Bar-Ilan University

Speech Signal Processing Algorithms Utilizing the Time-Frequency Domain Sparsity With Deep Neural Networks

Hodaya Hammer

Submitted in partial fulfillment of the requirements for the

Master's Degree in the Faculty of Engineering, Bar-Ilan University

Ramat Gan, Israel 2020

This work was carried out under the supervision of Prof. Sharon Gannot,

Faculty of Engineering, Bar-Ilan University.

Acknowledgment

I would like to thank my supervisor Prof. Sharon Gannot for his extensive help throughout my master's degree. I was fortunate to have the guidance and advice from someone with so much knowledge and talent. Thanks to Prof. Gannot, I was given the opportunity to fully experience academia and be a part of this fascinating world. In addition, I would like to thank Shlomo E. Chazan for introducing me to this field and to Prof. Gannot's lab. For the help, supervision and availability throughout the past few years. I feel extremely lucky to have such supervisors who cared so much about my work, and helped me accomplish all that I have done the past couple of years.

A special and much needed thank you goes to my Sabi and Savta who taught me that knowledge and education are priceless, for being supportive and proud of the path that I've chosen and showering us with the abundance that enabled me to do this easily.

Following, I would like to thank my parents who raised me to be the person that I am with high ambitions and the desire to do my best, for teaching me the right values that enabled me to succeed in all aspects of my life. Thanks for believing in me, for supporting me in every path that I choose and specifically, throughout this degree and the previous degree, and for your ongoing help.

A huge thank you goes to my husband, JJ, who stood by my side and supported me every step of the way, for pushing me to grasp every opportunity that was given to me and to make the most out of this learning experience. Every success and triumph of mine was truly a triumph of his.

Finally, I would like to thank Itai and Benaya, my kids, for cheering me on throughout these years and for teaching me how to find the perfect harmony in my life.

Contents

Abbreviations

TF time-frequency

DNN deep neural network

STSA short-time spectral amplitude

LSA log spectral amplitude

OMLSA optimally modified log spectral amplitude

IMCRA improved minima controlled recursive averaging

MoG Mixture of Gaussians

MoG-DNN mixture of Gaussians-deep neural network

EM expectation-maximization

SPP speech presence probability

SPP speech presence probability

STFT short-time Fourier transform

DUET degenerate unmixing estimation technique

DOA direction of arrival

CSD concurrent speakers detector

BLSTM bidirectional long short-term memory

CNN convolutional neural network

MUSIC multiple signal classification

SRP-PHAT steered response power with phase transform

GCC generalized cross correlation

WDO W-disjoint orthogonality

FCN fully convolutional network

RIR room impulse response

w.r.t. with respect to

SOTA state-of-the-art

PSD power spectral density

DFT discrete Fourier transform

p.d.f. probability density function

r.v. random variable

ReLU rectified linear unit

SNR signal to noise ratio

PESQ perceptual evaluation of speech quality

NN-MM neural network mixture-maximum

KL-divergence Kullback-Leibler divergence

DDESS deep direction estimation for speech separation

RTF relative transfer function

iRTF instantaneous relative transfer function

WSJ1 Wall Street Journal 1

SIR signal to interfering ratio

BSS blind source separation

SDR signal to distortion ratio

TF-DOAnet time-frequency direction-of-arrival net

VAD voice activity detector

MAE mean absolute error

CMS-DOA CNN multi-speaker DOA

Abstract

In this thesis, we present three algorithms for speech signal processing problems. First, a mixture of Gaussians-deep neural network (MoG-DNN) algorithm is presented to address the single-microphone speech enhancement task. We combine between the generative Mixture of Gaussians (MoG) model and the discriminative deep neural network (DNN). The proposed algorithm consists of two phases, the training phase and the test phase. In the training phase, the clean speech power spectral density (PSD) is modeled as a MoG representing an unsupervised assortment of the speech signal. Following, the database is labeled to fit the given MoG. DNN is then trained to classify noisy time-frame features to one of the Gaussians from the already inferred MoG. Given the classification results, a speech presence probability (SPP) is obtained in the test phase. Using the SPP, soft spectral subtraction is then applied, while, simultaneously updating the noise statistics. The generative unsupervised MoG can be applied to any unknown database, in addition to preserving the speech spectral structure. Furthermore, the discriminative DNN maintains the continuity of the speech. Experimental study shows that the proposed algorithm produces higher objective measurements scores compared to other speech enhancement algorithms.

Next, we approach the multi-microphone speech separation task. We present an algorithm based on masking inferred from the speaker's direction of arrival (DOA). According to the W-disjoint orthogonality property of speech signals, each time-frequency (TF) bin is dominated by a single speaker. This TF bin can therefore be associated with a single DOA. In our procedure, we apply a DNN with a U-net architecture to infer the DOA of each TF bin from a concatenated set of the spectra of the microphone signals. Separation

is obtained by multiplying the reference microphone by the masks associated with the different DOAs. Our proposed deep direction estimation for speech separation (DDESS) method is inspired by the recent advances in deep clustering methods. Unlike already established methods that apply the clustering in a latent embedded space, in our approach the embedding is closely associated with the spatial information, as manifested by the different speakers' directions of arrival.

Last, we approach the multi-microphone speaker localization task. We present a DNN-based online multi-speaker localization algorithm. Following the W-disjoint orthogonality principle in the spectral domain, each TF bin is dominated by a single speaker, and hence, by a single DOA. A fully convolutional network is trained with instantaneous spatial features to estimate the DOA for each TF bin. The high resolution classification enables the network to accurately and simultaneously localize and track multiple speakers, both static and dynamic. Elaborated experimental study using both simulated and real-life recordings in static and dynamic scenarios, confirms that the proposed algorithm outperforms both classic and recent deep-learning-based algorithms.

Chapter 1

Introduction and Background

Time-frequency distributions have been used to provide high resolution representation in a large number of signal processing applications. By exploiting sparsity in the TF domain, it is possible to obtain solutions for many speech processing problems. In this thesis, we focus on three core problem in speech processing: speech enhancement, speech separation and speaker localization. We present algorithms that combine model-based and DNN-based approaches.

1.1 Single Microphone Speech Enhancement

Single microphone speech enhancement is a broadly researched problem. An overabundance of algorithms utilizing speech and noise characteristics can be found in the literature [1]. Even though many current devices are equipped with multiple microphones, there are still many applications that require only a single microphone.

The short-time spectral amplitude (STSA) estimator and log spectral amplitude (LSA) estimator [2, 3] are widely-used model-based algorithms. The optimally modified log spectral amplitude (OMLSA) and particularly, the improved minima controlled recursive averaging (IMCRA) noise estimator are customized specifically for non-stationary noise environments [4, 5]. Still, fast changes in the noise statistics often yield the *musical noise* phenomenon.

In recent years, DNN techniques gained a lot of popularity due to the theoretical and algorithmic progress, in addition to the availability of more data and stronger processing power. Methods such as [6, 7] try to train a DNN to find a binary mask, which classifies the time-frequency frames into speech/noise classes. Given this mask, the noisy bins are reduced. These approaches use neither models nor assumptions for their speech enhancement. Nonetheless, they are trained with specific noise types and when used in an untrained noise environment, the enhancement is poorly executed. A comprehensive overview of the DNN-based algorithms can be found in [8]. Although these approaches demonstrate good results, we believe that the model-based approaches should not be neglected and that their advantages should be utilized alongside with the advantages of data-driven models.

Training-based algorithms, such as MixMax [9], were proposed to solve the speech enhancement problem. More recently, a combination between model-based and data-driven approaches was proposed in [10] in order to take advantage of the discriminative quality of DNNs. These algorithms consist of two phases, the training phase and the test phase. In [9], the *posteriori* distribution was calculated using the Bayes' rule. To improve the enhancement, in [10] the clean speech was modeled as a phoneme-based MoG, and the *posteriori* distribution was calculated using a phoneme-classifier DNN. The classification using the Bayes' rule is not always beneficial, resulting in poor enhancement. On the other hand, the phoneme-classifier, although showing good results, requires a phoneme labeled database, which is not always available. Additionally, the phonemes-based model might not be the best model for the speech enhancement task.

In this thesis, we present a MoG-DNN algorithm for speech enhancement. As in [9] the clean speech is modeled with a MoG and the noise is modeled with a single Gaussian. Following the maximization approximation [11], the *posteriori* distribution of the MoG is required. In the training phase, we first apply the expectation-maximization (EM) algorithm to estimate the MoG parameters. A noisy database is then built for training a DNN to classify noisy segments into their associate EM prior distributions over the MoG. In the test phase, using the results of the DNN, an SPP is calculated and soft spectral attenuation is applied to the noisy signal. Simultaneously, the noise statistics are updated. The supervised DNN is trained with the labels obtained with the unsupervised EM algorithm. In the proposed algorithm, we utilize the results of unsupervised training-based algorithms as a supervision of the DNN. This way, we circumvent the unlabeled database problem. Furthermore, we show that the proposed algorithm even outperforms the neural network mixture-maximum (NN-MM) algorithm [10] which uses a phoneme-labeled database.

Note that in [12], similar work was presented where a DNN was used in order to choose the correct Gaussian within the given MoG. The purpose of using the DNN here was solely for time and complexity reasons.

1.2 Multi-Microphone Static Speaker Separation

The next speech processing task that we targeted was audio and speech source separation. This is an active research field for the past two decades. A comprehensive survey of single- and multi-microphone approaches can be found in [13, 14, 15] and will hence not be explored here. We rather focus on learning-based approaches, most notably those using DNN.

Most single microphone approaches utilize masking operation. Similarly to the masking described in Sec. 1.1, masking for speech separation involves clustering of TF bins to the various speakers in the scene, and a multiplication of the noisy spectrogram by ‘1’ in TF bins clustered to the desired speaker, and ‘0’ otherwise. The underlying assumption of these masking algorithm is the W-disjoint orthogonality principle introduced in [16, 17], stating that each TF bin is dominated by a single speaker, at least if the number of speakers is small enough.

Recently, deep clustering approach was introduced for single-microphone speaker separation [18, 19]. In this approach, an embedding from the high-dimensional short-time Fourier transform (STFT) representation of the speech to a low-dimensional latent space was first inferred, followed by a clustering operation in the latent space. Another approach, which uses permutation invariant training (PIT) was presented in [20]. Both these approaches had a dramatic impact on the single-microphone speech separation field. Yet, as they only exploit spectral information, their performance deteriorates in the presence of high reverberation, or when the speakers are characterized by similar spectral patterns. In many cases, the outcome of these algorithms is characterized by *musical-noise* artifacts.

Spatial information, namely the attenuation and the time-delay between each of the sources' positions and a microphone pair, were utilized to estimate the separation mask in the degenerate unmixing estimation technique (DUET) approach [21]. Other multichannel separation algorithms are utilizing the single-channel deep clustering approach for estimating the building-blocks of the beamformer, specifically its steering vector [22, 23]. These approaches combine the advantages of the TF clustering operation, with the low distortion characteristics of the linear spatial processing that substitutes the masking operation. Other works train DNNs in order to estimate spectral masks. In [24] a DNN is applied to spatial features to infer a DOA-based mask, which is then used as a post-filtering stage at the output of a delay-and-sum beamformer. In [25] a group of DNNs, each applied in a different frequency band, is trained to predict a mask from spatial features. This information is then aggregated to generate a soft mask which is used for the final speech separation. In [26] an unsupervised deep clustering approach was applied to multiple mixtures of sources in a training stage. The trained DNN was then applied to the test mixture to predict the separating masks. In [27], a single-channel deep clustering network was trained in a supervised manner, where the supervision was obtained by a multichannel segmentation network.

Other approaches combining DNNs and beam-forming are presented in [28, 29]. In these methods, a concurrent speakers detector (CSD) is implemented to distinguish between noise-only frames, single-speaker frames and concurrently active speakers frames. In the first two classes, the noise spatial correlation matrix and the steering vectors are estimated, respectively. In the third class, the beamformer weights are not updated.

The deep clustering framework was extended to the multichannel setting in [30]. Spatial information was augmented with the spectral cues to form an input feature to the bidirectional long short-term memory (BLSTM) deep clustering network. The separation in this approach is still applied by single-channel masking using the clustering in the embedded latent domain.

In the current contribution, we are presenting a U-net architecture to address the speech separation task. It is assumed that the speakers are in different DOAs in the room. Consequently, rather than inferring a latent embedded domain, we utilize the DOA as the supervision of our network. Motivated by the great success of the U-net architecture in the computer vision field [31], and the high performance of the convolutional neural networks (CNNs) in estimation the DOA of multi speakers in noisy and reverberant environments [32], we train a U-net to classify each TF bin of the multichannel STFT image to one of the DOA candidates. The performance of the proposed schemes is demonstrated using recorded acoustic channels, while training is carried out using simulated data.

1.3 Multi-microphone Dynamic Source Localization

Consequent to the good results achieved in the speech separation task using the DOA as the supervisor, we changed our focus to refine and perfect the DOA of the speakers in order to achieve source localization.

Locating multiple sound sources recorded with a microphone array in an acoustic environment is an essential component in various cases such as source separation and scene analysis. The relative location of a sound source with respect to a microphone array is generally given in the term of the DOA of the sound wave originating from that location. DOA estimation and tracking are generating interest lately, due to the need for far-field enhancement and recognition in smart home devices. In real-life environments, sound sources are captured by the microphones together with acoustic reverberation. While propagating in an acoustic enclosure, the sound wave undergoes reflections from the room facets and from various objects. These reflections deteriorate speech quality and, in extreme cases, its intelligibility. Furthermore, reverberation increases the time dependency between speech frames, making source DOA estimation a very challenging task.

A plethora of classic signal processing-based approaches have been proposed throughout the years for the task of broadband DOA estimation. The multiple signal classification (MUSIC) algorithm [33] applies a subspace method that was later adapted to the challenges of speech processing in [34]. The steered response power with phase transform (SRP-PHAT) algorithm [35] uses generalizations of cross-correlation methods for DOA estimation. These methods are still widely used. However, in high reverberation enclosures, their performance is not satisfactory.

Supervised learning methods encompass an advantage for this task since they are data-driven. Deep-learning methods can be trained to find the DOA in different acoustic conditions. Moreover, if a network is trained using rooms with different acoustic conditions and multiple noise types, it can be made robust against noise and reverberation even for rooms which were not in the training set. Deep learning methods have recently been proposed for sound source localization. In [36, 37] simple feed-forward DNNs were trained using generalized cross correlation (GCC)-based audio features, demonstrating improved performance as compared with classic approaches. Yet, this method is mainly designed to deal with a single sound source at a time. In [38] the authors trained a DNN for multi-speaker DOA estimation. In high reverberation conditions, however, their performance is not satisfactory. In [39, 40] time domain features were used and they have shown performance improvement in highly-reverberant enclosures. In [41], a CNN based classification method was applied in the STFT domain for broadband DOA estimation, assuming that only a single speaker is active per time frame. The phase component of the STFT coefficients of the input signal were directly provided as input to the CNN. This work was extended in [32] to estimate multiple speakers' DOAs, and has shown high DOA classification performances. In this approach, the DOA is estimated for each frame independently. The main drawback of most DNN-based approaches, however, is that they only use low-resolution supervision, namely only time frame or even utterance-based labels. In speech signals, however, each time-frequency bin is dominated by a single speaker, a property referred to as W-disjoint orthogonality (WDO) [16] as mentioned in section 1.1. Adopting this model results in higher resolution, which might be beneficial for the task at hand. This model was also utilized in chapter 3 for speech separation where the authors recast the separation problem as a DOA classification at the TF domain. A fully convolutional network (FCN) was trained using spatial features to infer the DOA at every TF bin. Although the DOA resolution was relatively low, it was sufficient for the separation task at low reverberation conditions. When applying this method in high-reverberation enclosures or to separate adjacent speakers, a performance degradation was observed.

