1995, spectrograms have been used for the first time in machine learning to distinguish between genre of music [36]. Their use has been expanded, in 2002, to note onset detection [37], starting a new era of investigation from which researchers approach any task in an *end-to-end learning fashion*, meaning being able to solve a task learning a mapping system directly from a low level representation of the signal (waveform in this case). Together with it, the use of artificial neural network has been expanded also to classification tasks [33].

Around 2009, Deep Learning (multi-layered artificial neural networks) took over the field, with [38] being the first example of spectrogram-based CNN for music genre classification. From 2004 models started to be trained also using waveform and Mel-frequency cepstral coefficients (MFFCs) features.

However, one of the main limitations of deep learning and end-to-end systems is the requirement of large amounts of training data to reach good performances, limiting the application scenarios. To overcome this limitation, recent works are investigating alternatives to train deep neural networks with just few data [33]. Implementing models towards learning solutions inspired by musical or human perception seems to be a possible solution, together with data augmentation and unsupervised learning. The last two alternatives allow to create virtually infinite amounts of training data that have been studied to help increase deep learning performances [33].

2.3 Scene Description

2.3.1 Sound event detection

The sound event detection (SED) task is based on the identification of onset and offset of sound events and their correct classification, correctly identifying to which sound class they belong to. In literature, the task has mainly been approached with supervised learning algorithms.

In [39], which is one of the classical methods, a network of three-state left-to-right Hidden Markov Models (HMM) is used to model the events, which size and topology have been chosen based on a study of isolated events recognition. Based on these results, the event classification task has been performed using real-life recording, reaching a recognition performance not grater than 24%. Subsequently, performances have been improved with Fully Connected Neural Networks (FC) [40], Recurrent Neural Network and Convolutional Neural Network.

The introduction of spatial and harmonic features in combination with Long Short Term memory (LSTM) Recurrent Neural Network (RNN), as reported in [41], in-

creased the performance up to 48%. In [41], Advanne et al. extended the study to multichannel audio, mainly motivated by the human hearing system, which use two ears to localize and recognize sounds around them. In [42], Parascandolo et.al used Bi-directional Long Short Term Memory (BLSTM) Recurrent Neural Networks (RNNs) to approach the polyphonic sound event detection task (detection of overlapping sources), reaching performances of 64.6%. In the same study, the performance have been further increased up to 65.5% using data augmentation techniques such as *time stretching*, *sub-frame time shifting* and *blocks mixing*. In [43] and [44], state-of-the-art results have been overcome using CNN. The network proposed by Zhang et al. in [43] uses smoothed and de-noised high dimensional input features which yield to performance of 94%, which are further increased to 98.6% in [44] with a simplified CNN which selects the most discriminative features from the whole audio signals using only three layers: convolutional, pooling, and softmax layer. Recently, CNN, RNN and FC have been stacked together and First-Order Ambisonic (FOA) and binaural microphones recording have been used in order to better address polyphonic SED task. Relevant works are [45] and [46]. In [45], Çakir et al. proposed a CRNN which extracts high level features through multiple convolutional layers, pooling in frequency domain and feed the features to recurrent layers which obtain event activity probabilities through a feed-forward fully connected layer. In this way, performances of already proposed CRNN are increased gathering the ability of CNN to learn local translation invariant filters and the ability of RNN to model short and long term temporal dependencies in a single classifier. Those results have been further increased in [46], where Adavanne et al. show that using low-level features instead of high-level features allows the network to learn high-level equivalent information from simple low-level features. Other relevant studies are [47], introducing Bidirectional Gated Recurrent Unit (GRU) networks to model temporal evolution of audio features, and [48] where frequent validation with adaptive thresholds and class-wise early-stopping showed to optimize state-of-the-art strategies.

2.3.2 Sound event localization

Sound event localization (SEL) task is considered as the estimation of the sound event direction in space with respect to a microphone when an event is active, referred as Direction of Arrival (DOA) estimation. In literature, the task has been addressed with parametric-based approaches while, recently, deep learning methods have taken over.

Between the most prominent parametric methods there are MUSIC [49] and ESPRIT [50]. In [49], Schmidt et el . present a very general and of wide application multiple signal classification algorithm. The algorithm provides an asymptotically unbiased estimation of several signal parameters, including directions of arrival with no assumption of array geometry, which elements could be arranged in a regular or irregular way and may differ or be identical in directional characteristics. Nevertheless, MUSIC required high computational and storage cost which have been drastically reduced by Roy et al. with the introduction of Estimation of Signal Parameters Via Rotational Invariance Techniques (ESPRIT) [50]. In fact, ESPRIT allows to reduce those costs via rotational invariance technique, requiring that the sensor array possesses a displacement invariance. ESPRIT is also considerably more robust and less sensitive to array imperfections compared to previous techniques. Other relevant works are [51], based on time-difference-of-arrival (TDOA), [52], based on steered-response-power (SRP) and the directional audio coding (DirAC) technique [53], based on analyzing the sound direction and diffuseness depending on time at narrow frequency bands.

Recently, the same SEL task has been approached with deep learning in order to overcome limitations of parametric methods, while being robust towards reverberation and low signal-to-noise (SNR) scenarios (for which parametric methods were showed to be sensitive). In fact, neural network based methods require fewer strong assumptions about the environment. In [54], He et al. introduced a neural network not only able to localize single sound sources, but extending the method to the localization of an arbitrary number of overlapping sources, reaching a precision of

around 90%. [55] and [56] investigated CNN-based methods. In particular, in [55] Chakrabarty et al. explored the training of a CNN network with synthesized noise signals, using the assumption of disjoint speaker activities. In [56], Hirvonen et al. achieved cross-validation accuracy of 94.3% with a seven layers CNN network, where the first four layers are convolutional and the last three layers have full connectivity. In recent past, CNN and RNN have been stacked together, as reported in [57], where Advanne et al. proposed a CRNN (DOAnet) for estimating the directions of arrival (DOA) of multiple sound sources, which performed noticeably better than baseline outline. [58] and [59] are instead examples of regression approach, letting the network to produce continuous output.

2.3.3 Sound event detection and localization task

Sound event detection and localization (SELD) task is nothing but the combination of the previous two tasks: identification of onset and offset of multiple overlapped sound events and their classes classification. Related to SELD, the number of published approaches is limited and most of them do not localize more than one or maximum two sound sources, as reported in [1], [60] and [61]. Until 2018, only one method was based on deep neural network [5], which localizes sound events but exclusively at a predefined grid of directions and a large number of output classes were required for an higher number of sound event labels and increased spatial resolution.

In 2018, Advanne et al. introduced SELDnet [6], a convolutional recurrent neural network which recognizes and localizes sound event sources at the same time. The system is also the first to approach the localization and recognition of more than two concurrent sound events. Furthermore, the sound sources can be localized at any azimuth and elevation angles and it is able to maintain the same performances even with new spatial locations, reverberation, and ambiance. Other innovative aspects of the algorithm are the ability to be generic to any input array configuration (still being able to learn to properly perform SELD) and the extension of polyphony in SELD task. The system has also been proposed as baseline for the Detection and

Classification of Acoustic Scenes and Events (DCASE) 2019 Challenge Task 3 [3]. Different systems have been submitted as possible alternatives or solutions to the task challenge. Among the best architectures can be cited [9], [10] and [11].

In [9], Kapka et al. decompose the SELD task into four sub-tasks: estimation of number of active sources, estimation of direction of arrival of a single source, estimation of direction of arrival of the second source where the direction of the first one is known and a multi-label classification task. For each sub-task, the authors use a CRNN SELDnet-like single output model. The system outperforms the results of the proposed baseline model, but with some limitations, failing to consider general configuration.

Cao et al., [10], present a two-stage polyphonic sound event detection and localization method. In their study, the authors use log-mel input features for sound event detection. In order to help sound event localization predictions, intensity acoustic vector for Ambisonics and generalized cross-correlation (GCC) features for microphone array signals are also given as input features. Also in this case, the network is a CRNN with two branches, the SED branch and the DOA estimation branch. During training, the SED branch is trained first only for SED. The feature layers learned from the first step are transferred to the DOA estimation branch. The DOA estimation branch fine-tunes the transferred feature layers and, using the SED ground truth as a mask, learn only DOA estimation. During inference instead, the SED branch estimates the SED predictions first, which are used as the mask for the DOA estimation branch to infer predictions. The system significantly exceeds the baseline method.

In [11], Zhang et al. propose to use data augmentation and Mel-spectrogram as features for the network. The authors also propose a prior knowledge-based regularization (PKR) as post-processing method. In particular, it calculates the average value of the localization prediction of each event segment, replacing the prediction of this event with this average value. Their implementation helps to reduce the SED error by 59% and DOA error of 13%.

[3] `http://dcase.community/challenge2019/index`

The system presented as baseline [6] to the 2020 DCASE Challenge Task 3 is the same of the previous year, with some modifications inspired by one of the highest ranked architectures of last year challenge such as the use of log-mel spectral coefficients instead of spectrogram, together with generalized cross-correlation (GCC) features for the MIC format, and the acoustic intensity vector for the FOA format. Some changes have been implemented also for the neural network. In particular, the new model is initially trained with a SED loss only, then continued with a joint SELD loss and the localization part of the joint loss is masked with the ground truth activation of each class, avoiding to contribute to the training when an event is not active. Both changes have been motivated by [10]. For more information regarding the modifications introduced in the DCASE 2020 baseline system the reader is referred to [8].

Chapter 3

Methodology

The purpose of this chapter is to explain the details of the system implemented and submitted to the DCASE Challenge 2020 Task 3 and implemented for this project. In particular, Section 3.1 briefly presents the baseline system. Section 3.2, Section 3.3 and Section 3.4 respectively explain the feature extraction process, the network architecture and the data augmentation technique as implemented in this system. Section 3.5 reports the hyper-parameters used in this project. Section 3.6 details the different versions of the system submitted to the DCASE Challenge. In fact, the challenge allows up to four different submissions provided that each version should differ from another.

3.1 The baseline system

The state-of-the-art algorithm is known as SELDNet and it has been introduced by Adavannce et al. in [6]. SELDNet has been implemented to consider the SELD task, treating SED as a multi-label classification task, allowing the network to simultaneously estimate the presence of multiple sound events for each frame, and the DOA estimation as a multi-output regression task, where each sound event class is associated with three regressors that estimate the cartesian coordinates x, y and z of the sound source on a unit sphere around the microphone.

The system, as presented in [6] and illustrated in Figure 11, consists of a CRNN receiving raw multichannel magnitude and phase spectrogram as separate input features. The network implements three 2D CNN layers, each using a square 3x3 filter shape, rectified linear unit (ReLU) activation and max-pooling along the frequency axis to reduce dimensionality. The output of the CNN layers is fed to bidirectional RNN layers used to learn the temporal context information from the CNN output activations. Specifically, each RNN layer uses a Gated Recurrent Units (GRU) with tanh activation. This architecture is followed by two branches of FC layers in parallel, one for SED and one for DOA estimation, which share weights across time steps. In both cases, the first FC layer uses a linear activation. The last FC layer in the SED branch consists of N nodes with sigmoid activation, one for each sound event classes to be detected. The last FC layer in the DOA branch consists of 3N nodes with tanh activation, where each of the N sound event classes is represented by 3 nodes corresponding to the sound event location cartesian coordinates. Since the DOA estimation is considered on a unit sphere centered at the origin, the range of location along each axes is [-1, 1], so the tanh activation for these regressors keep the output of the network in a similar range.

Motivated by the positive results reached by the one of the highest ranked algorithm of the DCASE 2019 Challenge, in particular [10], the SELDNet authors decided to integrate some of the main enhancements in their system and proposed the enriched method as baseline for the DCASE 2020 Challenge. The main changes introduced in the baseline regard:

1. Feature extraction: the SELDNet proposed as baseline for last year challenge received raw multichannel magnitude and phase spectrograms as input features. The SELDNet proposed for the same challenge of this year receives the more compressed normalized log-mel spectral coefficients, together with generalized cross-correlation (GCC) for the microphone array (MIC) format, and the acoustic intensity vector in the First Order Ambisonic (FOA) domain.

2. Trainining loss functions: while last year system used mean square error as training loss function for the SED task, the 2020 model is initially trained with

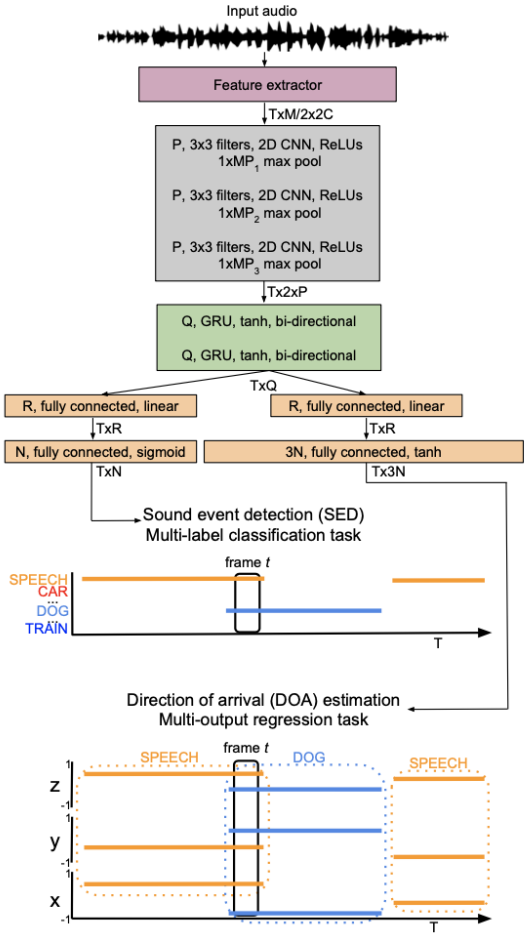

Figure 11: SELDNet architecture. Adapted from figure in [6].

a SED loss only, and then continued with a joint SELD loss. The localization part of the joint loss is masked with the ground truth activations of each class, so that if an event is not active, it does not contribute to the training of the network. Both models use binary cross entropy as loss function for SED predictions.

Table 1 summarizes the main differences between the 2019 and 2020 SELDNet system. Further information about the changes made in the baseline system of this year challenge can be found at [8].

SELDNet, with the integration of the previously mentioned modifications, has been considered the baseline of this project.

Year	Input features SED	Input features SEL	Loss functions
2019	Multichannel magnitude and phase spectrograms	Multichannel magnitude and phase spectrograms	Mean square error, binary cross entropy
2020	Normalize log-mel spectral coefficients	GCC (MIC format) Acoutic intensity vector (FOA format)	Masked mean square error, binary cross entropy

Table 1: Main differences between the SELDNet systems proposed as baseline for the DCASE Challenge 2019 and 2020.

3.2 Feature Extraction

The method presented in this report uses Ambisonic format data. Normalized log-mel magnitude spectrogram together with acoustic intensity vector are used as input features for the network. The log-mel magnitude spectrogram encourages the detection of sound, while the acoustic intensity vector intends to help the localization of sound. Both are represented in the log mel space to better concentrate the input information of the network, as proposed by Cao et. al in [10] and also implemented in the baseline system of this year challenge.

The Mel scale is a perceptual scale of pitches which is proportional to the perceived magnitude of subjective pitch, introduced by Stevens et al . in the 1937 [62]. The reference point between the Mel scale and frequency measurement is defined by assigning a perceptual pitch of 1000 Mels to a 1000 Hz tone. Starting from 500 Hz, increasingly large intervals are judged by listeners to produce equal pitch increments. The main formula used to convert frequency (Hertz) to mels (m) is the 3.1

$$m = 1127 \ln \left(1 + \frac{f}{700}\right) \tag{3.1}$$

where f is the frequency to be converted to mels. The log-mel spectrogram is intended as a spectrogram with the Mel Scale as its y axis. The Mel filterbank used in this project is shown is Figure 12. More information regarding the Mel scale can be found in [62].

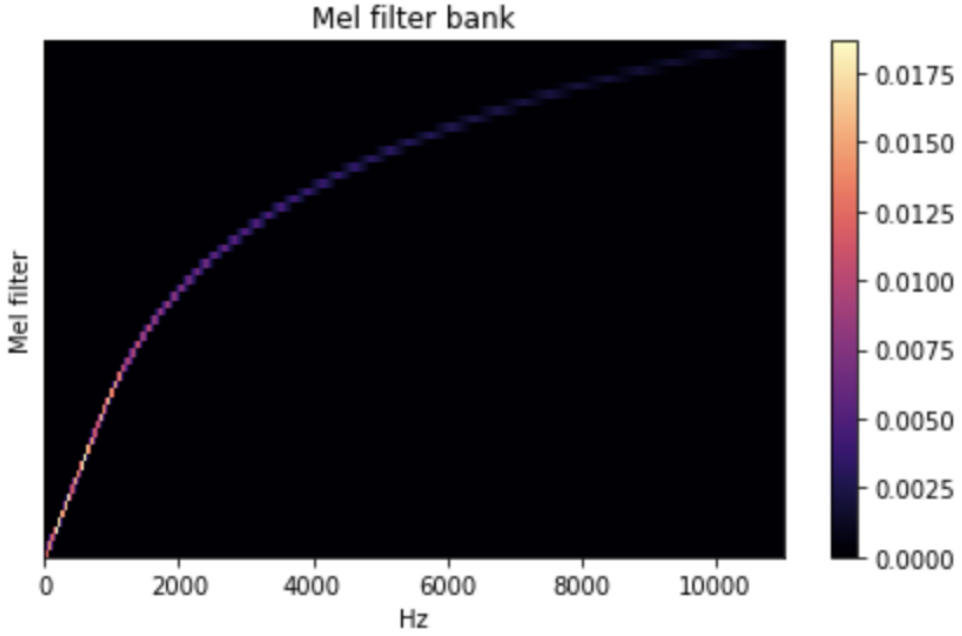

Figure 12: Mel filterbank used in the project as Mel scale for the spectrogram of the audio files.