In this work, we present a multi-speaker DOA estimation algorithm. According to the WDO property of speech signals [16, 17], each TF bin is dominated by (at most) a single speaker. This TF bin can therefore be associated with a single DOA. We use instantaneous spatial cues from the microphone signals. These features are used to train a FCN to infer the DOA of each TF bin. The FCN is trained to address various reverberation conditions. The TF-based classification facilitates the tracking ability for multiple moving speakers. In addition, unlike many other supervised domains, the DOA domain lacks a standard benchmark. The LOCATA dataset [42] was recorded in one room with relatively low reverberation (RT$_{60} = 0.55$). Furthermore, a training dataset with high TF labels is not publicly available. Therefore, we generated training and test datasets simulating various real-life scenarios. We tested the proposed method on simulated data, using publicly available room impulse responses (RIRs) recorded in a real room [43], as well as real-life experiments. We show that the proposed algorithm significantly outperforms state-of-the-art competing methods.

This is a high resolution TF-based approach that improves DOA estimation performances with respect to (w.r.t.) the state-of-the-art (SOTA) approaches, which are frame-based, and enables simultaneous tracking of multiple moving speakers.

	Chapter	Where published
MoG-DNN	2	[44]
DDESS	3	[45]
TF-DOAnet	4	Submitted

Table 1.1: Summary of the algorithms proposed in this thesis.

Chapter 2

Speech Enhancement With DNNs Using MoG Based Labels

In this chapter we present a MoG-DNN algorithm for single-microphone speech enhancement. We combine between the generative MoG model and the discriminative DNN. The proposed algorithm consists of two phases, the training phase and the test phase. In the training phase, the clean speech PSD is modeled as a MoG representing an unsupervised assortment of the speech signal. Following, the database is labeled to fit the given MoG. DNN is then trained to classify noisy time-frame features to one of the Gaussians from the already inferred MoG. Given the classification results, a SPP is obtained in the test phase. Using the SPP, soft spectral subtraction is then applied, while, simultaneously updating the noise statistics. The generative unsupervised MoG can be applied to any unknown database, in addition to preserving the speech spectral structure. Furthermore, the discriminative DNN maintains the continuity of the speech. Experimental study shows that the proposed algorithm produces higher objective measurements scores compared to other speech enhancement algorithms.

2.1 Problem formulation and probabilistic modeling

In this section, we present a generative model of the noisy speech signal.

2.1.1 Maximization approximation

Let $z(t) = x(t) + y(t)$ be the observed noisy speech at time t, where $x(t)$ and $y(t)$ denote the speech and noise signals, respectively.

Let $Z(n, k)$ denote the STFT of $z(t)$, with n the frame index and $k = 0, \ldots, L - 1$ the frequency index. The frame length is set to L with an overlap of 75% between two successive frames. The frame index n is henceforth omitted for brevity, whenever applicable. Denote the $L/2 + 1$ dimensional log-spectrum vector $\mathbf{z}$, with the k-th frequency component, z_k defined by:

$$z_k = \log |Z(k)| = \log |Z(e^{j2\pi k/L})|, \; k = 0, \ldots, L/2. \tag{2.1}$$

Note that $z_k, k = L/2 + 1, \ldots, L - 1$ may be obtained by the symmetry of the discrete Fourier transform (DFT), i.e., $z_k = z_{L-k}$. Similarly, we define $\mathbf{x}$ and $\mathbf{y}$ to be the log-spectral vectors of the clean speech signal, x_k and the noise signal, y_k, respectively.

Following Nádas et al. [11], the noisy log-spectrum vector can be approximated by:

$$\mathbf{z} \approx \max(\mathbf{x}, \mathbf{y}) \tag{2.2}$$

such that the maximization is component-wise over the elements of $\mathbf{x}$ and $\mathbf{y}$. This approximation was found useful for recognition [11], speech enhancement [9, 46, 10] and speech separation tasks [47, 48]. In speech enhancement tasks only the noisy signal $\mathbf{z}$ is observed, and the aim is to estimate the clean speech $\mathbf{x}$.

2.1.2 Clean speech model - MoG

A speech utterance can be described as a time-series of phonemes, as utilized in [10]. In our study, since we aim to find a general speech model that is not based on prior knowledge of the spoken language, we give this observation a probabilistic description. Specifically, the log-spectral vector of the clean speech signal $\mathbf{x}$, is modeled by a MoG distribution, where similar to [9], we use unsupervised clustering of the speech frames. Based on the MoG model, the probability density function (p.d.f.) $f(\mathbf{x})$ of the clean speech $\mathbf{x}$ [for simplicity, we avoid the more accurate notation, $f_{\mathbf{x}}(\mathbf{x})$], can be written as

$$f(\mathbf{x}) = \sum_{i=1}^{M} c_i f_i(\mathbf{x}) = \sum_{i=1}^{M} c_i \prod_k \frac{1}{\sqrt{2\pi}\sigma_{i,k}} e^{-\frac{(x_k-\mu_{i,k})^2}{2\sigma_{i,k}^2}} \tag{2.3}$$

where M is the number of mixture components.

Let I be the class (mixture) random variable (r.v.) linked to the MoG r.v. $\mathbf{x}$. The term $f_i(\mathbf{x})$ is the Gaussian p.d.f. of $\mathbf{x}$ given that $I = i$. The scalar c_i is the probability of the i-th mixture and $\mu_{i,k}$ and $\sigma_{i,k}$ are the mean and standard deviation of the k-th entry of the i-th mixture Gaussian, respectively. We neglect the residual correlation between the frequency bins due to the Fourier transform properties. Since for each class $I = i$ the r.v. $\mathbf{x}$ is a Gaussian, we use a simplified modeling of the clean speech signal by using a MoG with diagonal covariance matrices. Although this is not a precise model of the speech, it has an advantage of a robust modeling that circumvents the need for large matrices inversion.

2.1.3 Noise model

As previously defined, $\mathbf{y}$ denotes the log-spectral vector of the noise signal, and let $g(\mathbf{y})$ denote the p.d.f. of $\mathbf{y}$. We assume that the components of $\mathbf{y}$ are statistically independent. For simplicity, we also assume that $g(\mathbf{y})$ can be modeled by a single Gaussian, with diagonal covariance i.e.,

$$g(\mathbf{y}) = \prod_k \frac{1}{\sqrt{2\pi}\sigma_{y,k}} e^{-\frac{(y_k-\mu_{y,k})^2}{2\sigma_{y,k}^2}}. \tag{2.4}$$

In order to obtain the mean and covariance of the noise, we use the assumption that the first 200 msec of any given audio signal are solely noise. Using this assumption, we define the noise parameters based only on the beginning of the utterance. Updating the noise parameters throughout the utterance is required in order to generalize this problem to non-stationary noises. The noise statistics are updated as presented in [10].

2.1.4 Noisy speech model

The following generative model, nicknamed MixMax [9, 11], is based on the maximum assumption and on the modeling of the clean speech signal as a Gaussian mixture p.d.f. and the noisy speech is modeled as the maximum between the clean speech and the noise signal.

Let $F_{i,k}(x)$, $G_k(y)$ denote the cumulative distribution functions of $f_{i,k}(x)$ and $g_k(y)$, respectively. Using the maximum assumption as shown in (2.2), it can be verified [11] that the p.d.f. of $\mathbf{z}$ is given by the following mixture model:

$$h(\mathbf{z}) = \sum_{i=1}^{M} c_i h_i(\mathbf{z}) = \sum_{i=1}^{M} c_i \prod_{k} \left(f_{i,k}(z_k) G_k(z_k) + F_{i,k}(z_k) g_k(z_k) \right) \tag{2.5}$$

where $h_i(\mathbf{z})$ is the p.d.f. of $\mathbf{z}$ given that $I = i$.

2.2 Application to speech enhancement

In this section, we describe the speech presence probability (SPP) and how it is utilized for speech enhancement.

2.2.1 The speech presence probability

Define the SPP $\rho_k \in [0, 1]$ as the conditional probability, given the noisy speech vector $\mathbf{z}$, that the k-th frequency component is dominated by the clean speech, and not by the noise.

$$\rho_k = \sum_{i=1}^{M} \rho_{i,k} \cdot p(I = i|\mathbf{z}) = p(x_k > y_k|\mathbf{z}) \tag{2.6}$$

such that

$$\rho_{i,k} = p(y_k < x_k | z_k, I = i) = \frac{f_{i,k}(z_k) G_k(z_k)}{h_{i,k}(z_k)} \tag{2.7}$$

and the posterior probability $p(I = i|\mathbf{z})$ can be obtained from (2.5) using the Bayes' rule:

$$p_i \triangleq p(I = i|\mathbf{z}) = \frac{c_i h_i(\mathbf{z})}{h(\mathbf{z})}. \tag{2.8}$$

Given an SPP, the k-th component of the log-spectrum of the clean speech $\hat{x}_k$, is estimated using soft attenuation:

$$\hat{x}_k = z_k - (1 - \rho_k) \cdot \beta \tag{2.9}$$

where β is the noise attenuation level (in the $\log$ domain).

Respectively, in vector form:

$$\hat{\mathbf{x}} = \mathbf{z} - (\mathbf{1} - \boldsymbol{\rho}) \cdot \beta \tag{2.10}$$

where $\mathbf{1}$ is a vector of ones with the same dimensions as $\boldsymbol{\rho}$, the vector concatenation of $\rho_k,\ k = 0, 1, \ldots, L/2$.

The observed noisy phase is used for reconstructing the time-domain speech signal, similarly to most speech enhancement algorithms. In this work we aim to find an accurate estimate of $\boldsymbol{\rho}$, using a special purpose neural network architecture.

2.2.2 Hybrid approach

In [10] the following changes were made in order to improve enhancement. First, the MoG (2.3) was generated using a phoneme labeled database. Then, (2.8) was replaced by a phoneme classifier DNN.

2.3 Generating a Supervised Database

The changes made in [10], improved the results. However, a labeled database was required, which is not always available. Additionally, the phoneme distribution of the speech may not be the natural distribution for the task of speech enhancement.

In this section, we present a new approach to overcome the mentioned above obstacles. The main idea of the proposed approach is (i) first apply the unsupervised EM algorithm on clean speech samples to model the speech with a MoG. (ii) A supervision is then generated to a synthetic unsupervised noisy database, which will be utilized to train a DNN. The DNN task is to substitute the posterior probability (2.8). The rest of the enhancement is as described in Sec. 2.2.

2.3.1 Training the MoG model and generating labels

The calculation of the SPP ρ_k, (2.6) involves two terms, $\rho_{i,k}$ which is computed using (2.7) and the posterior probability $p(I = i|\mathbf{z})$. Similar to [10] we want to make use of the high performance of the DNN in classification tasks in order to find a more accurate posterior probability. Yet, we intend to maintain the ability to use this method for any unlabeled database. Therefore, our goal is to generate labels for the given data in order to train a DNN.

First, we construct a MoG to model the clean speech (2.3). Let the training data consist of N log-spectrum frames, where $\overline{\mathbf{x}} = (\overline{\mathbf{x}}^1, \ldots, \overline{\mathbf{x}}^N)$ and $\overline{\mathbf{y}} = (\overline{\mathbf{y}}^1, \ldots, \overline{\mathbf{y}}^N)$ denote the clean speech dataset and the noise dataset, respectively. Additionally, let $\overline{\mathbf{z}}$ denote the training dataset of the simulated noisy signal at frame n, and let the training data consist of N log-spectrum frames.

We aim to estimate c_i, $\mu_{i,k}$ and $\sigma_{i,k}$ so as to maximize the log-likelihood

$$\log f(\overline{\mathbf{x}}) = \sum_{n=1}^{N} \log f(\overline{\mathbf{x}}^n).$$

In order to do so, we apply the EM algorithm [49]. Let γ_i^n be defined by:

$$\gamma_i^n = c_i f(\overline{\mathbf{x}}^n | I^n = i) \tag{2.11}$$

and α_i^n be defined by:

$$\alpha_i^n = Pr(I^n = i | \overline{\mathbf{x}}^n) = \frac{\gamma_i^n}{\sum_{i'=1}^{M} \gamma_{i'}^n} \tag{2.12}$$

where c_i, $\mu_{i,k}$ and $\sigma_{i,k}^2$ denote the current values of the model parameters. The EM iteration for this problem are described in [9].

After the final EM iteration we obtain the following estimated parameters: 1) the mean vectors $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_M$, where $\boldsymbol{\mu}_i = [\mu_{i,0}, \ldots, \mu_{i,L/2}]^\top$; 2) the covariance elements $\sigma_{i,k}^2;\ i = 1, \ldots, M;\ k = 0, \ldots, L/2$ and 3) the MoG weights $c_1, \ldots, c_M$.

In order to create the label vector ℓ^n for every given noisy vector $\overline{\mathbf{z}}^n$, we find which of the M Gaussians is the most probable for the respective clean speech training vector $\overline{\mathbf{x}}^n$ as follows:

$$\ell^n = \operatorname*{argmax}_i \{\alpha_i^n\}. \tag{2.13}$$

Define $\boldsymbol{l}^n$ to be the 'one-hot' encoded label vector corresponding to ℓ^n.

2.3.2 Deep neural network for Gaussian classification

In our approach, we substitute (2.8) with a DNN. The DNN is trained using a labeled noisy speech database. We use the same given database of vectors that was used for the EM algorithm $\overline{\mathbf{x}}^1, \ldots, \overline{\mathbf{x}}^N$, label it with the corresponding labels, $\ell^1, \ldots, \ell^N$ and then contaminate it with noise, resulting in the noisy training database $\overline{\mathbf{z}}^1, \ldots, \overline{\mathbf{z}}^N$.

In order to preserve the smoothness of the speech, we add context frames prior to training the DNN (4 frames from the future and 4 from the past were added to the current frame as proposed in [50]). Let $\overline{\mathbf{v}}^1, \ldots, \overline{\mathbf{v}}^N$ denote the vectors $\overline{\mathbf{z}}^1, \ldots, \overline{\mathbf{z}}^N$ with the additional context frames, respectively, and let q_i^n denote the posterior probability computed by the DNN at frame n, i.e.,

$$q_i^n \triangleq p(I^n = i|\overline{\mathbf{v}}^n; \boldsymbol{\theta}) \tag{2.14}$$

where $\boldsymbol{\theta}$ is the DNN parameter-set.

To train the DNN we used a single hidden layers DNN, comprising of 512 rectified linear unit (ReLU) units. The output layer was set to be the ‘softmax’.

Given a training sequence of vectors $\overline{\mathbf{v}}^1, \ldots, \overline{\mathbf{v}}^N$, with their corresponding label vectors, $\boldsymbol{l}^1, \ldots, \boldsymbol{l}^N$, the DNN is trained to minimize the cross entropy function:

$$\mathcal{L}(\boldsymbol{\theta}) = -\sum_{n=1}^{N}\sum_{i=1}^{M}\left(\boldsymbol{l}_i^n \cdot \log q_i^n + (1 - \boldsymbol{l}_i^n) \cdot \log\left(1 - q_i^n\right)\right) \tag{2.15}$$

where $\boldsymbol{l}_i^n$ denotes the i-th component in $\boldsymbol{l}^n$. We normalized every utterance, so that the sample mean and variance of all utterances are zero and one, respectively, in order to avoid mismatch between the train and test conditions.