FOA uses four channels to encode spatial information of a sound field, typically denoted as W, X, Y and Z. The channel W corresponds to an omnidirectional microphone recording the sound pressure. The channels X, Y and Z correspond to directional *figure-of-eight* microphones oriented along the components of the homonym cartesian axes, and measure the acoustic velocity of each directional component. The acoustic intensity vector expresses the energy carried by sound waves per area unit in a direction perpendicular to that area, providing DOA information. The acoustic intensity vector is computed as in [10], using the formulas 3.2, 3.3, 3.4, 3.5

$$I(f,t)_{\mathrm{x}} = \frac{1}{\rho_0 c}\Re\{W^*(f,t)\cdot X(f,t)\} \tag{3.2}$$

$$I(f,t)_{\mathrm{y}} = \frac{1}{\rho_0 c}\Re\{W^*(f,t)\cdot Y(f,t)\} \tag{3.3}$$

$$I(f,t)_{\mathrm{z}} = \frac{1}{\rho_0 c}\Re\{W^*(f,t)\cdot Z(f,t)\} \tag{3.4}$$

$$I_{\text{norm, mel}}(k,t) = -H_{\text{mel}}(k,f)\frac{I(f,t)_{\text{i}}}{\parallel I(f,t)_{\text{i}} \parallel} \tag{3.5}$$

where, ρ_0 and c are the density and velocity of the sound, W, X, Y, Z are the STFT of w, x, y, z channel, respectively, $\Re\ \{\cdot\}$ indicates the real part, $*$ denotes the conjugate, $\| \cdot \|$ is a vector's l_2 norm, k is the index of the mel bins, H_{mel} is the mel-band filter banks. The three components of the intensity vector are given as input features as three additional input channels to the neural network.

3.3 The network

Figure 13 shows the overall architecture of the proposed system with the relative parameters values used in this implementation.

The system architecture is based on the SELDNet introduced by Adavanne et al. in [6], with some modifications. In a similar manner, it features a CRNN network using Gated Recurrent Units (GRU) as recurrent layers. This is followed by two parallel branches of Fully Connected (FC) layers, one for SED and one for DOA estimation, sharing weights along time dimension. The first FC layer of both branches uses linear activation, while the last FC layer of each branch uses a different activation function according to the task. The last FC layer in the SED branch contains 14 nodes using sigmoid activation (one node for each sound event class to be detected), while the last FC layer in the DOA branch consists of 42 nodes using tanh activation (each of the sound event classes is represented by 3 nodes relative to the 3-dimensional sound event location). As loss functions, binary cross-entropy is used for the SED branch and mean square error (MSE) loss for DOA estimation branch, keeping the two branches separated.

Regarding the differences respect to the baseline in this implementation, firstly, 2 CNN blocks have been added in order to help the network to learn more features, increasing the number of CNN blocks from 3 to 5. Each CRNN block consists of a convolutional layer with rectified linear unit (ReLU) activation, batch normalization to normalize the activation output, and MaxPooling along frequency axis to reduce

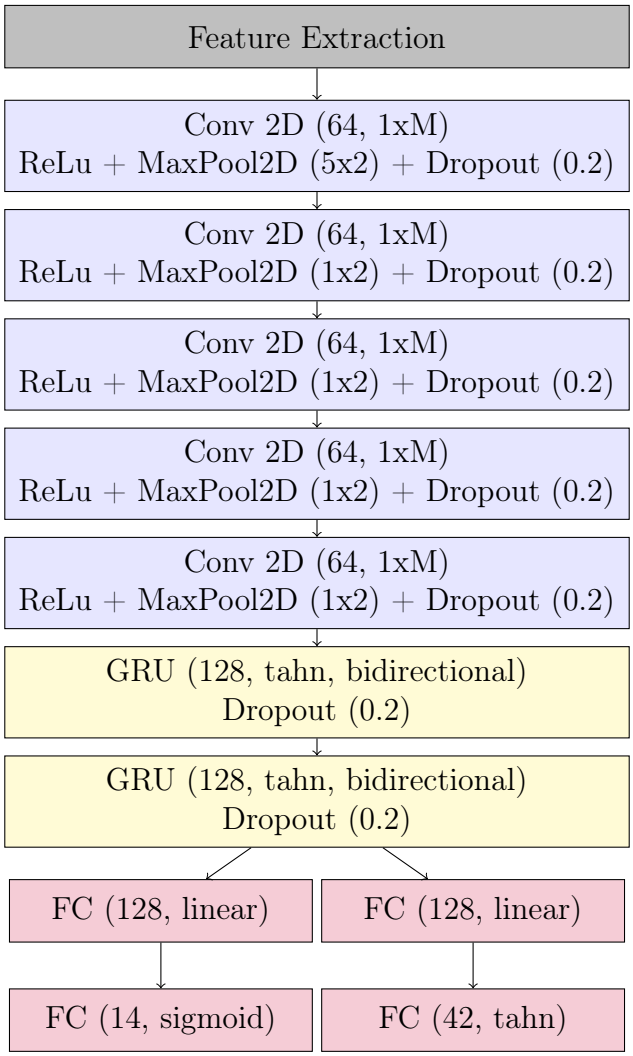

Figure 13: The proposed network architecture.

the dimensionality. Although adding layers to a neural network usually helps it to learn more features, it has the disadvantage of leading to possible overfitting, especially when the training dataset size is small as in this case. To prevent overfitting, each convolutional block uses a Dropout function, after reducing the dimensionality.

Secondly, rectangular filters are used instead of squared ones, mainly inspired by Pons et al. [63]. In the mentioned paper, the authors studied how filter shapes can help to proper model CNN motivated by musical aspects, achieving positive results in music classifications. Filters used in convolutional layers are principally inspired by image processing literature, typically being small and squared (usually being 3x3 or 5x5). Considering that, in the audio domain, filter dimensions corresponds to

time and frequency axes, wider filters may be capable of learning longer temporal dependencies in the audio, while higher filters may be capable of learning more frequency features. This study proposes the same concept, applying it to sound event detection. This system implementation uses rectangular filters of shape $1xM$, being 1 the time dimension and the M the frequency dimension. Setting the time dimension to 1 and increasing the frequency dimension, taking into consideration nearly all the mel-bands and the active frequencies in each of them for each frame, would hypothetically lead the network to better model the frequency dimension, helping the system to learn the presence or absence of an event, and consequently improving the metrics for location-aware detection. Such horizontal filters increase the reception field only on frequency dimension. Anyway, the temporal information is taken care by the recurrent GRU layers.

The last addition is the use of data augmentation as described in Section 3.4, expanding the number of DOA represented in it and consequently raising the scores related to classification-dependent localization metrics.

3.4 Data augmentation

With the aim of improving the results and to reduce system overfitting, the training dataset size has been increased using data augmentation based on channel rotations and reflection on the *xy* plane in the FOA domain.

Among others, this system implements the *16 patterns* technique proposed for the first time by Mazzon et al. in [12], with some small changes. This approach allows to increase DOA combination and correctly compute the corresponding ground truth DOA labels of the augmented data. Moreover, a relevant advantage of this method is the possibility to be applied regardless of the number of overlapping sound sources [64], which makes it an easy and straightforward data augmentation method. The data have been augmented following the transformations suggested in [64], considering only *channel swapping* and *channel sign inversion*. The proposed data manipulations correspond to rotations of 0, $-90°$, $+90°$, and $+180°$ of the az-

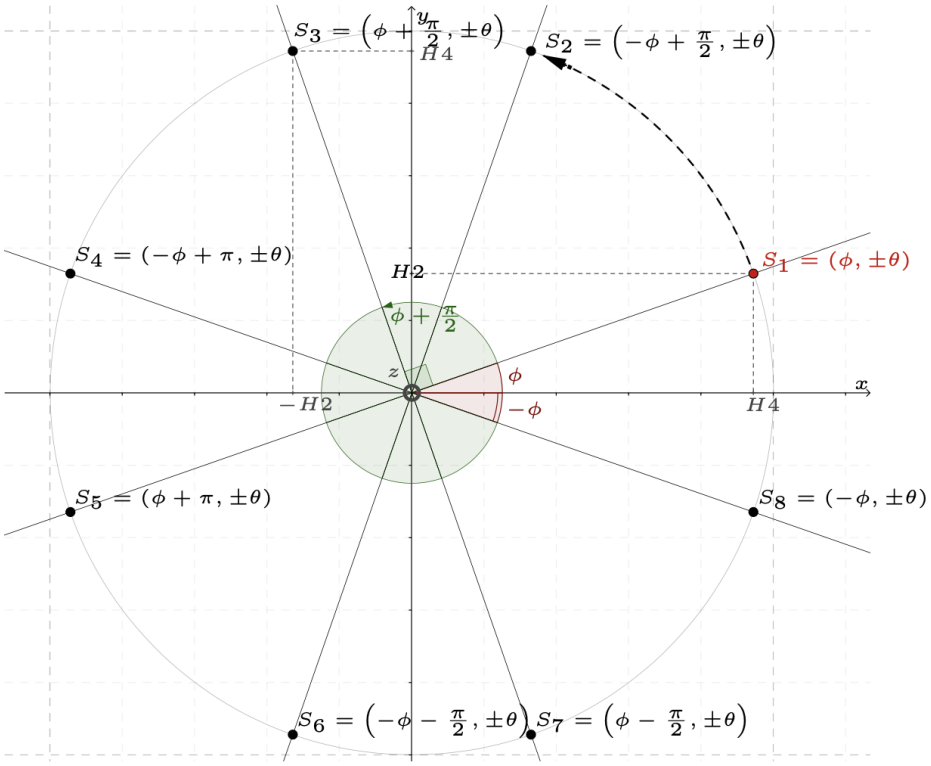

Figure 14: Augmented positions of a source S. Each position is represented with elevation coordinate θ and elevation coordinate $-\theta$. All the implemented translation of the azimuth angle ϕ and its negative $-\phi$ are also represented. Adapted from figure in [12].

imuth angle ϕ and its reflection with respect to the xz plane, leading to 8 rotations around the z axis, and a reflection with respect to the xy plane (considering the opposite elevation angle), for a total of 15 new patterns plus the original one. All the implemented augmented positions for a sound source S are illustrated in Figure 14.

Figure 15 shows an example of channel rotation on intensity vector, after applying a reflection with respect to xy plane. The figure shows the reflection of channel Z. The reader is referred to [64] for further details. In [12], Mazzon et al. compute the augmented dataset in time domain and extract the features offline for each of the augmented signals. All the possible transformations are computed offline, and the data generator randomly chooses one of them at each iteration. In this system only 15 patterns have been implemented, without considering the original

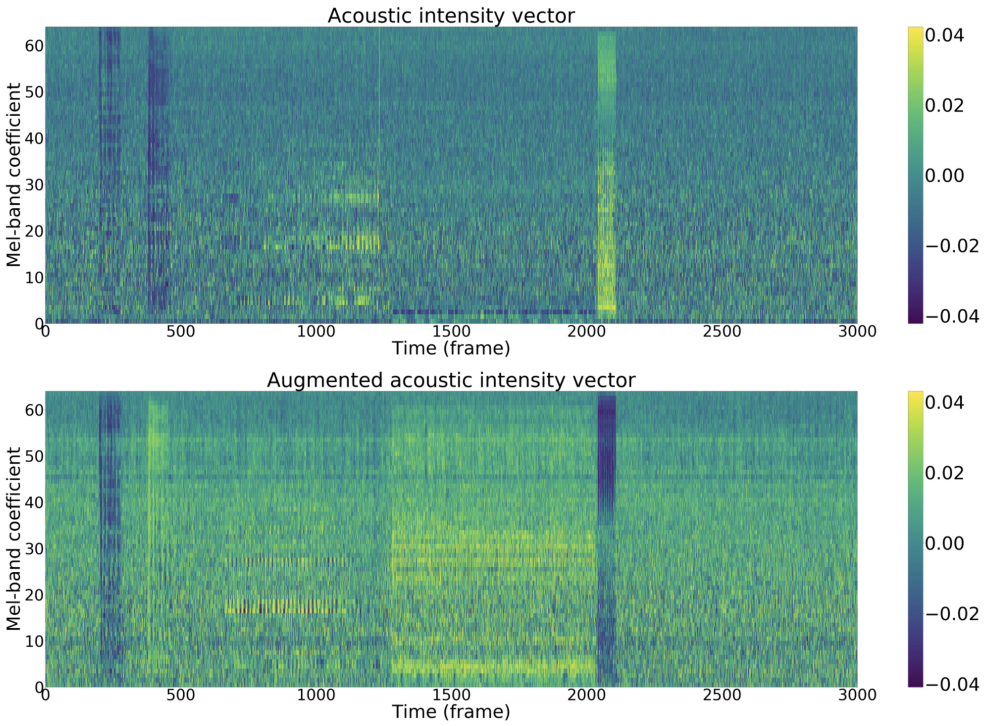

Figure 15: Example of augmented intensity vector. Pattern applied: reflection with respect to xy plane. The image shows the Z-component of the intensity vector, before and after applying a reflection on the Z-channel. The horizontal axis represents time (expressed in frames) and the vertical axis mel-band coefficients.

one as data augmentation pattern. Also in this case the data augmentation has been implemented offline, but, for memory reasons, instead of computing all the transformations for each audio file, only two out of the 15 patterns are randomly selected to augment the data during the feature extraction process. The pattern selected for a particular audio file is also used for augmenting the corresponding label. All the new generated files, together with the original ones, are used to train the network. In this case, only discrete angles of 90° and its multiple have been considered, mainly motivated by the paper [12] and the first analysis of the results. Random angle rotation, based on random rotation matrices will be focus of investigation in future work.

3.5 Hyper-parameters

All the dataset audio files are sampled at 24kHz. For the STFT, a 960 point Hanning window has been used with a 50% hop size. Considering that the temporal

resolution of the label is 100ms, the sub-frames of 20ms are interpolated as suggested in the baseline system. The number of mel-band filter is set to 64. The training development set has been increased from 400 to 1200 files using the aforementioned augmentation technique. Adam is used as optimization method [65]. A sound event is considered to be active, and its respective DOA estimation considered, if the SED output exceeds a threshold of 0.5. Across all the experiments, the batch size has been set to 128, and the systems have been trained for 50 epochs at most. An early stopping strategy has been implemented, stopping the training if the validation loss does not improve for 20 epochs.

3.6 Variations of submitted versions

Four different system outputs have been submitted to the DCASE Challenge 2020 Task 3. In fact, the DCASE Challenge committee admits up to four submissions as long as each each of them is coming from a different versions of the system. In this case, the submissions use at least different hyper-parameters of the network. The name of the submissions used in this report are the same used for the DCASE Challenge.

Submission *Ronchini_ UPF_ T3_ 1* uses dense rectangular filters of dimension 1x48, being 1 the time dimension and 48 the frequency dimension. The learning rate is constant and set to 0.001. Submission *Ronchini_ UPF_ T3_ 2* uses 1x48 rectangular filters, using AveragePooling instead of MaxPooling. The learning rate is constant and set to 0.001 for the first 40 epochs, while it is decreased by 0.95% every next epoch. In submission *Ronchini_ UPF_ T3_ 3*, convolutional layers use MaxPooling and rectangular filters of 1x50, with constant learning rate of 0.001. In the last submission, submission *Ronchini_ UPF_ T3_ 4*, the architecture is configured with the same hyper-parameters used in submission 1, using only 3 convolutional layers instead of 5.

The first three versions of the network have a size of 1 millions of parameters, while the network using only three convolutional layers has a size of 850k parameters.

Chapter 4

Experiment

Several architecture configurations, filter dimensions and data augmentation techniques have been explored before reaching the network architecture described in Section 3.3. The intention of this chapter is to described the progress of the study in order to reach the final system. The full experiment can be considered as the combination of two minor sub-experiments. The first one has concentrated on finding the rectangular filter shape which gives the best performance. The second has been focused on considering different data augmentation methods to select the most appropriate technique. The chapter is structured as follows: Section 4.1 describes the dataset used to train and evaluate the dataset, Section 4.2 and Section 4.3 explain the first and the second part of the experiment, respectively. Section 4.4 details the metrics used to evaluate the output of the system.

4.1 Dataset

The dataset used for the evaluation of the system is the one provided for the DCASE 2020 Challenge Task 3: TAU-NIGENS Spatial Sound Events 2020 [1] [66]. Sound event samples are sourced from the NIGENS General Sound Events Database[2]. This database provides 714 sound examples distributed among 14 classes such as

[1] https://zenodo.org/record/3740236
[2] https://zenodo.org/record/2535878

alarm, crying baby, crash, barking dog, running engine, burning fire, footsteps, knocking on door, female and male speech, female and male scream, ringing phone, and piano. The sound events of the dataset have been convolved with real spatial room impulse responses (RIRs) coming from 15 rooms of different shapes and types, such as lecture halls, large and small classrooms, meeting rooms, sports halls and open spaces in university buildings. The RIRs have been registered with a 32-channel spherical microphone array (SMA), the *em32 Eigenmike* [3]. The NIGENS dataset has been divided into 8 splits, 6 of which constitute the development set and 2 the evaluation set. The development dataset consists of 600 records divided in 6 predefined splits, while the evaluation dataset contains 200 records. In order to convolve the NIGENS sound events with the different RIRs, one or two rooms have been assigned to each split, and 100 mixtures of spatialized sound events were generated for each such combination of event samples and rooms. Each generated mixture is one minute long and each event could be static or dynamic. Ambient noise from the same room has been additionally mixed to the convolved sound events, with the signal-to-noise (SNR) levels between 30dB and 6dB. The dataset is presented in two spatial recording formats, MIC and FOA. This report only considered the FOA format. Further information regarding the dataset can be found at [66].

4.2 First experiment: filter shape

The first experiment has been concentrated in exploring different rectangular filter shapes in order to understand the filter's size which gives the best performance and helps the network to properly learn frequency features to correctly detect events. The alternative of analysing rectangular filter shapes has been mainly inspired and motivated by Pons et el. in [63]. In this article, the authors reach positive results using rectangular filter shape for music classification. This report proposes the same concept, applying it to sound event detection. Hypothetically, increasing the receptive field in the frequency dimension helps the network to better learn frequency properties. In fact, considering almost all the mel-bands for each time frame together

[3] `https://mhacoustics.com/products#eigenmike1`

with its frequencies content should help the network to learn when a particular event is active or not and to classify the event itself. In order to prove this assumption, the results of the proposed system using rectangular filters of shape $1xM$ have been compared with the results of the baseline system, which uses rectangular filters of shape 3x3. In this project have been tested filter size of 1x46, 1x48, 1x50, 1x52, 1x54, 1x56, and completely dense filters of dimension 1x64 (being 64 the size of mel-band filter). In particular, the experiment started testing the dense filter shape of 1x64 and 1x56, motivated by the results reached in [63]. Starting from the filter of shape 1x56, the frequency dimension has been decreased by two point each test until reaching the filter size which gave the best results.