In test phase, to calculate the SPP ρ_k we propose to use (2.6), where $\rho_{i,k}$ is calculated from the generative model using (2.7), and p_i is produced by the trained DNN (2.14):

$$\rho_k = \sum_{i=1}^{M} q_i \cdot \rho_{i,k}. \tag{2.16}$$

The proposed algorithm is summarized in Algorithm 1 and in the block diagram 2.1.

2.4 Experimental Study

In this section, we describe the experimental setup, evaluation measures and experimental results.

2.4.1 Experimental setup

The train set of the TIMIT database [51] was utilized for modeling the clean speech 2.3.1. To train the DNN, we used the train set of the TIMIT database, randomly contaminated by either *speech-like* noise or *Factory* noise from NOISEX-92 database [52]. The noise was added to the clean signal with the signal to noise ratio (SNR) level of 7 dB.

To test the proposed algorithm, we contaminated clean speech signals from the test set of the TIMIT database [51] with several types of noise from NOISEX-92 database [52], namely *speech-like*, *Babble*, *Factory* and *White*. The noise was added to the clean signal drawn from the test set of the TIMIT database, with 5 levels of SNR at -5 dB, 0 dB, 5 dB, 10 dB and 15 dB chosen to represent various practical conditions. Sampling rate is equal to 16 KHz and the frame length was set to $L = 512$, with overlap of 75% between two successive frames. The size of the input to the DNN, $\mathbf{z}$, prior to adding the context frames, thus equals to $L/2 + 1 = 257$. Note that there is no overlap between the train and test database. To assess the performance of the proposed algorithm, we have used the perceptual evaluation of speech quality (PESQ) measure, which is known to have a high correlation with subjective score [53].

We compared the proposed algorithm with three competing algorithms: 1) The OMLSA algorithm [4]. 2) The MixMax algorithm [9]. 3) The NN-MM algorithm [10]. All methods were trained with the same database.

2.4.2 Results

Fig. 2.2 portrays an example of the spectrogram of the clean input, the noisy input which is the clean signal contaminated with *Factory* noise [52], and the enhanced speech using the MixMax algorithm, the NN-MM algoirthm and the proposed MoG-DNN algorithm.

Fig. 2.3 depicts the PESQ results of all examined algorithms for the different types of noise as a function of the input SNR. In addition, we examined the oracle results for the presented algorithm. First, it is evident that the proposed MoG-DNN algorithm outperforms the OMLSA algorithm and the MixMax algorithm. Additionally, the MoG-DNN outperforms the NN-MM algorithm with a slight marginal improvement, a riveting result since the NN-MM algorithm is supervised. Finally, comparing the presented algorithm using the trained DNN and the oracle results, we conclude that by improving the DNN we can achieve even better results.

As previously stated, the difference between the MoG-DNN algorithm and the MixMax algorithm is in the classification method. In the MoG-DNN algorithm (2.14) is used as opposed to MixMax where (2.8) is used.

Define the Kullback-Leibler divergence (KL-divergence) from $\boldsymbol{p} = [p_1, \ldots, p_M]$ (2.8) to $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_M]$ to be:

$$D_{KL}(\boldsymbol{\alpha} \| \boldsymbol{p}) = \sum_i \alpha_i \log \frac{\alpha i}{p_i}. \tag{2.17}$$

We used the KL-divergence as a measure for the classification accuracy. Table 2.1 depicts the KL-divergence results of the two algorithms for different noise types and different levels of SNR. It is evident that the MoG-DNN algorithm exceeds the MixMax algorithm in the classification task.

Table 2.1: KL-divergence results for various noise types

	Speech-like noise				Babble noise			
Method \SNR	-5 [dB]	0 [dB]	5 [dB]	10 [dB]	-5 [dB]	0 [dB]	5 [dB]	10 [dB]
MoG-DNN	**3.74**	**2.81**	**2.22**	**2.15**	**5.87**	**4.70**	**3.94**	**3.80**
MixMax	16.97	15.11	14.61	14.05	22.43	21.81	21.26	18.56
	White noise				Factory noise			
Method \SNR	-5 [dB]	0 [dB]	5 [dB]	10 [dB]	-5 [dB]	0 [dB]	5 [dB]	10 [dB]
MoG-DNN	**5.16**	**4.38**	**3.51**	**2.85**	**4.31**	**3.22**	**2.53**	**2.24**
MixMax	14.73	15.33	14.0	13.48	20.53	20.60	18.65	19.48

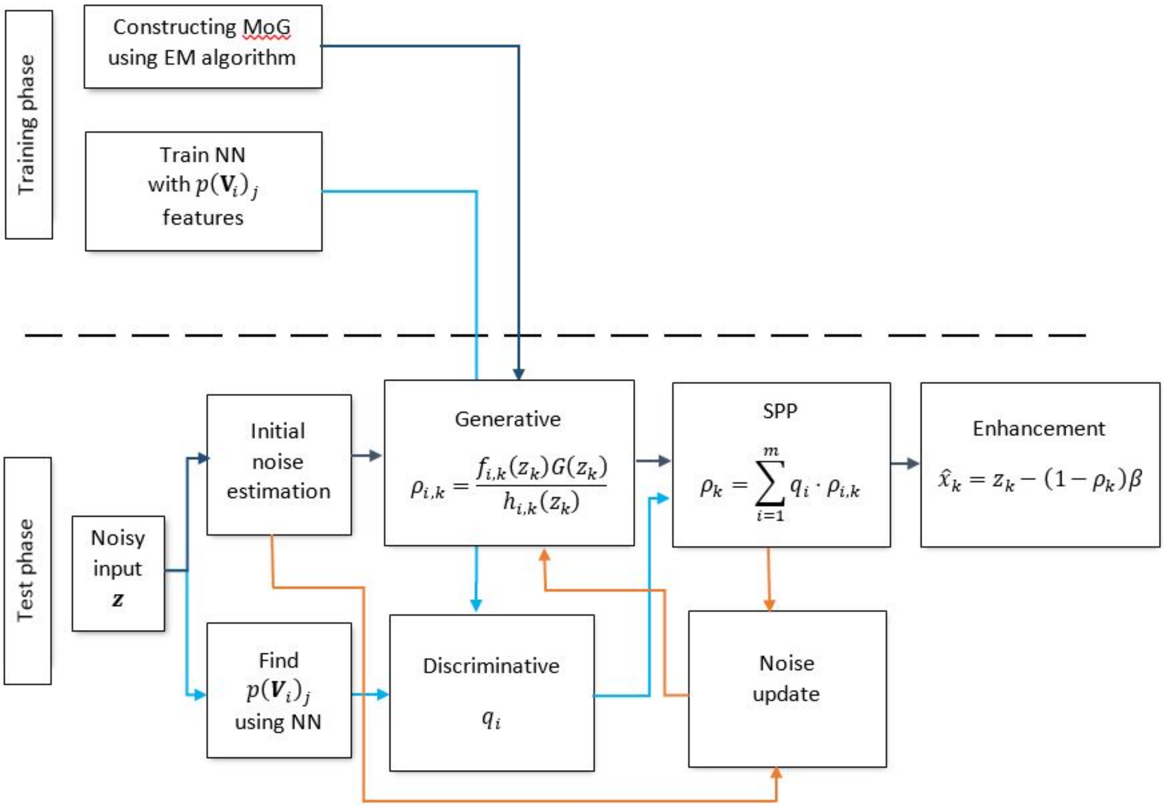
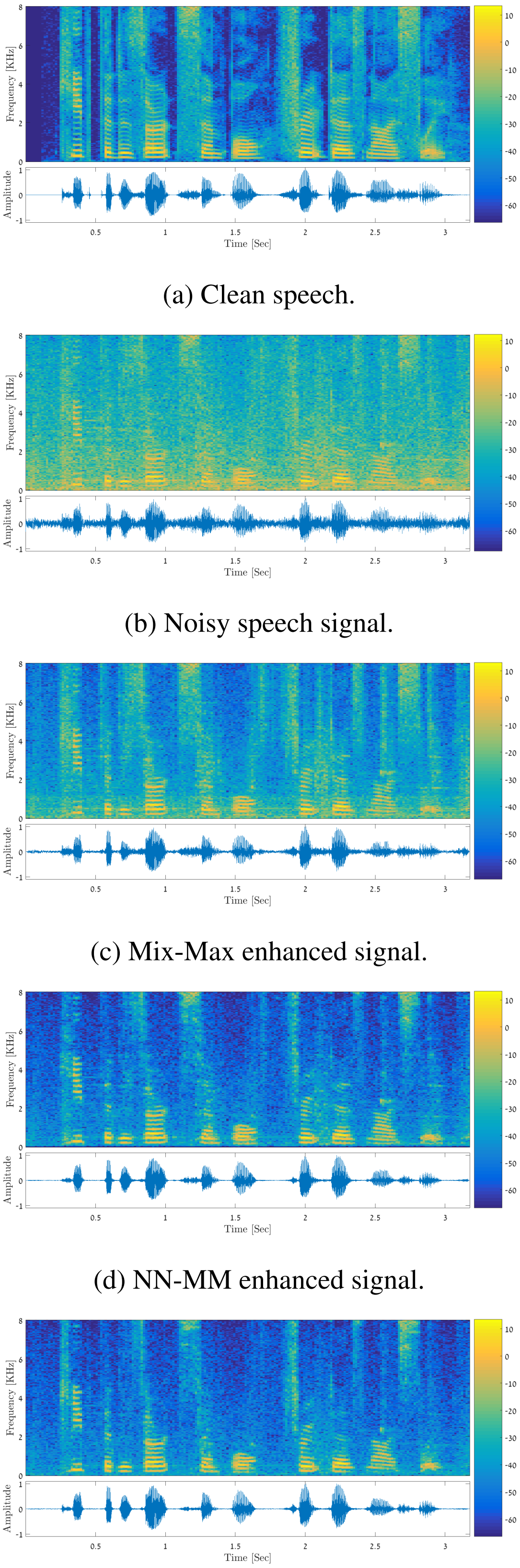

Training phase
Constructing MoG using EM algorithm
Train NN with $p(\mathbf{V}_i)_j$ features
Test phase
Noisy input $\boldsymbol{z}$
Initial noise estimation
Find $p(\boldsymbol{V}_i)_j$ using NN
Generative
$\rho_{i,k} = \frac{f_{i,k}(z_k)G(z_k)}{h_{i,k}(z_k)}$
Discriminative
q_i
SPP
$\rho_k = \sum_{i=1}^{m} q_i \cdot \rho_{i,k}$
Noise update
Enhancement
$\hat{x}_k = z_k - (1 - \rho_k)\beta$

Figure 2.1: Block diagram of MoG-DNN algorithm

Algorithm 1 Algorithm 1: Summary of the proposed mixture of Gaussians-deep neural network (MoG-DNN) algorithm

```
Train phase:
input: Log-spectral vectors of the clean speech $\overline{\mathbf{x}}^1, \dots, \overline{\mathbf{x}}^N$ and log-spectral vectors of the noisy speech $\overline{\mathbf{z}}^1, \dots, \overline{\mathbf{z}}^N$.
MoG training:
Using the EM algorithm, given $\overline{\mathbf{x}}^1, \dots, \overline{\mathbf{x}}^N$ we obtain the MoG parameters for the speech model.
Generating data labels: Create the posterior probability labels $\boldsymbol{l}^1, \dots, \boldsymbol{l}^N$ (2.13) .
DNN training: Train a DNN to output posterior probability of data using $(\overline{\mathbf{z}}^1, \boldsymbol{l}^1), \dots, (\overline{\mathbf{z}}^N, \boldsymbol{l}^N)$.
Test phase:
input: Log-spectral vectors of the noisy speech $\mathbf{z}$.
output: Estimated log-spectral vector of the clean speech $\hat{\mathbf{x}}$.
Compute the classification probabilities for every frame (2.14):
    $q_i^n \triangleq p(I^n = i | \overline{\mathbf{v}}^n; \boldsymbol{\theta}), \quad i = 1, \dots, M; \; n = 1, \dots, N$
for $k = 1 : L/2$ do
    Compute (2.7):
        $\rho_{i,k} = p(y_k < x_k | z_k, I = i) = \frac{f_{i,k}(z_k) G_k(z_k)}{h_{i,k}(z_k)}, \quad i = 1, \dots, M$
    Compute the speech presence probability (SPP) (2.16):
        $\rho_k = \sum_{i=1}^{M} q_i \cdot \rho_{i,k}$
    Estimate the clean speech (2.9):
        $\hat{x}_k = z_k - (1 - \rho_k) \cdot \beta$
    Adapt the noise parameters:
        $\mu_{\mathbf{y},k}^{\text{new}} = \rho_k \cdot \mu_{\mathbf{y},k}^{\text{old}} + (1 - \rho_k)(\delta \cdot z_k + (1 - \delta) \cdot \mu_{\mathbf{y},k}^{\text{old}})$
        $\sigma_{\mathbf{y},k}^{\text{new}} = \rho_k \cdot \sigma_{\mathbf{y},k}^{\text{old}} + (1 - \rho_k) \left( \delta \cdot \sqrt{(z_k - \mu_{\mathbf{y},k}^{\text{new}})^2} + (1 - \delta) \cdot \sigma_{\mathbf{y},k}^{\text{old}} \right)$
end
```



(a) Clean speech.