During the experiment have been tested also shape of $2x48$ and $3x48$, in order to investigate if increasing the time dimension would facilitate the network to learn temporal information.

4.3 Second experiment: data augmentation

After selecting the filter which best performs, the second experiment considered different data augmentation techniques to increase the SELD score and to reduce overfitting. In this second step, the research has been concentrated on three data augmentation techniques: *time stretching*, *pitch shifting* and *channel rotations*. Each technique has been independently explored. In particular, pitch shifting is a data augmentation technique where the original pitch of a sound is raised or lowered. Time strecthing is intent as the process of changing the speed or duration of an audio signal without affecting its pitch. The two techniques have been implemented using the corresponding functions `time_stretch`[4] and `pitch_shift`[5] from librosa library. Regarding time stretching, a stretch factor between 0.9 and 1.1 has been considered. Regarding pitch shifting, the sound has been augmented in a range between -2 and 2 semitones, considering only integer values. The values of the stretch factor and the pitch shifting range have been thus selected in order to do

[4] `https://librosa.org/librosa/0.6.0/generated/librosa.effects.time_stretch.html`
[5] `https://librosa.org/librosa/0.6.0/generated/librosa.effects.pitch_shift.html`

not generate sound samples which would differ too much compared to the original ones. In fact, if the augmented sound samples are too different compared to the original ones, the system would probably fall into a underfitting situation, where it would not be possible to capture the dominant trend. For each data augmentation technique test, two values within the chosen ranges have been randomly selected for each audio file, generating two augmented samples for each sound file plus the original one, thus increasing the dataset from 400 to 1200 files. The third technique, channel rotation, is detailed in Section 3.4. The main advantage of this technique is the fact that, even creating new randomized augmented samples, the physical relations between channels are maintained. Channel rotation has been selected as the data augmentation technique to implement in the system.

4.4 Evaluation method

The network predictions have been evaluated considering the joint nature of localization and detection, as proposed in [13]. In particular, each task is evaluated considering two metrics. Regarding the SED task, the metrics ER_{20° and F_{20° are considered. They are location-dependent, which means that a prediction is considered true positive only if it is under a distance threshold of 20° from the reference. LE_{CD} (localization error) and LR_{CD} (localization recall) are related to DOA estimation, being classification-dependent, which means they are going to be calculated only between sounds which have been correctly classified. All metrics are computed in one-second non-overlapping frames. An ideal SELD method will have $ER_{20^\circ} = 0$ and $F_{20^\circ} = 100\%$ for localization-aware detection metrics and $LE_{\mathrm{CD}} = 0$ and $LR_{\mathrm{CD}} = 100\%$ for the class-aware localization metrics. Moreover, in order to give a unique score to the system, a *SELD score* has been defined, calculated as in formula 4.1

$$SELDscore = \frac{(SEDscore + DOAscore)}{2} \tag{4.1}$$

where

$$SEDscore = \frac{(ER + (1 - F))}{2} \tag{4.2}$$

$$DOAscore = \frac{\frac{DOAerror}{180} + (1 - framerecall))}{2} \tag{4.3}$$

An ideal SELD method will have an SELD score of zero.

The ER metrics is calculated as:

$$ER = \frac{D + I + S}{N} \tag{4.4}$$

where D = false negatives (also known as *Deletions*), I = false positives (also known as *Insertions*), S = substitution error (one true positive and one true negative appearing at the same time count as a single substitution error S) and N is the total number of reference events. The F (F-score) is calculated as follow:

$$F = \frac{2PR}{P + R} \tag{4.5}$$

where

$$P = \frac{TP}{TP + FP} \tag{4.6}$$

and

$$R = \frac{TP}{TP + FN} \tag{4.7}$$

where TP = True Positives, TN = True Negatives, FP = False Positives and TP = True Positive. The *DOA error* is calculated as the distance between the cartesian

position vector of the reference and the prediction, as shown in the formula 4.8.

$$d = \| x_{\text{ref}} - x_{\text{so}} \| \tag{4.8}$$

For more information about the evaluation metrics the author is refer to [13].

Chapter 5

Results and discussion

This chapter reports the results based on both the FOA development and the evaluation set of the dataset. Regarding the development dataset, the same specifications given in the DCASE Challenge related task description have been followed for the evaluation, using split 1 for testing, split 2 for evaluation and splits 3-6 for training. More information regarding the task specifications can be found at [8]. This chapter presents and discusses the results related to the development dataset in Section 5.1 while results and discussion based on the evaluation dataset are reported in Section 5.2, with a subsection dedicated to the DCASE results.

5.1 Results on Development Dataset

Tables 2, 3, 4 and 5 report the evaluation results on the development dataset of the first and second stage of the study. In particular, Table 2 and 3 show the evaluation results for the development dataset on the testing split using different rectangular filters shapes. Table 4 presents the performance of the system considering different training loss functions. Table 5 details the results on the same dataset, using 1x48 filters with different data augmentation methods. In all tables, the results are compared with the baseline system.

Table 2 reports the results related to the first sub-experiment. In order to select

Filter shape	ER_{20°	F_{20°	LE_{CD}	LR_{CD}	SELD
Baseline (3x3)	0.72	37.4%	22.8°	60.7%	0.47
1x46	0.69	40.1%	21.7°	62.3%	0.45
1x48	**0.69**	**40.3%**	**20.9°**	**62.4%**	**0.45**
1x50	**0.70**	**40.4%**	**21.1°**	**61.1%**	**0.45**
1x52	0.70	39.9%	21.6°	61.4%	0.45
1x54	0.72	37.1%	21.7°	59.7%	0.47
1x56	0.72	38.2%	21.2°	59.4%	0.47
1x64	0.72	38.1%	23.1°	61.4%	0.46

Table 2: Evaluation results on development set for first stage of the experiment using different filters shapes increasing only the frequency dimension.

Filter shape	ER_{20°	F_{20°	LE_{CD}	LR_{CD}	SELD
Baseline (3x3)	0.72	37.4%	22.8°	60.7%	0.47
1x48	**0.69**	**40.3%**	**20.9°**	**62.4%**	**0.45**
2x48	0.70	40.6%	20.8°	61.1%	0.45
3x48	0.75	36.0%	23.1°	58.6%	0.48

Table 3: Evaluation results on development set for first stage of the experiment using different filters shapes increasing only the time dimension.

the filters shapes which give the best results, several sizes has been tested. As it is possible to observe, the most desirable results are reached with filter shapes of 1x48 and 1x50. Almost all the tested rectangular shapes outperfom the baseline results. The only filter which does not improve the metrics is the 1x54, which is otherwise comparable to it. Between the two filter shapes which better perform, the 1x48 has been selected to continue the experiment with. In particular, the selection of the 1x48 filters instead of 1x50 is based on the fact that, while the difference on sound event detection metrics is negligible, localization-aware classification scores for the first shape are better than the second one. Table 3 reports the results of filters's size 2x48 and 3x48, which have been tested (after selecting 1x48 filter shape) to explore their relation with temporal information. As it possible to observe, increasing the receptive field also in time dimension does not help to improve the results. Probably this is due to the fact that the network is looking for patterns to recognize the onset of an event, which is usually distinguish by the high content of frequency at a particular frame. Presumably, considering that the high frequency content

Method	ER_{20°	F_{20°	LE_{CD}	LR_{CD}	SELD
Masked MSE	**0.69**	**40.3%**	**20.9°**	**62.4%**	**0.45**
MSE	0.75	37.50%	23.2°	60.8%	0.47

Table 4: Evaluation results on development set for the different training functions tested for the DOA estimation task. MSE stands for mean square error.

Method	ER_{20°	F_{20°	LE_{CD}	LR_{CD}	SELD
Baseline	0.72	37.4%	22.8°	60.7%	0.47
Time Stretching	0.86	22.8%	27.9°	51.0%	0.57
Pitch shifting	0.86	22.3%	28.7°	51.0%	0.57
Channel rotations	**0.59**	**50.6%**	**17.6°**	**66.2%**	**0.38**

Table 5: Evaluation results on development set for the second stage of the experiment using different data augmentation techniques. TS stands for time stretching, PS stands for pitch shifting and CR stands for channel rotations.

of an onset is instantaneous, increasing the time dimension of the receptive field does not help the network because all the useful frequency content information is contained in a single frame, so setting the time dimension to 1 is already sufficient for the system to properly learn how to detect an event. Comparing 2x48 and 1x48, the difference in terms of results is neglectable, while increasing the time dimension also increase the training time of the network and its size. For these reasons, the 1x48 has been selected as the best performant and as the final filter size to follow to study with. This confirms that using rectangular filters increasing the receptive field only in frequency dimension helps the network to better model frequency features. Moreover, it also highlight the importance of domain specific knowledge, understanding the training datasets in order to properly design the model architecture for a determined task.

Before moving to the second part of the experiment, the network have been evaluated using two different training loss functions for DOA evaluation, masked mean-square error, as proposed on the improved version of the SELDNet system [6] presented as baseline for the DCASE 2020 Challenge, and the mean-square error as proposed in [6]. This analysis had the purpose of confirming that using mean square error instead of masked mean square error gives betters results. As Table 4 shows, the results

confirm this hypothesis, so the mean square error has been selected as training loss function for the system. The training function test has been made on the system using filter of shape 1x48.

The second sub-experiment is based on the exploration of several data augmentation techniques with the aim to increase the score and prevent overfitting. The study has been focused on three methods: time stretching, pitch shifting and channel rotation. Results are shown in Table 5. As it possible to observe, channel rotation is the only method that outperforms the baseline results, substantially increasing location-aware detection scores. This might be explained by the fact that channel rotations maintain the physical relations between channels, making the manipulations of sound more realistic. Moreover, the noticeable improvement of SED-related scores (ER and $F - score$) could be explained by the joint nature of localization and detection of the evaluation metrics. In fact, the results suggest that channel rotation augmentation technique increases the DOA information, helping the network to better localize sound sources. It is possible that, without the expansion of the dataset size, the same sound source would have been considered as active, but the system would have been wrong in localizing it, considering the prediction wrong. By augmenting the data and by giving the network more DOA examples for training, the algorithm's results improve in a substantial way.

5.2 Results on Evaluation Dataset

Once the system has been evaluated on the development dataset and the final architecture has been defined, the final neural network has been evaluated on the evaluation dataset. The evaluation dataset has been provided without the ground truth so the predictions have been directly submitted to the challenge system and the results have been made available after 15 days. The network has been firtsly trained on the development dataset, using splits 2-6 as training splits and split 1 as validation split, while the evaluation dataset has been used as testing split. 15 teams participated to the challenge for a total of 43 submissions. The tables showing the results in this section only report the results related to the different versions of

Submission version	ER_{20°	F_{20°	LE_{CD}	LR_{CD}
Ronchini_UPF_T3_2 (28/43)	**0.58**	**50.8%**	**16.9°**	**65.5%**
Ronchini_UPF_T3_1 (29/43)	0.59	50.3%	16.8°	65.5%
Ronchini_UPF_T3_3 (32/43)	0.61	49.1%	16.7°	63.3%
Ronchini_UPF_T3_4 (33/43)	0.60	49.1%	17.1°	63.7%
DCASE2020_baseline (42/43)	0.70	39.5%	23.2°	63.1%

Table 6: DCASE results on evaluation dataset for the full system considering all the four versions submitted to the DCASE challenge and the general system ranking.

this particular project compared with the baseline. Table 6 shows the results based on the evaluation dataset, considering the full network architecture as presented in Section 3.3. The table reports the results of the four different versions submitted to the DCASE challenge, considering the overall system ranking. Details regarding the differences between the different submitted versions can be found in Section 3.6. The tables also report the rank of the system over the total submissions, written in parenthesis next to the name of the system version. For example, the version *Ronchini_ UPF_ T3_ 2* is the 28th out of 43 in the overall system ranking. The overall ranking is based on the cumulative rank of the four evaluation metrics described in Section 4.4, sorted in ascending order. If two systems ended up with the same cumulative rank, they are assumed to have equal place in the challenge, even though they will be listed alphabetically in the ranking tables. More information about the ranking system can be found at [8]. Tables 7 and 8 report the results related to the acoustic environment-wise performance, considering two unseen acoustic environments of the evaluation dataset. Tables 9 and 10 detail the results related to event polyphony-wise performance, considering different numbers of overlapping events of the evaluation dataset. In particular, Table 9 considers no overlapping, while Table 10 considers two overlapping sources. For more information regarding the challenge results and a wider overview considering all the submitted systems and the relative ranking the reader is refer to [67].

Submission version	ER_{20°	F_{20°	LE_{CD}	LR_{CD}
Ronchini_UPF_T3_2 (28/43)	**0.56**	**52.8%**	**15.4°**	**65.0%**
Ronchini_UPF_T3_1 (29/43)	0.59	50.4%	16.7°	64.7%
Ronchini_UPF_T3_3 (32/43)	0.62	48.5%	16.3°	61.1%
Ronchini_UPF_T3_4 (33/43)	0.57	52.4%	16.0°	65.0%
DCASE2020_baseline (42/43)	0.66	43.3%	20.5°	65.0%

Table 7: DCASE results on evaluation dataset for the full system regarding all the four versions submitted to the DCASE challenge, considering the first unseen location.

Submission version	ER_{20°	F_{20°	LE_{CD}	LR_{CD}
Ronchini_UPF_T3_2 (28/43)	**0.60**	**48.9%**	**18.3°**	**66.1%**
Ronchini_UPF_T3_1 (29/43)	0.60	50.2%	17.0°	66.3%
Ronchini_UPF_T3_3 (32/43)	0.60	49.7%	17.0°	65.4%
Ronchini_UPF_T3_4 (33/43)	0.64	45.9%	18.4°	62.5%
DCASE2020_baseline (42/43)	0.74	35.5%	26.2°	59.1%

Table 8: DCASE results on evaluation dataset for the full system regarding all the four versions submitted to the DCASE challenge, considering the second unseen location.

5.2.1 DCASE Challenge results discussion

As it possible to observe from the results coming from development and evaluation dataset, the results of all the different versions of the system are similar between each other. In fact, for all the different situations considered (the two unseen locations and the different number of overlapping sources), the metrics values do not differ too much from version to version. For this reason, the system can be considered stable and robust, also in cases of unseen and new situations. As Table 6 shows, the proposed system considerably outperforms the state-of-the-art method presented as baseline, significantly increasing the location-dependent metrics related to SED task. Table 7 and 8 prove that the system implemented in this project improves the localization and detection of sources when considering unseen location, reaching good results on both unknown locations. Table 9 demonstrates that the changes implemented in this network significantly enhances the baseline behavior when the source are not overlapped. Anyway, Table 10 shows that the proposed system has some shortage when the different sound sources are overlapped each other.

Submission version	ER_{20°	F_{20°	LE_{CD}	LR_{CD}
Ronchini_UPF_T3_2 (28/43)	**0.45**	**64.1%**	**12.2°**	**72.8%**
Ronchini_UPF_T3_1 (29/43)	0.47	63.7%	11.8°	72.0%
Ronchini_UPF_T3_3 (32/43)	0.52	59.8%	12.5°	68.9%
Ronchini_UPF_T3_4 (33/43)	0.49	61.6%	11.9°	69.3%
DCASE2020_baseline (42/43)	0.75	32.5%	26.7°	57.4%

Table 9: DCASE results on evaluation dataset for the full system regarding all the four versions submitted to the DCASE challenge, considering no overlapping sources.

Submission version	ER_{20°	F_{20°	LE_{CD}	LR_{CD}
Ronchini_UPF_T3_2 (28/43)	0.65	43.1%	20.2°	61.3%
Ronchini_UPF_T3_1 (29/43)	0.66	42.5%	20.2°	61.7%
Ronchini_UPF_T3_3 (32/43)	0.66	42.7%	19.5°	59.9%
Ronchini_UPF_T3_4 (33/43)	0.67	41.9%	20.7°	60.5%
DCASE2020_baseline (42/43)	**0.58**	**51.3%**	**18.3°**	**69.9%**

Table 10: DCASE results on evaluation dataset for the full system regarding all the four versions submitted to the DCASE challenge, considering two overlapping sources.

In overall, it is conspicuous that both location-dependent and classification-dependent metrics improve, a part for the case of two overlapping sources, which is the only case where the baseline performs better compared to the different versions of the system. The no overlapping sound sources situation is where it is possible to consider the main improvement of this system, with ER_{20° that decreases up to 30 points and LE_{CD} improves up to 15 points, while the F_{20° increases of more than 30 points and LR_{CD} of 7. It is clear that the system has some difficulties when the sources are overlapped, which will need to be better investigated in the future work. Between the different submitted versions, the one that better performs is the *Ronchini_UPF_T3_2*. This particular version uses AveragePooling instead of MaxPooling and implements an adaptive learning rate instead of a fixed one, as explained in Section 3.6. This version could have reached the best performance mainly due to the adaptive learning rate, which help the network to reduce the overfitting. Further investigation will be needed in the future work. The less performant version is the *Ronchini_UPF_T3_4*, which is the version using only three convolutional

layers. This confirms that increasing the receptive field of the network adding more convolutional layers helps the network to better learn spectral characteristics of sounds.

Anyway, it is clear that the modification proposed and integrated in the SELDNet system positively increase the performance of the system. Furthermore, results based on the evaluation dataset confirm the results coming from the development dataset. The use of rectangular filters, increasing the receptive field only in frequency dimension, helps the network to better model frequency features, showing off the importance of domain specific knowledge. Moreover, channel rotation augmentation technique increases the DOA information supports the network to better localize sound sources.

5.3 DCASE 2020 highest ranking

Figure 16 shows the results of the best performing system per submitting team, highlighting the position of our submission over the total rank. As it is possible to observe, the proposed system outperforms the results of the baseline system but it has been exceeded from several teams. In order to give an overall view to the reader and explain the reasons why the top ranking systems are able to reach such results, the three top ranked systems will be briefly described. For further information, the reader can find all the detailed results and the technical reports of all the submitted systems at [67].