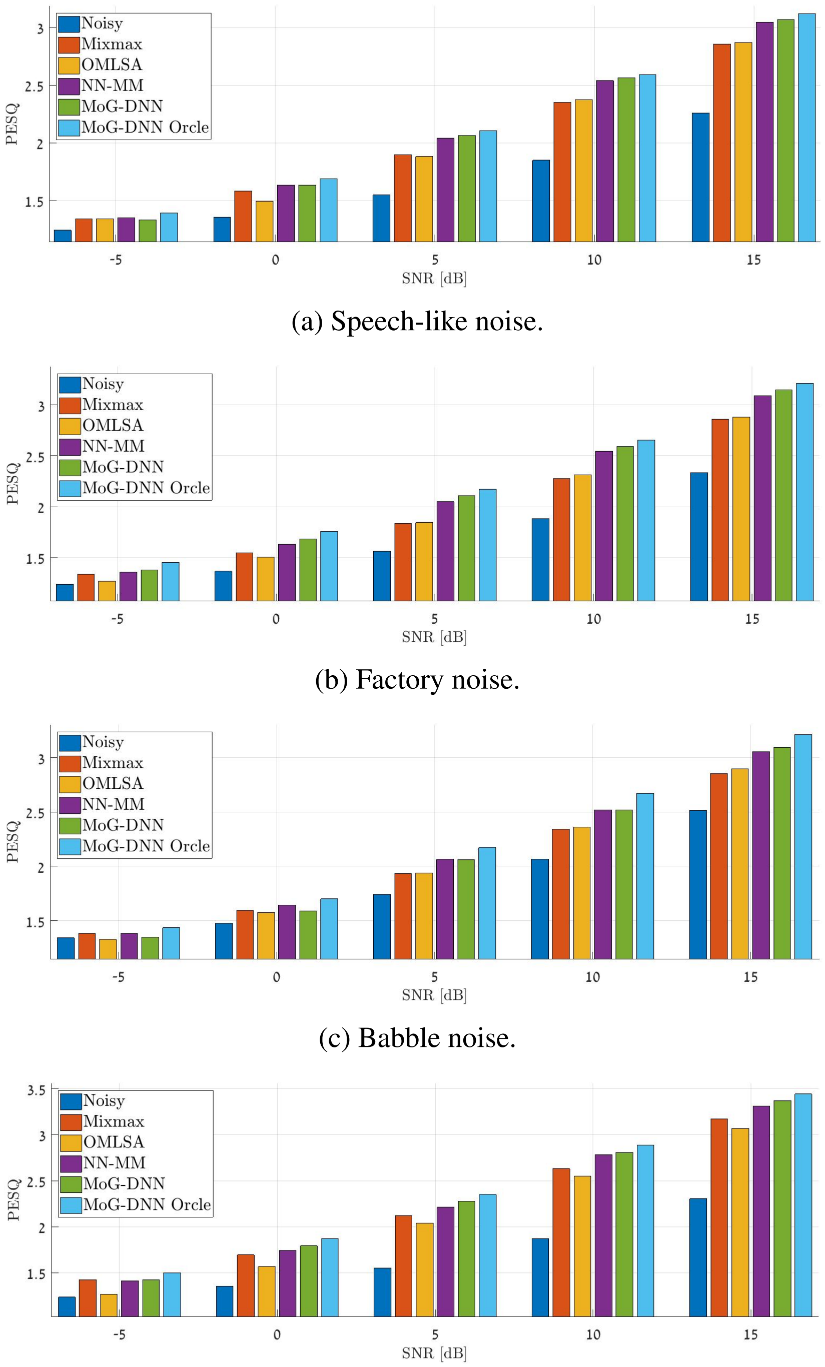
(b) Noisy speech signal.

(c) Mix-Max enhanced signal.

(d) NN-MM enhanced signal.

(e) MoG-DNN enhanced signal.

Figure 2.2: An example of the enhancement results of the MoG-DNN algorithm compared to the MixMax and the NN-MM algorithms.



(a) Speech-like noise.

(b) Factory noise.

(c) Babble noise.

(d) White noise.

Figure 2.3: PESQ results for different noise types.

Chapter 3

Multi-Microphone Speaker Separation Based on Deep DOA Estimation

In this chapter, we present a multi-microphone speech separation algorithm based on masking inferred from the speakers' DOA. According to the W-disjoint orthogonality property of speech signals, each TF bin is dominated by a single speaker. This TF bin can therefore be associated with a single DOA. In our procedure, we apply a DNN with a U-net architecture to infer the DOA of each TF bin from a concatenated set of the spectra of the microphone signals. Separation is obtained by multiplying the reference microphone by the masks associated with the different DOAs. Our proposed DDESS method is inspired by the recent advances in deep clustering methods. Unlike already established methods that apply the clustering in a latent embedded space, in our approach the embedding is closely associated with the spatial information, as manifested by the different speakers' directions of arrival.

3.1 Deep Speech Separation

3.1.1 The separation algorithm

Consider an array of M microphones capturing a mixture of N speech sources in a reverberant enclosure. The i-th speech signal $s^i(t)$ propagates through the acoustic channel before being captured by the mth microphone:

$$z_m(t) = \sum_{i=1}^{N} s^i(t) * h^i_m(t), \qquad m \in \{1, \ldots, M\} \tag{3.1}$$

where, h^i_m is the RIR relating the ith speaker and the mth microphone. In the STFT domain, (3.1) can be rewritten as:

$$z_m(l,k) = \sum_{i=1}^{N} s^i(l,k) h^i_m(l,k), \tag{3.2}$$

where l and k, are the time-frame and the frequency-bin (TF) indexes, respectively.

Following the W-disjoint orthogonality assumption [16], each TF bin is dominated by a single speaker. We assume that each speaker is located at a different DOA and therefore each bin is dominated by a single DOA. The crux of our speech separation method is to estimate the DOA for each TF bin by a neural network and then separate the speakers by grouping these bins according to their estimated DOA.

The main building block of the algorithm is a neural network that uses the microphone signals to infer the DOA at each TF bin of a given time-frequency image. The network input is a $L \times K$ time-frequency "image" where L is the number of time frames and K is the number of frequency bins. We have chosen to substitute the raw microphone signals with the phase of the instantaneous relative transfer function (iRTF) estimate, calculated as the phase of the bin-wise ratio between the mth microphone signal and the reference microphone signal. The phase angle is encoded as a point in the unit circle. The input features to the network, therefore, is an $L \times K$ matrix $\mathcal{R}$ where each (l,k) entry has M

channels each correspond to a microphone:

$$\mathcal{R}(m,l,k) = (\cos(\angle \frac{z_m(l,k)}{z_{\text{ref}}(l,k)}), \sin(\angle \frac{z_m(l,k)}{z_{\text{ref}}(l,k)})). \tag{3.3}$$

Due to the W-disjoint assumption, the normalized features $\mathcal{R}(m,l,k)$ are dominated by a single speaker and hence correspond to a specific DOA. Ideally, the speech contribution to $\mathcal{R}(m,l,k)$ is negligible. Hence, it is expected that these are better features than the raw data for DOA estimation.

We form the DOA estimation as a classification task by discretizing the possible angles to be in the set $\Theta = \{0°, 15°, 30° \ldots, 180°\}$. Let $D_{l,k}$ be a random variable indicating the active direction at bin (l,k). The target of the network is to infer the conditional distribution of the discrete set of candidate DOAs in Θ for each TF bin, given the recorded signal:

$$p_{l,k}(\theta) = p(D_{l,k} = \theta | \mathcal{R}), \quad \theta \in \Theta. \tag{3.4}$$

where $\mathcal{R}$ is an $M \times L \times K$ matrix of all the TF bins. The image-to-image DOA prediction task in (3.4) is implemented by a U-net, which details are given in the next section.

Next, the direction-dependent power is calculated by the instantaneous power of the reference microphone, weighted by the U-net output:

$$E(\theta) = \sum_{l,k} p_{l,k}(\theta) \cdot |z_{\text{ref}}(l,k)|^2, \qquad \theta \in \Theta. \tag{3.5}$$

Note that the total power is satisfying the following equation:

$$E = \sum_{\theta \in \Theta} E(\theta) = \sum_{l,k} |z_{\text{ref}}(l,k)|^2. \tag{3.6}$$

High power from a specific direction is an indication for an active speaker in this direction. To find all directions of the active speakers in the scene, we sort the powers according to their power level:

$$E(\theta_1) \geq E(\theta_2) \geq E(\theta_3) \ ...$$

where θ_1 corresponds to the direction with the highest power, θ_2 the second highest, etc. The speakers' directions are then determined by the N DOAs with the highest power level. If the number of speakers N is not known in advance, we can set N as the minimal value such that $\sum_{i=1}^{N} E(\theta_i) > \alpha E$, with α is a predefined threshold.

The next step is to use the estimated DOA to form a *mask* for each detected speaker in the scene. The estimated mask of the ith speaker is the U-net output:

$$\hat{M}_i(l,k) = p_{l,k}(\theta_i) \tag{3.7}$$

and the absolute value of the ith speaker signal is reconstructed as follows:

$$|\hat{s}^i(l,k)| = |z_{\text{ref}}(l,k)| \cdot \hat{M}_i(l,k). \tag{3.8}$$

The noisy phase is then used to reconstruct the separated signals in the time-domain, by the application of the inverse STFT. We dub the proposed algorithm deep direction estimation for speech separation (DDESS).

Note, that if a static acoustic scene can be assumed, namely that the sources do not significantly change their DOA during the entire utterance, permutation problems, which are typical to clustering-based approaches [19], are circumvented.

Note that estimating the DOA is modeled here as a classification problem and not as a regression task. We are not interested in finding the exact DOAs of the speakers in the scenario but rather, grouping them into distinct directions. That is, even with inaccurate DOA estimate, the speech separation can still work, provided that most TF bins are clustered to mutually exclusive classes.

3.1.2 The U-net for DOA estimation

The input to the network is the feature matrix $\mathcal{R}$. The overlap between successive STFT frames is set to $75\ \%$. Hence, to improve the estimation accuracy of the relative transfer functions (RTFs), we have used an average of three consecutive frames both in the numerator and denominator of (3.3).

In our U-net architecture, the input shape is $(2(M-1), L, K)$ where, $K = 256$ is the number of frequency bins, $L = 96$ is the number of frames, and M is the number of microphones. The output shape is $(|\Theta|, L, K)$ where $|\Theta|$ is the cardinality of the set Θ.

The U-net architecture is presented in Fig. 3.1. The blue boxes depict the encoder and the green boxes the decoder. In this architecture, in the encoder part, the input image is squeezed into a bottleneck using 2×2 max pooling operations (downsample), and then in the encoder part, it is upsampled back to the original image shape. The main problem with this architecture is that during the pooling operation, important local information is lost. To tackle this problem, a U-shape architecture was developed in [31]. The U-net connects between mirrored layers in the encoder and decoder by passing the information without going through the bottleneck and thus, alleviating the information loss problem.

Let CE$_{l,s}$ denote a 2D convolution layer with 'elu' as the activation function, where l is the number of filters and $s \times s$ is the filter size. Similarly, let DE$_{l,s}$ is the de-convolution 'elu' layer. Finally, let P$_s$ denote the max-pooling operation with filter size $s \times s$.

The encoder down-sampling path is given by:

$\text{CE}_{16,3} \rightarrow \text{CE}_{16,3} \rightarrow \text{P}_2 \rightarrow \text{CE}_{32,3} \rightarrow \text{CE}_{32,3} \rightarrow \text{P}_2 \rightarrow \text{CE}_{64,3} \rightarrow \text{CE}_{64,3} \rightarrow \text{P}_2 \rightarrow \text{CE}_{128,3} \rightarrow \text{CE}_{128,3} \rightarrow \text{P}_2 \rightarrow \text{CE}_{256,3} \rightarrow \text{CE}_{256,3}$.

The decoder up-sampling path is given by:

$\text{DE}_{128,3} \rightarrow \text{CE}_{128,3} \rightarrow \text{CE}_{128,3} \rightarrow \text{DE}_{64,3} \rightarrow \text{CE}_{64,3} \rightarrow \text{CE}_{32,3} \rightarrow \text{DE}_{32,3} \rightarrow \text{CE}_{32,3} \rightarrow \text{CE}_{32,3} \rightarrow \text{DE}_{16,3} \rightarrow \text{CE}_{16,3} \rightarrow \text{CE}_{16,3} \rightarrow \text{CE}_{13,1}$.

The output DOA distribution is finally obtained by a softmax layer. To overcome the problem of overfitting, we add dropout layers[54] after every CE$_{l,s}$ layer. Additionally, the raw data input is normalized to zero mean and unit variance.

To train the network, we use a simulated data where both the location and a clean recording of each speaker are given. We can thus easily find for each TF bin (l, k) the dominant speaker and the corresponding DOA $y_{k,l} \in \Theta$. The network is trained to minimize the cross entropy between the correct and the estimated DOA. The cross entropy

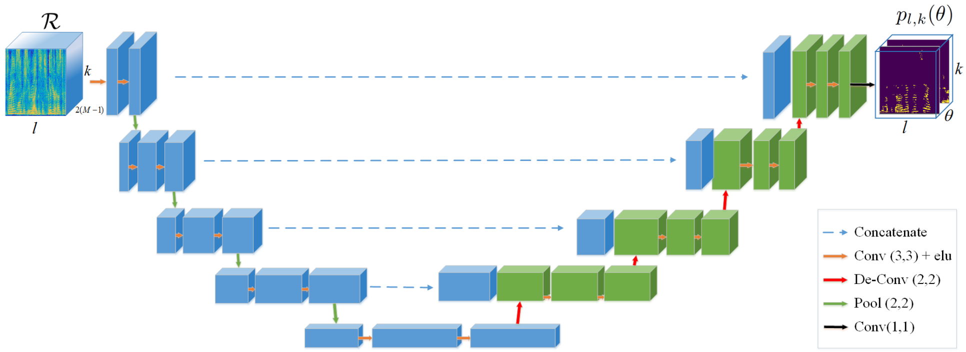


Figure 3.1: U-net architecture for DOA-mask speech separation. The blue blocks depict the encoder and the green blocks depict the decoder.

cost function is summed over all the images in the training set. The network was implemented with Tensor-Flow and training was done using the ADAM optimizer [55]. The number of epochs was set to be 100, and the training stopped after validation loss was going up for 3 successive epochs. The minibatch size was 64 images.

3.2 Experimental study

In this section, we evaluate the proposed DDESS algorithm and compare its performance to the DUET algorithm [21].

3.2.1 Training database

To generate the training data, we used the RIR generator[1] efficiently implementing the image method [56]. We simulated an eight microphone array with $(3, 3, 3, 8, 3, 3, 3)$ cm between microphones. Similar microphone inter-distance was used in the test phase. The dimensions of the room are $6 \times 6 \times 2.4$ (width, length and height), similar to the acoustic lab used in the test phase. The microphone array was positioned at $(3, 1, 1.5)$ m.

For each scenario, two clean signals from the Wall Street Journal 1 (WSJ1) database [57] were randomly selected and two different DOAs were also randomly selected from the possible values in the range $\Theta = \{0, 15, \ldots, 180\}$. The speakers were located in a

[1] Available online at `github.com/ehabets/RIR-Generator`

radius of $r = 1.5m$ from the center of the microphone array. To increase the training diversity, the radius of the speakers was perturbed by a Gaussian noise with variance 0.3 m. The DOA of each speaker was computed with respect to the center of the array. We used $\mathrm{T}_{60} \in \{0.2, 0.3, 0.4\}$sec. Once the scenario is set, the RIRs were generated, and the clean signals were separately convolved with them. Finally, we added the signals with signal to interfering ratio (SIR) randomly chosen in the range SIR $\in [-2, 2]$. Sampling rate was set to 16KHz and the frame length of the STFT was set to $K = 512$, with overlap of 75% between two successive frames. The training set comprises two hours of recordings with 6000 different scenarios of mixtures of two speakers.

3.2.2 Separation results

For each test scenario, we selected two speakers (male or female) from the test set of the TIMIT database, placed them in two different angles between $0°$ to $180°$ relative to the microphone array, at the distance of either 1 m or 2 m.

Each clean speech signal was convolved with a real RIR, drawn from the multichannel impulse response database recorded in our acoustic lab [43] (similar room dimensions and microphone inter-distances to the simulated scenarios), and then mixed the with SIR=0 dB. We used $T_{60} = 160/360$ ms for the room reverberation. Overall, in the test dataset, we had 30 different scenarios for each T_{60}, and the results are averaged over all scenarios.

We used a standard blind source separation (BSS) evaluation toolbox [58] to test the separation capabilities of the DDESS algorithm and the DUET algorithm [21]. Tables 3.1 and 3.2 present the SIR and signal to distortion ratio (SDR) results for the two source distances, 1m and 2m, respectively. It is evident that the DDESS algorithm outperforms the DUET in all experiments.

Fig. 3.2 depicts the spectrogram of the noisy input, the clean signals and the estimates obtained by the proposed algorithm for two equi-power speakers positioned at $90°$ and $180°$ and $r = 2$ m and for $\mathrm{T}_{60} = 160$ ms. It is evident that the DDESS separates the signals. Fig. 3.2b depicts the power level for DOA candidates. It is clear that the DOAs were accurately classified. Sound samples can be found in the lab website.[2]

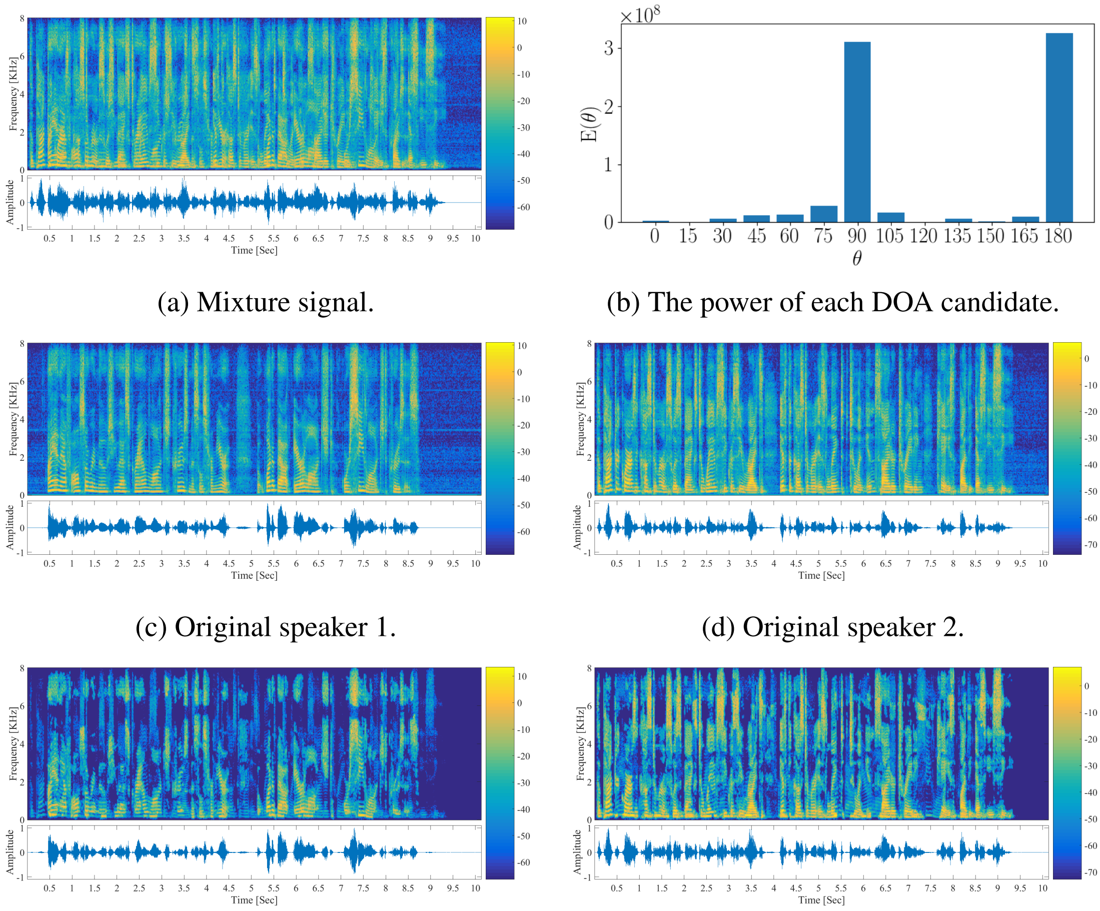
Table 3.1: SDR and SIR results with two T_{60} and distance 1m.

	$T_{60} = 160$				$T_{60} = 360$			
	SDR		SIR		SDR		SIR	
Speaker	1	2	1	2	1	2	1	2
Noisy	-1.05	-1.41	0.23	-0.11	-0.91	-1.75	0.5	-0.41
DUET	1.3	0.7	4.24	3.38	0.87	-0.33	3.59	2.24
DDESS	**2.26**	**1.95**	**12.6**	**12.43**	**1.68**	**1.69**	**13.06**	**12.76**

Table 3.2: SDR and SIR results with two T_{60} and distance 2m.

	$T_{60} = 160$				$T_{60} = 360$			
	SDR		SIR		SDR		SIR	
Speaker	1	2	1	2	1	2	1	2
Noisy	-1.22	-1.49	0.19	-0.07	-2.07	-1.07	-0.5	0.68
DUET	-0.31	-0.26	2.24	2.41	-1.79	-0.1	1.04	2.44
DDESS	**1.38**	**1.31**	**11.46**	**11.44**	**0.08**	**1.02**	**11.1**	**11.68**

[2] www.eng.biu.ac.il/gannot/speech-enhancement/



(a) Mixture signal.

(b) The power of each DOA candidate.

(c) Original speaker 1.

(d) Original speaker 2.

(e) Estimated speaker 1.

(f) Estimated speaker 2.

Figure 3.2: An example of the separation results of the DDESS algorithm.

Chapter 4

FCN Approach for Dynamically Locating Multiple Speakers

In this chapter, we present a deep neural network-based online multi-speaker localisation algorithm. Following the W-disjoint orthogonality principle in the spectral domain, each TF bin is dominated by a single speaker, and hence by a single DOA. A fully convolutional network is trained with instantaneous spatial features to estimate the DOA for each TF bin. The high resolution classification enables the network to accurately and simultaneously localize and track multiple speakers, both static and dynamic. Elaborated experimental study using both simulated and real-life recordings in static and dynamic scenarios, confirms that the proposed algorithm outperforms both classic and recent deep-learning-based algorithms.

4.1 Multiple speakers' location algorithm

4.1.1 Time-frequency features

Consider an array with M microphones acquiring a mixture of N speech sources in a reverberant environment. The i-th speech signal $s^i(t)$ propagates through the acoustic channel before being acquired by the m-th microphone:

$$z_m(t) = \sum_{i=1}^{N} s^i(t) * h^i_m(t), \qquad m = 1, \dots, M, \tag{4.1}$$

where h^i_m is the RIR relating the i-th speaker and the m-th microphone. In the STFT domain (4.1) can be written as (provided that the frame-length is sufficiently large w.r.t. the filter length):

$$z_m(l,k) = \sum_{i=1}^{N} s^i(l,k) h^i_m(l,k), \tag{4.2}$$

where l and k, are the time frame and the frequency indices, respectively.

The STFT (4.2) is complex-valued and hence comprises both spectral and phase information. It is clear that the spectral information alone is insufficient for DOA estimation. It is therefore a common practice to use the phase of the TF representation of the received microphone signals, or their respective phase-difference, as they are directly related to the DOA in non-reveberant environments. We decided to use an alternative feature, which is generally independent of the speech signal and is mainly determined by the spatial information. For that, we have selected the RTF [59] as our feature, since it is known to encapsulate the spatial fingerprint for each sound source. Specifically, we use the instantaneous relative transfer function (iRTF), which is the bin-wise ratio between the m-th microphone signal and the reference microphone signal $z_{\text{ref}}(l,k)$:

$$\text{iRTF}(m,l,k) = \frac{z_m(l,k)}{z_{\text{ref}}(l,k)}. \tag{4.3}$$

Note, that the reference microphone is arbitrarily chosen. Reference microphone selection is beyond the scope of this chapter (see [60] for a reference microphone selection method). The input feature set extracted from the recorded signal is thus a 3D tensor $\mathcal{R}$:

$$\mathcal{R}(m,l,k) = [\mathfrak{Re}(\text{iRTF}(m,l,k)), \mathfrak{Im}(\text{iRTF}(m,l,k))]. \tag{4.4}$$

The matrix $\mathcal{R}$ is constructed from $L \times K$ bins, where L is the number of time frames and K is the number of frequencies. Since the iRTFs are normalized by the reference microphone, it is excluded from the features. Then for each TF bin (l, k), there are $P = 2(M - 1)$ channels, where the multiplication by 2 is due to the real and imaginary parts of the complex-valued feature. For each TF bin the spatial features were normalized to have a zero mean and a unit variance.

Recall that the WDO assumption [16] implies that each TF bin (l, k) is dominated by a single speaker. Consequently, as the speakers are spatially separated, i.e. located at different DOAs, each TF bin is dominated by a single DOA. Our goal in this work is to accurately estimate the speaker direction at every TF bin from the given mixed recorded signal.

4.1.2 FCN for DOA estimation

We formulated the DOA estimation as a classification task by discretizing the DOA range. The resolution was set to 5°, such that the DOA candidates are in the set $\Theta = \{0°, 5°, 10°, \ldots, 180°\}$. Let $D_{l,k}$ be a r.v. representing the active dominant direction, recorded at bin (l, k). Our task boils down to deducing the conditional distribution of the discrete set of DOAs in Θ for each TF bin, given the recorded mixed signal:

$$p_{l,k}(\theta) = p(D_{l,k} = \theta | \mathcal{R}), \quad \theta \in \Theta. \tag{4.5}$$

For this task, we use a DNN. The network output is an $|\Theta| \times L \times K\times$ tensor, where $|\Theta|$ is the cardinality of the set Θ. Under this construction of the feature tensor and output probability tensor, a pixel-to-pixel approach for mapping a 3D input ‘image’, $\mathcal{R}$ and a 3D output ‘image’, $p_{l,k}(\theta)$, can be utilized. An FCN is used to compute (4.5) for each TF bin. The pixel-to-pixel method is beneficial in two ways. First, for each TF bin in our input image the network estimates the DOA distribution separately. Second, the TF supervision is carried out with the spectrum of the different speakers. The FCN hence takes advantage of the spectral structure and the continuity of the sound sources in both the time and frequency axes. These structures contribute to the pixel-wise classification task, and prevent discontinuity in the DOA decisions over time. In our implementation, we used a U-net architecture, similar to the one described in [61]. We dub our algorithm time-frequency direction-of-arrival net (TF-DOAnet).

The input to the network is the feature matrix $\mathcal{R}$ (4.4). In our U-net architecture, the input shape is (P, L, K) where $K = 256$ is the number of frequency bins, $L = 256$ is the number of frames, and $P = 2M - 2$ where M is the number of microphones. The overlap between successive STFT frames is set to 75%. This allows to improve the estimation accuracy of the RTFs, by averaging three consecutive frames both in the numerator and denominator of (4.3), without sacrificing the instantaneous nature of the RTF.

TF bins in which there is no active speech are non-informative. Therefore, the estimation is carried out only on speech-active TF bins. As we assume that the acquired signals are noiseless, we define a TF-based voice activity detector (VAD) as follows:

$$\text{VAD}(l,k) = \begin{cases} 1 & |z_{\text{ref}}(l,k)| \geq \epsilon \\ 0 & \text{o.w.} \end{cases}, \tag{4.6}$$

where ϵ is a threshold value. In noisy scenarios, we can use a robust SPP estimator instead of the VAD [8].

The task of DOA estimation only requires time frame estimates. Hence, we aggregate over all active frequencies at a given time frame to obtain a frame-wise probability:

$$p_l(\theta) = \frac{1}{K'} \sum_{k=1}^{K} p_{l,k}(\theta) \text{VAD}(l,k). \tag{4.7}$$

where K' is the number of active frequency bands at the l-th time frame. We thus obtain for each time frame a posterior distribution over all possible DOAs. If the number of speakers is known in advance, we can choose the directions corresponding to the highest posterior probabilities. If an estimate of the number of speakers is also required, it can be determined by applying a suitable threshold. Figure 3.1 summarizes the TF-DOAnet in a

Table 4.1: Configuration of training data generation. All rooms are 2.7 m in height

Simulated training data					
	Room 1	Room 2	Room 3	Room 4	Room 5
Room size	(6×6) m	(5×4) m	$(10 \times 6$ m$)$	(8×3) m	(8×5) m
RT_{60}	0.3 s	0.2 s	0.8 s	0.4 s	0.6 s
Signal	Noiseless signals from WSJ1 **training** database				
Array position in room	6 arbitrary positions in each room				
Source-array distance	1.5 m with added noise with 0.1 variance				

block diagram.

4.1.3 Training phase

The supervision in the training phase is based on the WDO assumption in which each TF bin is dominated by (at most) a single speaker. The training is based on simulated data generated by a publicly availble RIR generator software[1], efficiently implementing the image method [56]. A four microphone linear array was simulated with $(8, 8, 8)$ cm inter-microphones distances. Similar microphone inter-distances were used in the test phase. For each training sample, the acoustic conditions were randomly drawn from one of the simulated rooms of different sizes and different reverberation levels RT_{60} as described in Table 4.1. The microphone array was randomly placed in the room in one out of six arbitrary positions.

For each scenario, two clean signals were randomly drawn from the WSJ1 database [57] and then convolved with RIRs corresponding to two different DOAs in the range $\Theta = \{0, 5, \ldots, 180\}$. The sampling rate of all signals and RIRs was set to 16KHz. The speakers were positioned in a radius of $r = 1.5m$ from the center of the microphone array. To enrich the training diversity, the radius of the speakers was perturbed by a Gaussian noise with a variance of 0.1 m. The DOA of each speaker was calculated w.r.t. the center of the microphone array.

[1] Available online at `github.com/ehabets/RIR-Generator`

Table 4.2: Configuration of test data generation. All rooms are 3 m in height

	Simulated test data	
	Room 1	Room 2
Room size	(5 × 7) m	(9 × 4) m
RT_{60}	0.38 s	0.7 s
Source-array distance	1.3 m	1.7 m
Signal	Noiseless signals from WSJ1 **test** database	
Array position in room	4 arbitrary positions in each room	

The contributions of the two sources were then summed with a random SIR selected in the range of SIR $\in [-2, 2]$ to obtain the received microphone signals. Next, we calculated the STFT of both the mixture and the STFT of the separate signals with a frame-length $K = 512$ and an overlap of 75% between two successive frames.

We then constructed the audio feature matrix $\mathcal{R}$ as described in Sec. 4.1.1. In the training phase, both the location and a clean recording of each speaker were known, hence they could be used to generate the labels. For each TF bin (l, k), the dominating speaker was determined by:

$$\text{dominant speaker} \leftarrow \underset{i}{\operatorname{argmax}} \, |s^i(l,k) h^i_{\text{ref}}(l,k)|. \tag{4.8}$$

The ground-truth label $D_{l,k}$ is the DOA of the dominant speaker. The training set comprised four hours of recordings with 30000 different scenarios of mixtures of two speakers. It is worth noting that as the length of each speaker recording was different, the utterances could also include non-speech or single-speaker frames. The network was trained to minimize the cross-entropy between the correct and the estimated DOA. The cross-entropy cost function was summed over all the images in the training set. The network was implemented in Tensorflow with the ADAM optimizer [55]. The number of epochs was set to be 100, and the training stopped after the validation loss increased for 3 successive epochs. The mini-batch size was set to be 64 images.

4.2 Experimental Study

4.2.1 Experimental setup

In this section we evaluate the TF-DOAnet and compare its performance to classic and DNN-based algorithms. To objectively evaluate the performance of the TF-DOAnet, we first simulated 2 unfamiliar test rooms. Then, we tested our TF-DOAnet with real RIR recordings in different rooms. Finally, a real-life scenario with fast moving speakers was recorded and tested.

For each test scenario, we selected two speakers from the test set of the WSJ1 database [57], placed them at two different angles between 0° and 180° relative to the microphone array, at a distance of either 1m or 2m. The signals were generated by convolving the signals with RIRs corresponding to the source positions and with either simulated or recorded acoustic scenarios.