Between the highest rank systems submitted to the DCASE Challenge 2020 can be cited [68], [69] and [70].

In [68], the ten authors proposed a solution consisting of data augmentation, network training, model ensemble and post-processing. To overcome the lack of training data, they use voice channel switching for MIC signals, to simulate the rotation relationship. The FOA data have been similarly processed. Besides, Wang et al . conduct multichannel data simulation to augment data size by estimating one sound event signal and four RIRs from each static sound event segment, aside form using a

technique where they cut the non overlapping recording and randomly add them in time to generate more overlapping files. They also used SpecAugment augmentation approach to perform time and frequency block masking on the spectrogram. After the pre-processing of the data, the authors used several Deep Neural Networks for the task. In particular, they use different combination of high-level feature representation modules and temporal context representation modules. Specifically, they trained ResNet-GRU, ResNet-TDNNF, Xception-GRU, and Xception-TDNNF. After training, they used the weighted mean of the outputs predicted by different DNN architectures to ensemble the results and generate the SELD estimation. As a last step, they post-process the data using an dynamic optimal threshold chosen for each sound event on the validation set. The presented system has 123 millions of parameters. More information regarding the system can be found at [68].

In [69], the authors proposed a two-step method. In particular, they used a CRNN-based network that uses log-mel spectrogram as input features for the sound event detection task, using different data augmnetation techniques to increase the dataset size. To estimate the DOAs, they proposed a single-source histogram algorithm. The single-source histogram finds all the time-frequency (TF) bins that contains energy from mostly one source. A TF bin is considered to be a single-source TF bin when it passes three tests: magnitude, onset, and coherence test. After all the single-source TF bins are found, the DOA at each bin is computed using the theoretical steering vector of the microphone array, which are discretized using the required resolution of azimuth and elevation angles. Subsequently, these DOAs are populated into two 1D histograms, one for azimuth, one for elevation. The SED predictions and the two histograms are given as input features to a Sequence Matching Network (SMN), which is a multiple-input multiple-output CRNN. They train different input lengths of the network and ensemble the different models into a SMN by averaging the SED and DOA outputs. The system counts 11 millions of parameters. The reader can find more information at [69].

In [70], Shimada et al. consider two systems: a unified training framework that uses an activity-coupled Cartesian DOA vector (ACCDOA) representation as a sin-

Rank	Submission Information		Evaluation dataset				
	Submission name	Technical Report	Best official system rank	Error Rate (20°)	F-score (20°)	Localization error (°)	Localization recall
1	Du_USTC_task3_4		1	0.20	84.9 %	6.0	88.5 %
2	Nguyen_NTU_task3_2		4	0.23	82.0 %	9.3	90.0 %
3	Shimada_SONY_task3_4		5	0.25	83.2 %	7.0	86.2 %
4	Cao_Surrey_task3_4		11	0.36	71.2 %	13.3	81.1 %
5	Park_ETRI_task3_4		13	0.43	65.2 %	16.8	81.9 %
6	Phan_QMUL_task3_3		15	0.49	61.7 %	15.2	72.4 %
7	PerezLopez_UPF_task3_2		16	0.51	60.1 %	12.4	65.1 %
8	Sampathkumar_TUC_task3_1		20	0.53	56.6 %	14.8	66.5 %
9	Patel_MST_task3_4		22	0.55	55.5 %	14.4	65.5 %
10	Ronchini_UPF_task3_2		28	0.58	50.8 %	16.9	65.5 %
11	Naranjo-Alcazar_VFY_task3_2		30	0.61	49.1 %	19.5	67.1 %
12	Song_LGE_task3_3		31	0.57	50.4 %	20.0	64.3 %
13	Tian_PKU_task3_1		36	0.64	47.6 %	24.5	67.5 %
14	Singla_SRIB_task3_2		38	0.88	18.0 %	53.4	66.2 %
15	DCASE2020_MIC_baseline		39	0.69	41.3 %	23.1	62.4 %

Figure 16: Table including only the best performing system per submitting team.

gle target that solve sound event localization (SEL) and sound event detection (SED) simultaneously, and a two-stage system, RD3Net, that first handles the SED and SEL tasks individually and later combines those results. To generalize the models, they also apply three data augmentation techniques: equalized mixture data augmentation (EMDA), rotation of first-order Ambisonic (FOA) signals, and multichannel extension of SpecAugment. They use multichannel amplitude spectrograms and inter-channel phase differences (IPDs) as frame-wise input features. To further improve the performance, they also conduct a post-processing rotating the FOA data, estimating the ACCDOA vectors, rotating the vectors back, and averaging the vectors of different rotation patterns. They considered eight rotations. Also this system counts 11 millions of parameters. More details can be found at [70].

Chapter 6

Conclusion and Future work

This study describes a system which has been implemented for sound event detection and localization task, which has been also submitted for the DCASE Challenge 2020 Task 3. The method is based on SELDNet presented by Adavanne et. al [6], with some improvements. The main changes regard data augmentation based on Ambisonic rotations, network architecture and training loss functions. The main contribution of the proposed system, beside an enhancements of the state-of-the-art system, is the usage of rectangular filters, confirming that increasing the receptive field only in frequency dimension helps the network to better model frequency features while GRU layers take care of the temporal information. Moreover, the results on development and evaluation dataset highlights the importance of modeling the network architecture according to the dataset used for training and the importance of domain specific knowledge. Data augmentation also helped to increase the evaluation score, especially ER_{20° and F_{20° related to the SED task. The proposed system considerably outperforms the state-of-the-art method presented as baseline, significantly increasing the location-dependent metrics related to SED task.

Future work will include further investigation of different data augmentation methods based on preserving physical relations between channels, such as random rotation matrices to rotate the acoustic scene of random angles (instead of only discrete angles), and exploring rectangular filter shapes performances in different dataset for-

mats. Among the future work there is also the study of different data augmentation techniques merged together and the investigation of a two-stage system based on the SELDNet alternative. Regarding the neural network, further investigation should be made on if and how the adaptive learning rate helps the system to improve the results, compared to the static learning rate, and what would be the difference, in terms of performance, between using an AveragePooling and a MaxPooling. Moreover, it will be necessary to investigate how to improve the results of the presented system in all different situations, specially for the overlapping sound cases.

List of Figures

List of Tables

Bibliography

[1] Butko, T., Pla, F. G., Segura, C., Nadeu, C. & Hernando, J. Two-source acoustic event detection and localization: Online implementation in a smart-room. In *2011 19th European Signal Processing Conference*, 1317–1321 (IEEE, 2011).

[2] Atrey, P. K., Maddage, N. C. & Kankanhalli, M. S. Audio based event detection for multimedia surveillance. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 5, V–V (IEEE, 2006).

[3] Busso, C. *et al.* Smart room: Participant and speaker localization and identification. In *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, vol. 2, ii–1117 (IEEE, 2005).

[4] Marques, T. A. *et al.* Estimating animal population density using passive acoustics. *Biological Reviews* **88**, 287–309 (2013).

[5] Hirvonen, T. Classification of spatial audio location and content using convolutional neural networks. In *Audio Engineering Society Convention 138* (Audio Engineering Society, 2015).

[6] Adavanne, S., Politis, A., Nikunen, J. & Virtanen, T. Sound event localization and detection of overlapping sources using convolutional recurrent neural networks. *IEEE Journal of Selected Topics in Signal Processing* **13**, 34–48 (2018).

[7] Detection and classification of acoustic scenes and events challenge 2019. `http://dcase.community/challenge2019/task-sound-event-localization-and-detection`.

[8] Detection and classification of acoustic scenes and events challenge 2020. `http://dcase.community/challenge2020/task-sound-event-localization-and-detection`.

[9] Kapka, S. & Lewandowski, M. Sound source detection, localization and classification using consecutive ensemble of crnn models. *arXiv preprint arXiv:1908.00766* (2019).

[10] Cao, Y. *et al.* Polyphonic sound event detection and localization using a two-stage strategy. *arXiv preprint arXiv:1905.00268* (2019).

[11] Zhang, J., Ding, W. & He, L. Data augmentation and prior knowledge-based regularization for sound event localization and detection. Tech. Rep., Tech. Report of Detection and Classification of Acoustic Scenes and Event (2019).

[12] Mazzon, L., Yasuda, M., Koizumi, Y. & Harada, N. Sound event localization and detection using foa domain spatial augmentation. In *Proc. of the 4th Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE)* (2019).

[13] Mesaros, A., Adavanne, S., Politis, A., Heittola, T. & Virtanen, T. Joint measurement of localization and detection of sound events. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)* (New Paltz, NY, 2019). Accepted.

[14] Ronchini, F., López, A. P. & Arteaga, D. Sound event localization and detection based on crnn using dense rectangular filters and channel rotation data augmentation. Tech. Rep., DCASE2020 Challenge (2020).

[15] Valin, J.-M., Michaud, F. & Rouat, J. Robust 3D Localization and Tracking of Sound Sources Using Beamforming and Particle Filtering (2016). URL `http://arxiv.org/abs/1604.01642http://dx.doi.org/10.1109/ICASSP.2006.1661100. 1604.01642`.