Performance measures Two different measures to objectively evaluate the results were used: the mean absolute error (MAE) and the localization accuracy (Acc.). The MAE, computed between the true and estimated DOAs for each evaluated acoustic condition, is given by

$$\mathrm{MAE}(^\circ) = \frac{1}{N \cdot C} \sum_{c=1}^{C} \min_{\pi \in S_N} \sum_{n=1}^{N} |\theta_n^c - \hat{\theta}_{\pi(n)}^c|, \tag{4.9}$$

where N is the number of simultaneously active speakers and C is the total number of speech mixture segments considered for evaluation for a specific acoustic condition. In our experiments $N = 2$. The true and estimated DOAs for the n-th speaker in the c-th mixture are denoted by θ_n^c and $\hat{\theta}_n^c$, respectively.

The localization accuracy is given by

$$\mathrm{Acc.}(\%) = \frac{\hat{C}_{\text{acc.}}}{C} \times 100 \tag{4.10}$$

where $\hat{C}_{\text{acc.}}$ denotes the number of speech mixtures for which the localization of the speakers is accurate. We considered the localization of speakers for a speech frame to be accurate if the distance between the true and the estimated DOA for all the speakers was less than or equal to 5°.

Compared algorithms We compared the performance of the TF-DOAnet with two frequently used baseline methods, namely the MUSIC and SRP-PHAT algorithms. In addition, we compared its performance with the CNN multi-speaker DOA (CMS-DOA) estimator [32].[2] To facilitate the comparison, the MUSIC pseudo-spectrum was computed for each frequency sub-band and for each STFT time frame, with an angular resolution of 5° over the entire DOA domain. Then, it was averaged over all frequency subbands to obtain a broadband pseudo-spectrum followed by averaging over all the time frames L. Next, the two DOAs with the highest values were selected as the final DOA estimates. Similar post-processing was applied to the computed SRP-PHAT pseudo-likelihood for each time frame.

4.2.2 Speaker localization results

Static simulated scenario We first generated a test dataset with simulated RIRs. Two different rooms were used, as described in Table 4.2. For each scenario, two speakers (male or female) were randomly drawn from the WSJ1 test database, and placed at two different DOAs within the range $\{0, 5, \ldots, 180\}$ relative to the microphone array. The microphone array was similar to the one used in the training phase. Using the RIR generator, we generated the RIR for the given scenario and convolved it with the speakers' signals.

The results for the TF-DOAnet compared with the competing methods are depicted in Table 4.3. The tables shows that the deep-learning approaches outperformed the classic approaches. The TF-DOAnet achieved very high scores and outperformed the DNN-based CMS-DOA algorithm in terms of both MAE and accuracy.

[2] the trained model is available here `https://github.com/Soumitro-Chakrabarty/Single-speaker-localization`

Static real recordings scenario The best way to evaluate the capabilities of the TF-DOAnet is testing it with real-life scenarios. For this purpose, we first carried out experiments with real measured RIRs from a multi-channel impulse response database [43]. The database comprises RIRs measured in an acoustics lab for three different reverberation times of RT$_{60} = 0.160, 0.360$, and 0.610 s. The lab dimensions are $6 \times 6 \times 2.4$ m.

The recordings were carried out with different DOA positions in the range of $[0°, 180°]$, in steps of $15°$. The sources were positioned at distances of 1 m and 2 m from the center of the microphone array. The recordings were carried out with a linear microphone array consisting of 8 microphones with three different microphone spacings. For our experiment, we chose the [8, 8, 8, 8, 8, 8, 8] cm setup. In order to construct an array setup identical to the one in the training phase, we selected a sub-array of the four center microphones out of the total 8 microphones in the original setup. Consequently, we used a uniform linear array (ULA) with $M = 4$ elements with an inter-microphone distance of 8 cm.

The results for the TF-DOAnet compared with the competing methods are depicted in Table 4.4. Again, the TF-DOAnet outperforms all competing methods, including the CMS-DOA algorithm. Interestingly, for the 1 m case, the best results for the TF-DOAnet were obtained for the highest reverberation level, namely RT$_{60} = 610$ ms, and for the 2 m case, for RT$_{60} = 360$ ms. While surprising at first glance, this can be explained using the following arguments. There is an accumulated evidence that reverberation, if properly addressed, can be beneficial in speech processing, specifically for multi-microphone speech enhancement and source extraction [59, 62, 63] and for speaker localization [64, 65]. In reverberant environments, the intricate acoustic propagation pattern constitutes a specific "fingerprint" characterizing the location of the speaker(s). When reverberation level increases, this fingerprint becomes more pronounced and is actually more informative than its an-echoic counterpart. An inference methodology that is capable of extracting the essential driving parameters of the RIR will therefore improve when the reverberation is higher. If the acoustic propagation becomes even more complex, as is the case of high reverberation and a remote speaker, a slight performance degradation may occur, but as evident from the localization results, for sources located 2 m from the array, the performance for RT$_{60}$ $= 610$ ms was still better than the performance for RT$_{60}$ $= 160$ ms.

Real-life dynamic scenario To further evaluate the capabilities of the TF-DOAnet, we also carried out real dynamic scenarios experiments. The room dimensions are $6 \times 6 \times 2.4$ m. The room reverberation level can be adjusted and we set the RT$_{60}$ at two levels, 390 ms and 720 ms, respectively. The microphone array consisted of 4 microphones with an inter-microphone spacing of 8 cm. The speakers walked naturally on an arc at a distance of about 2.2 m from the center of the microphone array. Figure 4.1a depicts the real-life experiment setup and Fig. 4.1b depicts a schematic diagram of the setup of these experiments. The ground truth labels of these experiment were measured with the Marvelmind indoor 3D tracking set.[3]

For the first experiment, the two speakers started at the angles $20°$ and $160°$ and walked until they reached $70°$ and $100°$, respectively, turned around and walked back to their starting point. This was done several times throughout the recording. Figures 4.2 and 4.3 depict the results of the this experiment in both RT$_{60}$ levels.

For the second experiment, the two speakers started at the angles $30°$ and $150°$ and walked until they reached $150°$ and $30°$, respectively. Note that in this experiment there is an overlap between the DOAs of the speakers. Figures 4.4 and 4.5 depict the results of the this experiment in both RT$_{60}$ levels.

It is clear that the TF-DOAnet outperformed the CMS-DOA algorithm, especially for the high RT$_{60}$ conditions. Whereas the CMS-DOA fluctuated rapidly, the TF-DOAnet output trajectory was smooth and noiseless.

[3] https://marvelmind.com/product/starter-set-ia-02-3d/

Table 4.3: Results for two different test rooms with simulated RIRs

Test Room	Room 1		Room 2	
Measure	MAE	Acc.	MAE	Acc.
MUSIC [34]	26.2	28.4	31.5	16.9
SRP-PHAT [35]	25.1	26.7	35.0	15.6
CMS-DOA [32]	13.1	71.1	24.0	38.1
TF-DOAnet	**0.3**	**99.5**	**1.7**	**94.3**

Table 4.4: Results for three different rooms at distances of 1 m and 2 m with measured RIRs

Distance	1 m						2 m					
RT_{60}	0.160 s		0.360 s		0.610 s		0.160 s		0.360 s		0.610 s	
Measure	MAE	Acc.	MAE	Acc.	MAE	Acc.	MAE	Acc.	MAE	Acc.	MAE	Acc.
MUSIC	18.7	57.6	19.2	53.2	21.9	42.9	18.4	54.1	26.1	35.8	25.4	32.2
SRP-PHAT	9.0	39.0	13.9	39.4	18.6	29.9	9.7	36.0	16.5	24.7	27.7	21.3
CMS-DOA	1.6	76.3	7.3	75.2	8.4	71.9	5.1	79.5	9.7	60.1	17.5	40.0
TF-DOAnet	**1.3**	**97.5**	**3.5**	**83.5**	**0.9**	**98.3**	**5.0**	**89.5**	**1.7**	**95.7**	**4.8**	**84.2**

4.2.3 Ablation study

In our implementation, we used the real and imaginary part of the RTF (4.4). Other approaches might be beneficial. For example, in chapter 3, the $\cos$ and the $\sin$ of the phase of the RTF were used. In other approaches, the spectrum was added to the spatial features [30].

In this section, the different features were tested with the same model. We compared the proposed features with two other features. First, we used the proposed features as described in (4.4). The second approach was a variant of our approach with the spectrum added (‘TF-DOAnet with Spec.’). The third, used the $\cos$ and the $\sin$ features as presented in chapter 3 (‘Cos-Sin’). All features were crafted from the same training data described in Sec. 4.1.3. We tested the different approaches in the test conditions described in 4.2.

First, it is clear that all the features with our high resolution TF model outperformed the frame-based CMS-DOA algorithm, as reported in Table 4.3. This confirms that the TF supervision is beneficial for the task at hand. Second, the proposed features were shown to be better than the Cos-Sin features. Finally, it is very interesting to note that the addition of the spectrum features slightly deteriorated the results for this task.

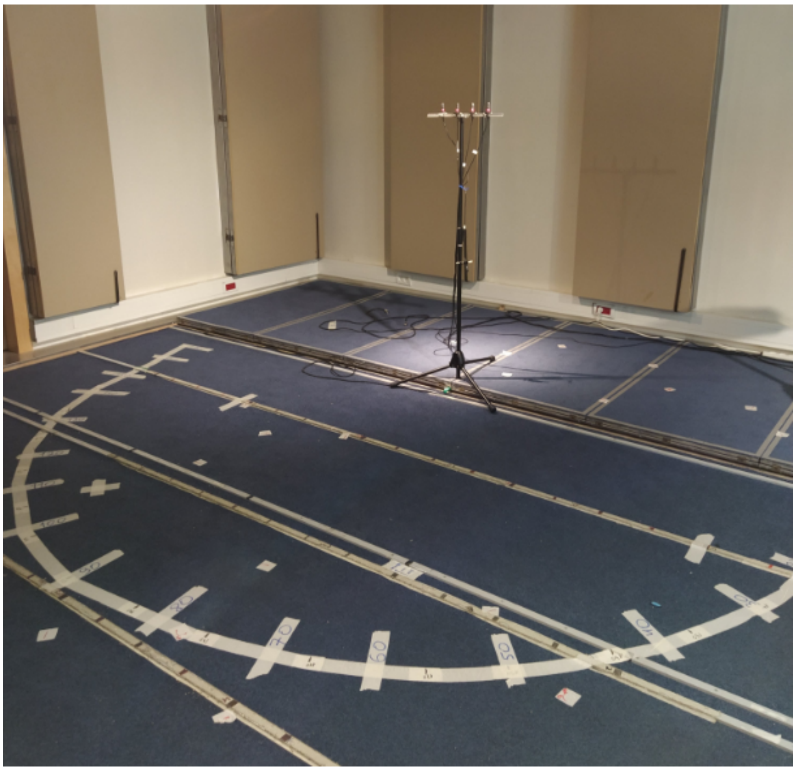
Table 4.5: Ablation study results with different features

Test Room	Room 1		Room 2	
Measure	MAE	Acc.	MAE	Acc.
Cos-Sin	1.2	96.1	2.8	91.3
TF-DOAnet with Spec.	0.6	98.4	3.3	86.7
TF-DOAnet	**0.3**	**99.5**	**1.7**	**94.3**

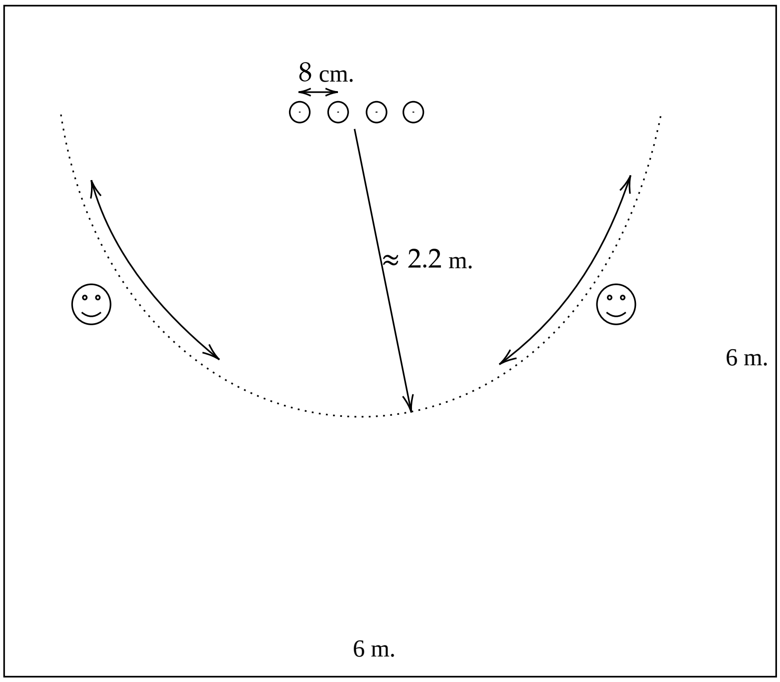
Broader impact

Several modern technologies can benefit from the proposed localization algorithm. We already mentioned the emerging technology of smart speakers in the Introduction. These devices are equipped with multiple microphones and are implementing location-specific tasks, e.g. the extraction of the speaker of interest. Of particular interest are socially assistive robots (SARs), as they are likely to play an important role in healthcare and psychological well-being, in particular during non-medical phases inherent to any hospital process.

The algorithm neither uses the content nor the identity of the speakers and hence does not to violate the privacy of the users. Moreover, since normally speech signal cannot propagate over long distances, the algorithm application is limited to small enclosures.

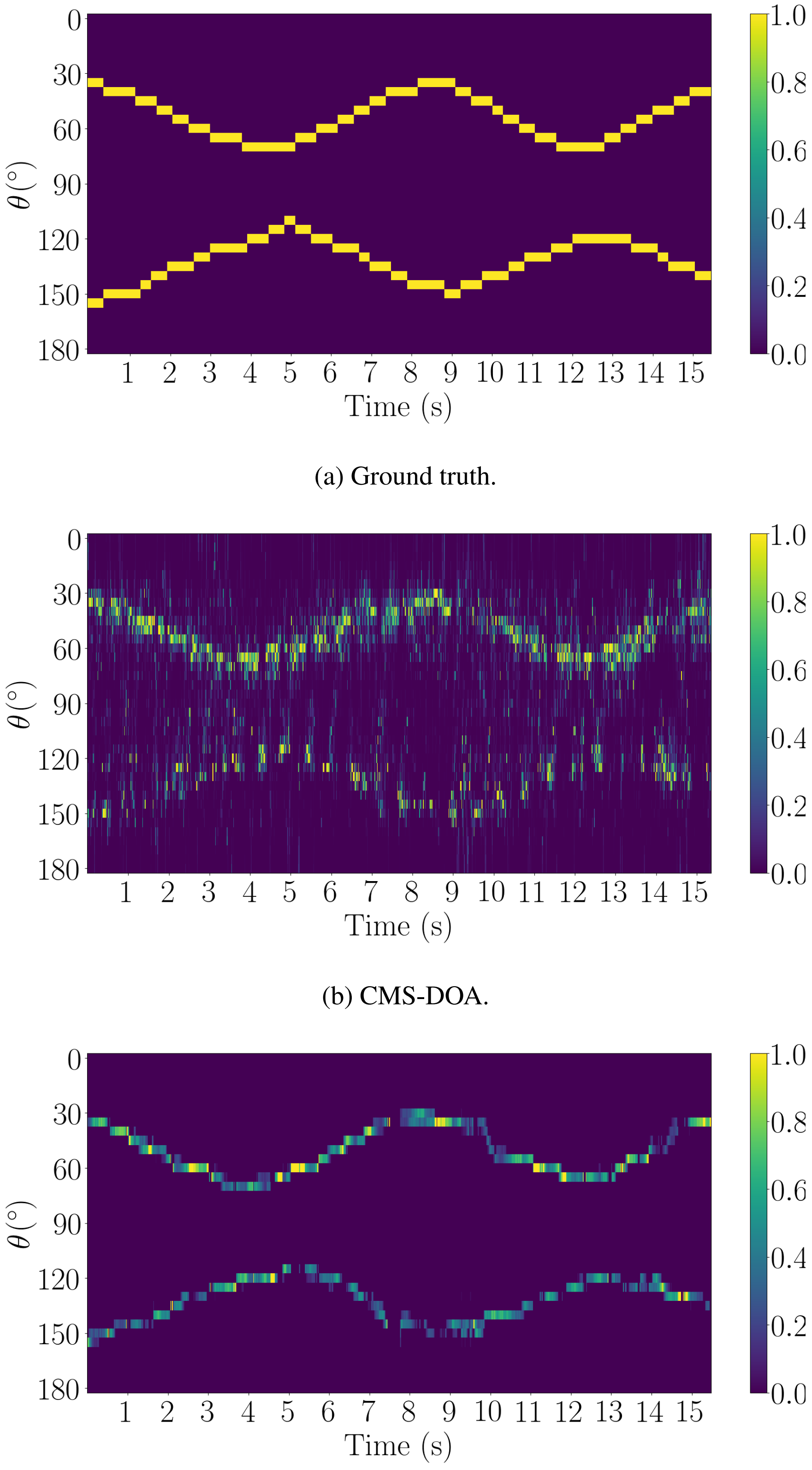


(a) Room view.



(b) Speakers' trajectory.

Figure 4.1: Real-life experiment setup.

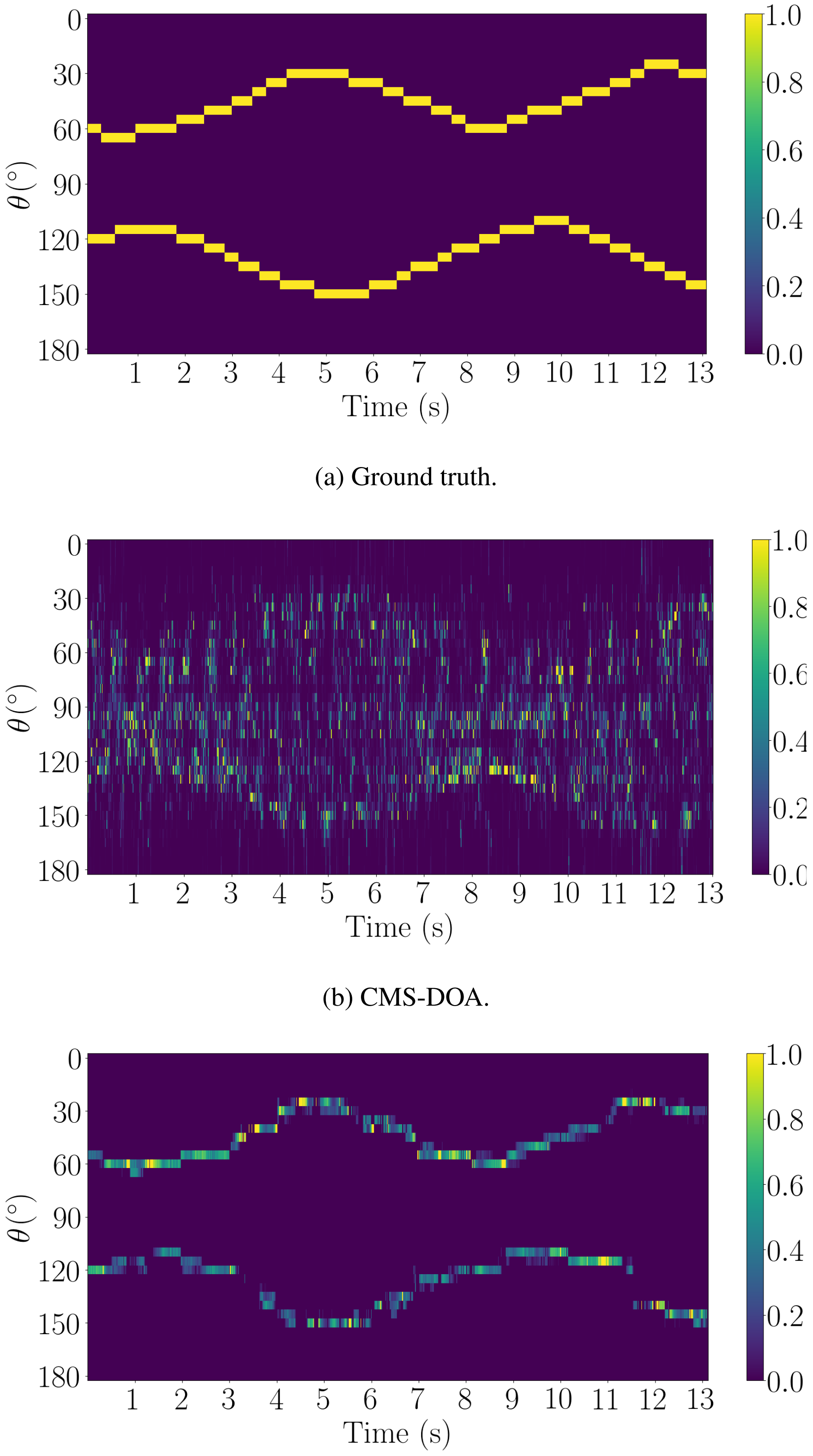


(a) Ground truth.

(b) CMS-DOA.

(c) The TF-DOAnet.

Figure 4.2: Real-life recording of two moving speakers in a $6 \times 6 \times 2.4$ room with $\mathrm{RT}_{60} = 390$ ms.

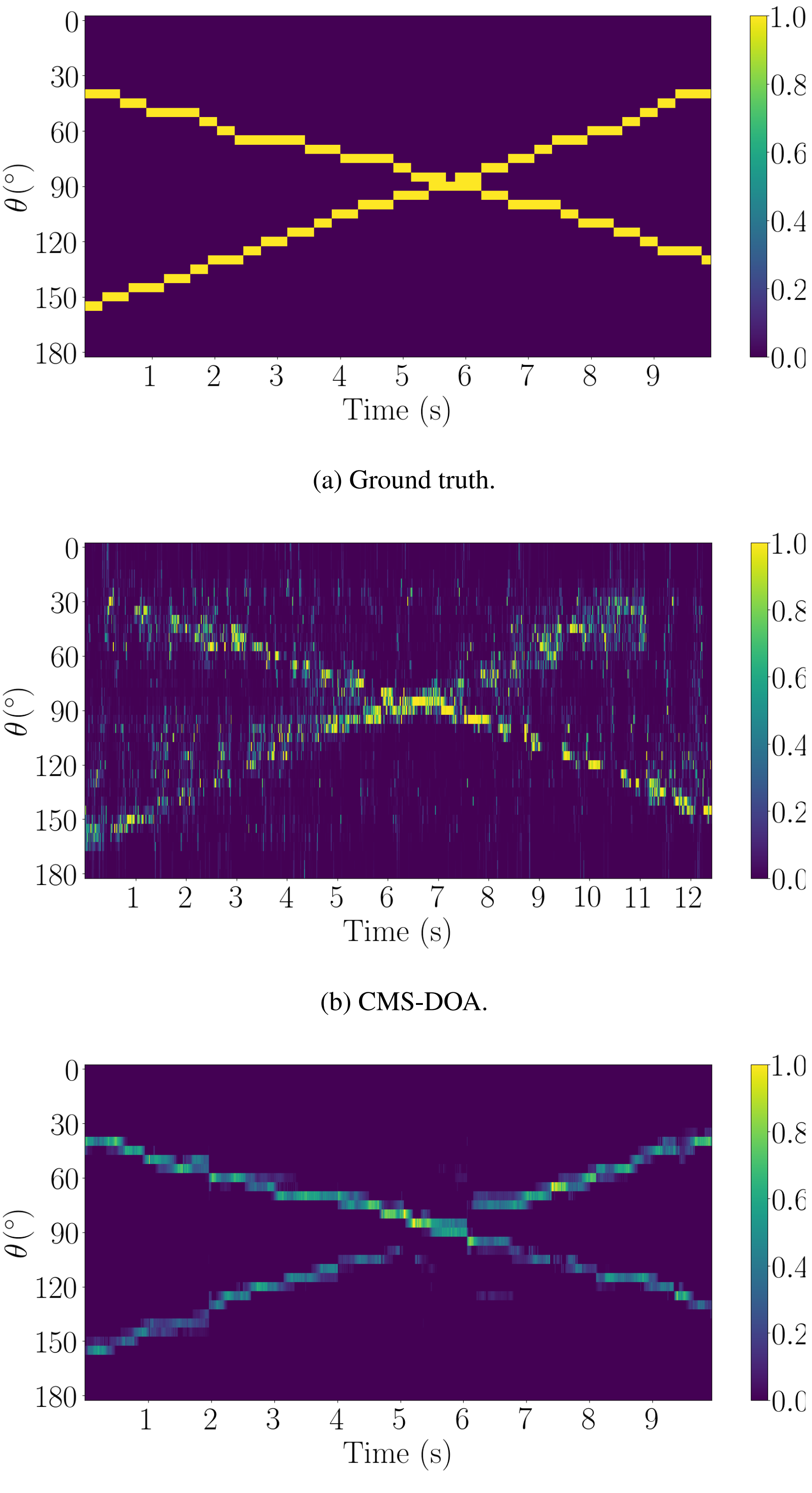


(a) Ground truth.

(b) CMS-DOA.

(c) The TF-DOAnet.

Figure 4.3: Real-life recording of two moving speakers in a $6 \times 6 \times 2.4$ room with $\mathrm{RT}_{60} =$ 720 ms.

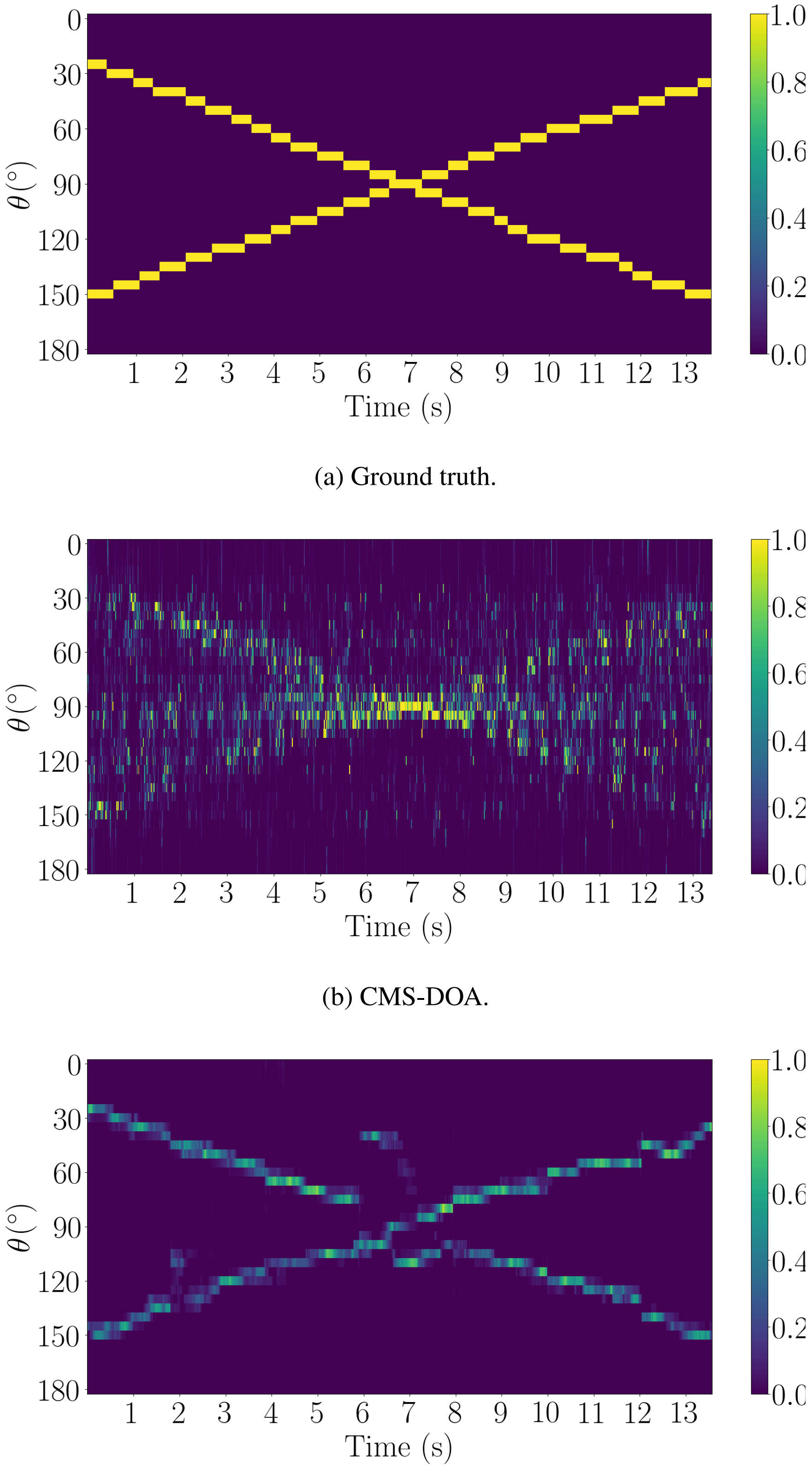


(a) Ground truth.

(b) CMS-DOA.

(c) The TF-DOAnet.

Figure 4.4: Real-life recording of two moving speakers, crossing each other, in a $6{\times}6{\times}2.4$ room with $\mathrm{RT}_{60} = 390$ ms.



(a) Ground truth.

(b) CMS-DOA.

(c) The TF-DOAnet.

Figure 4.5: Real-life recording of two moving speakers, crossing each other, in a $6 \times 6 \times 2.4$ room with $\mathrm{RT}_{60} = 720$ ms.

Chapter 5

Conclusions

In this thesis, we considered three different core problems in speech signal processing.

The MoG-DNN algorithm for speech enhancement: The proposed algorithm combines a model-based generative model for the clean speech signal with a discriminative, DNN-based SPP estimator. In this algorithm, we strive to use the advantages of model-based approaches along with the advantages of data driven DNN approaches and by that, turn an unsupervised problem into a supervised one. Additionally, we take advantage of the discriminative nature of the DNN that preserves speech smoothness by using context frames.

Speech separation based on DOA classification and masking: A DNN with a U-net architecture is trained to classify TF bins to DOAs. The association of each TF bin to specific DOA is used to construct spectral masks, which when applied to the spectrogram of the reference microphone obtain spectral source separation. The U-net was trained in a simulated room and tested with real RIR recordings, demonstrating the proposed algorithm capabilities in the task of blindly separating the sources.