[16] Gerzon, M. A. Periphony: With-height sound reproduction. *Journal of the audio engineering society* **21**, 2–10 (1973).

[17] Arteaga, D. Introduction to ambisonics (2015).

[18] Wikipedia contributors. Ambisonics — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Ambisonics&oldid=951187447 (2020). [Online].

[19] Furness, R. K. Ambisonics-an overview. In *Audio Engineering Society Conference: 8th International Conference: The Sound of Audio* (Audio Engineering Society, 1990).

[20] Meyer, J. & Elko, G. W. Spherical microphone arrays for 3d sound recording. In *Audio signal processing for next-generation multimedia communication systems*, 67–89 (Springer, 2004).

[21] Carpentier, T. Normalization schemes in ambisonic: does it matter? In *Audio Engineering Society Convention 142* (Audio Engineering Society, 2017).

[22] Zotter, F. & Frank, M. *Ambisonics* (Springer, 2019).

[23] Wikipedia contributors. Microphone array — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Microphone_array&oldid=950431423 (2020).

[24] Wikipedia contributors. Legendre polynomials — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Legendre_polynomials&oldid=944863819 (2020).

[25] Wikipedia contributors. Bessel function — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Bessel_function&oldid=945793169 (2020).

[26] Livingstone, D. J. *Artificial neural networks: methods and applications* (Springer, 2008).

[27] Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning* (MIT Press, 2016). http://www.deeplearningbook.org.

[28] Wikipedia contributors. Artificial neural network — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Artificial_neural_network&oldid=952890338` (2020).

[29] Tao, S. Deep neural network ensembles. In *International Conference on Machine Learning, Optimization, and Data Science*, 1–12 (Springer, 2019).

[30] Wikipedia contributors. Convolutional neural network — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Convolutional_neural_network&oldid=953234856` (2020).

[31] Krizhevsky, A., Sutskever, I. & Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105 (2012).

[32] Wikipedia contributors. Recurrent neural network — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Recurrent_neural_network&oldid=951606758` (2020).

[33] Wikipedia contributors. Recurrent neural network — Wikipedia, the free encyclopedia. `http://www.jordipons.me/apps/teaching-materials/` (2020).

[34] Laden, B. & Keefe, D. H. The representation of pitch in a neural net model of chord classification. *Computer Music Journal* **13**, 12–26 (1989).

[35] Dannenberg, R. B., Thom, B. & Watson, D. A machine learning approach to musical style recognition (1997).

[36] Matityaho, B. & Furst, M. Neural network based model for classification of music type. In *Eighteenth Convention of Electrical and Electronics Engineers in Israel*, 4.3.4/1–4.3.4/5 (1995).

[37] Marolt, M., Kavcic, A., Privosnik, M. & Divjak, S. On detecting note onsets in piano music. In *11th IEEE Mediterranean Electrotechnical Conference (IEEE Cat. No. 02CH37379)*, 385–389 (IEEE, 2002).

[38] Lee, H., Pham, P., Largman, Y. & Ng, A. Y. Unsupervised feature learning for audio classification using convolutional deep belief networks. In *Advances in neural information processing systems*, 1096–1104 (2009).

[39] Mesaros, A., Heittola, T., Eronen, A. & Virtanen, T. Acoustic event detection in real life recordings. In *2010 18th European Signal Processing Conference*, 1267–1271 (IEEE, 2010).

[40] Cakir, E. Deep neural networks for sound event detection. *Tampere University Dissertations* **12** (2019).

[41] Adavanne, S., Parascandolo, G., Pertilä, P., Heittola, T. & Virtanen, T. Sound event detection in multichannel audio using spatial and harmonic features. *arXiv preprint arXiv:1706.02293* (2017).

[42] Parascandolo, G., Huttunen, H. & Virtanen, T. Recurrent neural networks for polyphonic sound event detection in real life recordings. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6440–6444 (IEEE, 2016).

[43] Zhang, H., McLoughlin, I. & Song, Y. Robust sound event recognition using convolutional neural networks. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 559–563 (IEEE, 2015).

[44] Phan, H., Hertel, L., Maass, M. & Mertins, A. Robust audio event recognition with 1-max pooling convolutional neural networks. *arXiv preprint arXiv:1604.06338* (2016).

[45] Cakır, E., Parascandolo, G., Heittola, T., Huttunen, H. & Virtanen, T. Convolutional recurrent neural networks for polyphonic sound event detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **25**, 1291–1303 (2017).

[46] Adavanne, S., Pertilä, P. & Virtanen, T. Sound event detection using spatial features and convolutional recurrent neural network. In *2017 IEEE In-*

ternational Conference on Acoustics, Speech and Signal Processing (ICASSP), 771–775 (IEEE, 2017).

[47] Lu, R. & Duan, Z. Bidirectional gru for sound event detection. *Detection and Classification of Acoustic Scenes and Events* (2017).

[48] Jeong, I.-Y., Lee, S., Han, Y. & Lee, K. Audio event detection using multiple-input convolutional neural network. *Detection and Classification of Acoustic Scenes and Events (DCASE)* (2017).

[49] Schmidt, R. Multiple emitter location and signal parameter estimation. *IEEE transactions on antennas and propagation* **34**, 276–280 (1986).

[50] Roy, R. & Kailath, T. Esprit-estimation of signal parameters via rotational invariance techniques. *IEEE Transactions on acoustics, speech, and signal processing* **37**, 984–995 (1989).

[51] Huang, Y., Benesty, J., Elko, G. W. & Mersereati, R. M. Real-time passive source localization: A practical linear-correction least-squares approach. *IEEE transactions on Speech and Audio Processing* **9**, 943–956 (2001).

[52] DiBiase, J. H. *A high-accuracy, low-latency technique for talker localization in reverberant environments using microphone arrays* (Brown University Providence, RI, 2000).

[53] Pulkki, V. Spatial sound reproduction with directional audio coding. *Journal of the Audio Engineering Society* **55**, 503–516 (2007).

[54] He, W., Motlicek, P. & Odobez, J.-M. Deep neural networks for multiple speaker detection and localization. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 74–79 (IEEE, 2018).

[55] Chakrabarty, S. & Habets, E. A. Multi-speaker localization using convolutional neural network trained with noise. *arXiv preprint arXiv:1712.04276* (2017).

[56] Hirvonen, T. Classification of spatial audio location and content using convolutional neural networks. In *Audio Engineering Society Convention 138* (Audio Engineering Society, 2015).

[57] Adavanne, S., Politis, A. & Virtanen, T. Direction of arrival estimation for multiple sound sources using convolutional recurrent neural network. In *2018 26th European Signal Processing Conference (EUSIPCO)*, 1462–1466 (IEEE, 2018).

[58] Ferguson, E. L., Williams, S. B. & Jin, C. T. Sound source localization in a multipath environment using convolutional neural networks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2386–2390 (IEEE, 2018).

[59] Vesperini, F., Vecchiotti, P., Principi, E., Squartini, S. & Piazza, F. A neural network based algorithm for speaker localization in a multi-room environment. In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, 1–6 (IEEE, 2016).

[60] Grobler, C., Kruger, C. P., Silva, B. J. & Hancke, G. P. Sound based localization and identification in industrial environments. In *IECON 2017-43rd Annual Conference of the IEEE Industrial Electronics Society*, 6119–6124 (IEEE, 2017).

[61] Lopatka, K., Kotus, J. & Czyzewski, A. Detection, classification and localization of acoustic events in the presence of background noise for acoustic surveillance of hazardous situations. *Multimedia Tools and Applications* **75**, 10407–10439 (2016).

[62] Stevens, S. S., Volkmann, J. & Newman, E. B. A scale for the measurement of the psychological magnitude pitch. *The Journal of the Acoustical Society of America* **8**, 185–190 (1937).

[63] Pons, J., Lidy, T. & Serra, X. Experimenting with musically motivated convolutional neural networks. In *2016 14th international workshop on content-based multimedia indexing (CBMI)*, 1–6 (IEEE, 2016).

[64] Mazzon, L., Koizumi, Y., Yasuda, M. & Harada, N. First order ambisonics domain spatial augmentation for dnn-based direction of arrival estimation. *arXiv preprint arXiv:1910.04388* (2019).

[65] Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[66] Politis, A., Adavanne, S. & Virtanen, T. A dataset of reverberant spatial sound scenes with moving sources for sound event localization and detection. *arXiv preprint arXiv:2006.01919* (2020).

[67] Detection and classification of acoustic scenes and events challenge 2020. `http://dcase.community/challenge2020/task-sound-event-localization-and-detection-results`.

[68] Wang, Q. *et al.* The ustc-iflytek system for sound event localization and detection of dcase2020 challenge. Tech. Rep., DCASE2020 Challenge (2020).

[69] Nguyen, T. N. T., Jones, D. L. & Gan, W. S. Dcase 2020 task 3: Ensemble of sequence matching networks for dynamic sound event localization, detection, and tracking. Tech. Rep., DCASE2020 Challenge (2020).

[70] Shimada, K., Takahashi, N., Takahashi, S. & Mitsufuji, Y. Sound event localization and detection using activity-coupled cartesian doa vector and rd3net. Tech. Rep., DCASE2020 Challenge (2020).