Future directions:

1. Increase the robustness of the proposed algorithm to mismatch between train

and test conditions. This can be done by adding more scenarios to the training dataset, which can assist the DNN in learning specific features that are scenario independent.

2. Address dynamic scenarios and provide a trajectory estimate for the speakers. This can be done by generating a higher resolution dataset. Instead of discretizing the possible angles to be in the set $\Theta = \{0°, 15°, 30° \ldots, 180°\}$ we can discretize them to be in the set $\Theta = \{0°, 5°, 10° \ldots, 180°\}$. This way we can track the speakers movement more smoothly.

A FCN approach for DOA estimation: Instantaneous RTF features were used to train the model. The high TF resolution facilitated the tracking of multiple moving speakers simultaneously. A comprehensive experimental study was carried out with simulated and real-life recordings. The proposed approach outperformed both the classic and CNN-based SOTA algorithms in all experiments. Training and test datasets which represent different real-life scenarios were constructed as a DOA benchmark and will become available after publication.

Future directions:

1. It would be interesting to apply the high resolution DOA labels which we received using the TF-DOAnet algorithm in order to obtain better speech separation results.
2. We would like to re-examine the feature vector in the TF-DOAnet algorithm and find more specific and accurate features. This may lead to better results both for the source localization and the speech separation tasks.
3. Improve the TF-DOAnet algorithm, so it can be applied when the microphone array is moving on top of the speakers' movement.

Bibliography

[1] P. C. Loizou, *Speech enhancement: theory and practice.* CRC press, 2013.

[2] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Tran. on Acoustics, Speech and Signal Processing*, vol. 32, no. 6, pp. 1109–1121, 1984.

[3] ——, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Tran. on Acoustics, Speech and Signal Processing*, vol. 33, no. 2, p. 443–445, 1985.

[4] I. Cohen and B. Berdugo, "Speech enhancement for non-stationary noise environments," *Signal processing*, vol. 81, no. 11, p. 2403–2418, 2001.

[5] ——, "Noise estimation by minima controlled recursive averaging for robust speech enhancement," *IEEE Signal Processing Letters*, vol. 9, no. 1, pp. 12–15, Jan. 2002.

[6] Y. Wang and D. Wang, "Towards scaling up classification-based speech separation," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 21, no. 7, pp. 1381–1390, Jul. 2013.

[7] A. N. Yuxuan Wang and D. Wang, "On training targets for supervised speech separation," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1849–1858, Dec. 2014.

[8] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1702–1726, 2018.

[9] D. Burshtein and S. Gannot, "Speech enhancement using a mixture-maximum model," *IEEE Trans. on Speech and Audio Processing*, vol. 10, no. 6, pp. 341–351, Sep. 2002.

[10] S. E. Chazan, J. Goldberger, and S. Gannot, "A hybrid approach for speech enhancement using mog model and neural network phoneme classifier," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 12, pp. 2516–2530, Dec. 2016.

[11] A. Nádas, D. Nahamoo, and M. Picheny, "Speech recognition using noise-adaptive prototypes," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 37, no. 10, pp. 1495–1503, Oct. 1989.

[12] D. Rosenbaum and Y. Weiss, "The return of the gating network: Combining generative models and discriminative training in natural image priors," *Advances in Neural Information Processing Systems*, vol. 28, pp. 2683–2691, 2015.

[13] S. Gannot, E. Vincent, S. Markovich-Golan, and A. Ozerov, "A consolidated perspective on multimicrophone speech enhancement and source separation," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 25, no. 4, pp. 692–730, 2017.

[14] E. Vincent, T. Virtanen, and S. Gannot, Eds., *Audio Source Separation and Speech Enhancement*. Wiley, Sep. 2018.

[15] S. Makino, *Audio Source Separation*. Springer, 2018.

[16] S. Rickard and O. Yilmaz, "On the approximate w-disjoint orthogonality of speech," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2002.

[17] O. Yilmaz and S. Rickard, "Blind separation of speech mixtures via time-frequency masking," *IEEE Transactions on Signal Processing*, vol. 52, no. 7, pp. 1830–1847, 2004.

[18] Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-channel multi-speaker separation using deep clustering," *arXiv preprint arXiv:1607.02173*, 2016.

[19] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

[20] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[21] S. Rickard, *Blind speech separation*. Springer, 2007, vol. 615, ch. The DUET blind source separation algorithm, pp. 217–241.

[22] T. Higuchi, K. Kinoshita, M. Delcroix, K. Zmolkova, and T. Nakatani, "Deep clustering-based beamforming for separation with unknown number of sources," in *Proc. Interspeech*, 2017.

[23] L. Drude and R. Haeb-Umbach, "Tight integration of spatial and spectral features for BSS with deep clustering embeddings," in *Proc. Interspeech*, 2017.

[24] P. Pertilä and J. Nikunen, "Distant speech separation using predicted time–frequency masks from spatial features," *Speech Communication*, vol. 68, pp. 97–106, 2015.

[25] Y. Yu, W. Wang, and P. Han, "Localization based stereo speech source separation using probabilistic time-frequency masking and deep neural networks," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2016, no. 1, p. 7, 2016.

[26] E. Tzinis, S. Venkataramani, and P. Smaragdis, "Unsupervised deep clustering for source separation: Direct learning from mixtures using spatial information," *arXiv preprint arXiv:1811.01531*, 2018.

[27] P. Seetharaman, G. Wichern, J. L. Roux, and B. Pardo, "Bootstrapping single-channel source separation via unsupervised spatial clustering on stereo mixtures," *arXiv preprint arXiv:1811.02130*, 2018.

[28] S. E. Chazan, J. Goldberger, and S. Gannot, "DNN-based concurrent speakers detector and its application to speaker extraction with LCMV beamforming," in *IEEE International Conference on Audio and Acoustic Signal Processing (ICASSP)*, 2018.

[29] S. E. Chazan, S. Gannot, and J. Goldberger, "Attention-based neural network for joint diarization and speaker extraction," in *IEEE International Workshop on Acoustic Signal Enhancement (IWAENC)*, 2018.

[30] Z.-Q. Wang, J. L. Roux, and J. R. Hershey, "Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[31] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

[32] S. Chakrabarty and E. A. P. Habets, "Multi-speaker DOA estimation using deep convolutional networks trained with noise signals," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 1, pp. 8–21, 2019.

[33] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[34] J. P. Dmochowski, J. Benesty, and S. Affes, "Broadband music: Opportunities and challenges for multiple source localization," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2007.

[35] M. S. Brandstein and H. F. Silverman, "A robust method for speech signal time-delay estimation in reverberant rooms," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1997.

[36] X. Xiao, S. Zhao, X. Zhong, D. L. Jones, E. S. Chng, and H. Li, "A learning-based approach to direction of arrival estimation in noisy and reverberant environments," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[37] F. Vesperini, P. Vecchiotti, E. Principi, S. Squartini, and F. Piazza, "A neural network based algorithm for speaker localization in a multi-room environment," in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2016.

[38] R. Takeda and K. Komatani, “Discriminative multiple sound source localization based on deep neural networks using independent location model,” *IEEE Spoken Language Technology Workshop (SLT)*, 2016.

[39] H. Pujol, E. Bavu, and A. Garcia, “Source localization in reverberant rooms using deep learning and microphone arrays,” in *International Congress on Acoustics (ICA)*, 2019.

[40] J. M. Vera-Diaz, D. Pizarro, and J. Macias-Guarasa, “Towards end-to-end acoustic localization using deep learning: From audio signals to source position coordinates,” *Sensors*, vol. 18, no. 10, p. 3418, 2018.

[41] S. Chakrabarty and E. A. P. Habets, “Broadband DOA estimation using convolutional neural networks trained with noise signals,” *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2017.

[42] C. Evers, H. Loellmann, H. Mellmann, A. Schmidt, H. Barfuss, P. Naylor, and W. Kellermann, “The locata challenge: Acoustic source localization and tracking,” *arXiv preprint arXiv:1909.01008*, 2019.

[43] E. Hadad, F. Heese, P. Vary, and S. Gannot, “Multichannel audio database in various acoustic environments,” in *International Workshop on Acoustic Signal Enhancement (IWAENC)*, 2014.

[44] H. Hammer, G. Rath, S. E. Chazan, J. Goldberger, and S. Gannot, “Speech enhancement with deep neural networks using mog based labels,” in *2018 IEEE International Conference on the Science of Electrical Engineering in Israel (ICSEE)*, 2018.

[45] S. E. Chazan, H. Hammer, G. Hazan, J. Goldberger, and S. Gannot, “Multi-microphone speaker separation based on deep doa estimation,” in *2019 27th European Signal Processing Conference (EUSIPCO)*, 2019.

[46] S. E. Chazan, S. Gannot, and J. Goldberger, “A phoneme-based pre-training approach for deep neural network with application to speech enhancement,” in *IEEE International Workshop on Acoustic Signal Enhancement (IWAENC)*, 2016.

[47] S. T. Roweis, “One microphone source separation,” in *Neural Information Processing Systems (NIPS)*, vol. 13, 2000, pp. 793–799.

[48] M. Radfar and R. Dansereau, “Single-channel speech separation using soft mask filtering,” *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2299–2310, Nov. 2007.

[49] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 1–38, 1977.

[50] A.-R. Mohamed, D. Yu, and L. Deng, “Investigation of full-sequence training of deep belief networks for speech recognition.” in *Proceedings of the annual conference of the International Speech Communication Association (INTERSPEECH)*, 2010, pp. 2846–2849.

[51] J. Garofalo, L. Lamel, W. Fisher, J. G. Fiscus, D. Pallett, and N. Dahlgren, "The DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM," Linguistic Data Consortium, Tech. Rep., 1993.

[52] A. Varga and H. J. Steeneken, "Assessment for automatic speech recognition: Ii. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech communication*, vol. 12, no. 3, pp. 247–251, 1993.

[53] "Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," ITU-T, Recommendation ITU-T P.862, 2001.

[54] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[55] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[56] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[57] D. B. Paul and J. M. Baker, "The design for the Wall Street Journal-based CSR corpus," in *Workshop on Speech and Natural Language*, 1992. [Online]. Available: https://www.aclweb.org/anthology/H92-1073

[58] C. Févotte, R. Gribonval, and E. Vincent, "BSS_EVAL toolbox user guide–revision 2.0," online, 2005.

[59] S. Gannot, D. Burshtein, and E. Weinstein, "Signal enhancement using beamforming and nonstationarity with applications to speech," *IEEE Transactions on Signal Processing*, vol. 49, no. 8, pp. 1614–1626, 2001.

[60] S. Stenzel, J. Freudenberger, and G. Schmidt, "A minimum variance beamformer for spatially distributed microphones using a soft reference selection," in *Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA)*, 2014.

[61] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, 2015.

[62] S. Markovich-Golan, S. Gannot, and I. Cohen, "Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals," *IEEE Tran. on Audio, Speech, and Language Processing*, vol. 17, no. 6, pp. 1071–1086, Aug. 2009.

[63] I. Dokmanić, R. Scheibler, and M. Vetterli, "Raking the cocktail party," *IEEE journal of selected topics in signal processing*, vol. 9, no. 5, pp. 825–836, 2015.

[64] A. Deleforge, F. Forbes, and R. Horaud, “Acoustic space learning for sound-source separation and localization on binaural manifolds,” *International journal of neural systems*, vol. 25, no. 01, p. 1440003, 2015.

[65] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, “Semi-supervised sound source localization based on manifold regularization,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 8, pp. 1393–1407, 2016.

תקציר

בעבודה זו אנו מציגים אלגוריתמים לפתרון שלוש בעיות בעיבוד אותות דיבור. האלגוריתמים מתבססים על תכונת הדלילות של הדיבור במרחב הזמן-תדר ועושים שימוש ברשתות נוירונים.

ניקוי אותות דיבור מרעש עם מיקרופון יחיד

אנו מציגים אלגוריתם MoG-DNN. אנו משלבים בין Mixture of Gaussians (MOG) שהוא מודל גנרטבי לבין רשת נוירונים שהיא מודל דיסקרימינטיבי. האלגוריתם מכיל שני שלבים, שלב האימון ושלב הבחינה. בשלב האימון, נמדל את צפיפות ההספק הספקטרלית של הדיבור הנקי והרעש באמצעות MoG המשמש למיון בלתי-מונחה של אותות הדיבור ועל ידי גאוסיאן יחיד בהתאמה. בהתאם ל-MoG הנתון, נתייג את מערך הנתונים. לאחר מכן, נאמן רשת נוירונים על מנת לסווג את מסגרות הזמן של הדיבור המורעש לאחד מהגאוסיאנים מבין גאוסיאני ה-MoG. בשלב הבחינה, נחשב את ההסתברות לקיום דיבור (SPP) ונבצע חיסור ספקטרלי במקביל לעדכון פרמטרי הרעש. מודל MoG שומר על המבנה הספקטרלי של הדיבור ובנוסף הוא מודל גנרטיבי, בלתי-מונחה ולכן ניתן ליישם אותו על כל מערך נתונים בלתי ידוע. רשת הנוירונים הדיסקרימינטיבית שומרת על רציפות הדיבור.

הפרדת דוברים רב-מיקרופוני

אנו מציגים אלגוריתם DDESS המתבסס על מסכה המחושבת מתוך כיוון ההגעה (DOA) של הדובר. על פי תכונת ה- W-disjoint orthogonality של אותות דיבור, בכל נקודת זמן-תדר רק דובר יחיד הוא דומיננטי. לכן, כל נקודת זמן-תדר יכולה להשתייך ל-DOA יחד. באלגוריתם הנתון, נשתמש ברשת נוירונים עם ארכיטקטורת U-net, כאשר הכניסה לרשת היא שרשור הספקטרום של אותות המיקרופונים והמוצא הוא ה-DOA בכל נקודת זמן-תדר. לאחר מכן, נייצר את המסכות עבור כל DOA ונבצע הפרדה על ידי הכפלת האות ממיקרופון הייחוס עם כל אחת מהמסכות.

איכון דוברים רב-מיקרופוני

אנו מציגים אלגוריתם TF-DOAnet. גם כאן, אנו מתבססים על העיקרון שמניח שכל נקודת זמן-תדר נשלטת על ידי דובר יחיד ולכן משוייכת לDOA יחיד. באלגוריתם הנתון, נאמן רשת נוירונים עם ארכיטקטורה זהה לזו שבאלגוריתם DDESS אלא שכאן הרזולוציה גבוהה בהרבה ובכך מאפשרת לרשת לאכן בצורה מדוייקת כמה דוברים במקביל, כאשר הדוברים יכולים להיות סטטיים או דינמיים.

ביצענו סימולציות רבות וניסויי מעבדה המדגימים כי האלגוריתמים המוצעים משיגים תוצאות טובות מהתוצאות המושגות על ידי אלגוריתמים מתחרים מובילים.

עבודה זו נעשתה בהדרכתו של פרופ׳ שרון גנות

מהפקולטה להנדסה של אוניברסיטת בר-אילן

אוניברסיטת בר-אילן

אלגוריתמים בעיבוד אותות דיבור המתבססים על תכונת הדלילות של אותות דיבור במרחב הזמן-תדר ועושים שימוש ברשתות נוירונים

הודיה המר

עבודה זו מוגשת כחלק מהדרישות לשם קבלת תואר מוסמך
בפקולטה להנדסה של אוניברסיטת בר-אילן

רמת גן תש״פ