Deliverable D3.3: Audio speaker diarisation and extraction with a static robot

Due Date: 01/03/2023

Main Author: Sharon Gannot (BIU)

Contributors: -

Dissemination: Public Deliverable

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

DOCUMENT FACTSHEET

Deliverable	D3.3: Audio speaker diarisation and extraction with a static robot
Responsible Partner	BIU
Work Package	WP3: Robust Audio-visual Perception of Humans
Task	T3.2: Extraction of Desired Sources (Static Robot)
Version & Date	01/03/2023
Dissemination	Public Deliverable

CONTRIBUTORS AND HISTORY

Version	**Editor**	**Date**	**Change Log**
1	BIU	01/03/2023	First Draft

APPROVALS

Authors/editors	Sharon Gannot (BIU)
Task Leader	BIU
WP Leader	BIU

This project has received funding from the European Union's Horizon 2020
Research and Innovation Programme under Grant Agreement No. 871245.

Contents

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

Executive Summary

Deliverable 3.3 reports the progress on task T3.2 on Audio speaker diarisation and extraction with a static robot, which is part of WP3: Robust Audio-visual Perception of Humans. The goal of task 3.2 is to provide several separated audio streams to be transcribed by the automatic speech recognition (ASR) and fed to the multi-party conversational system that will be deployed on ARI, the robotic platform designed by PAL Robotics for the SPRING project.

The main achievements reported in this document are:

1. Single-microphone speaker extraction algorithm.
2. Multi-microphone beamformer-based speaker separation algorithm.

Both methods are extensively tested with common databases and also with real recordings from ARI.

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

1 Introduction

This deliverable is part of WP3 of the H2020 SPRING project. The objective of WP3 is "the robust extraction, from the raw auditory and visual data, of users' low-level characteristics, namely: position, speaking status and speech signal." Following this objective, WP3 has two main outcomes:

1. The Multi-Person Tracking module, jointly exploiting auditory and visual raw data to detect, localise and track multiple speakers (corresponds to T3.1).
2. The Diarisation and Separation and the Speech Recognition modules, extracting the desired speaker(s) from a speech dynamic mixture and recognising the speech utterances from the separated sources, for a static T3.2 and a moving T3.3 robot

In this deliverable, we report on two algorithms for extracting the speaker of interest from a mixture of multiple speakers.

Single microphone speaker extraction: This algorithm uses a reference signal, which is an utterance from one of the speakers in the mixed signal. Such a reference can be obtained from segments for which only a single speaker is active.[1]

Multi-microphone speaker separation: This algorithm applies a linearly constrained minimum variance (LCMV) beamformer that directs a beam towards the desired speaker while directing a null towards the competing speakers. Simultaneously, the algorithm detects the activities of the speakers and localizes them.[2]

From a system perspective, the separation/extraction algorithm will have multiple outputs of separated signals. An ASR will be applied to each signal and a text stream will be published for further processing with the dialogue system. For the multi-microphone algorithm, each speaker will also be tagged by its direction relative to ARI.

[1] Available at https://gitlab.inria.fr/spring/wp3_av_perception/1ch_speaker_extraction
[2] Will be available on June 1, 2023, at https://gitlab.inria.fr/spring/wp3_av_perception/4ch_lcmv_bf

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

2 Single microphone speaker extraction using unified time-frequency Siamese-Unet

We present a unified time-frequency method for speaker extraction in clean and noisy conditions. Given a mixed signal, along with a reference signal, the common approaches for extracting the desired speaker are either applied in the time domain or in the frequency domain. In our approach, we apply a Siamese-Unet architecture that uses both representations. The Siamese encoders are applied in the frequency domain to infer the embedding of the noisy and reference spectra, respectively. The concatenated representations are then fed into the decoder to estimate the real and imaginary components of the desired speaker, which are then inverse-transformed to the time domain. The model is trained with the Scale-Invariant Signal-to-Distortion Ratio (SI-SDR) loss to exploit the time-domain information. The time-domain loss is also regularized with frequency-domain loss to preserve speech patterns.

2.1 Introduction

Extracting a desired speaker from a mixture of overlapping speakers is a challenging task that is usually solved using microphone array processing [14]. With a single microphone, no spatial information is available thus making the task even more challenging. Here, we address the single-microphone speaker extraction task and focus on the extraction of a single participant from a mixed signal, given a pre-recorded sample of the speaker to be extracted.

In recent years significant progress has been achieved in the field of speaker separation. The Conv-Tasnet algorithm [31] is applied in the time domain. Self-learned representations of the signal are inferred using 1-D conventional layers. The model estimates a mask for each speaker, which is then applied in the learned representations domain for the separation task. The gist of this algorithm is the use of the SI-SDR loss function [28], which is designed to exploit the time-domain information. The authors show that by being independent of the traditional hand-crafted features, improved performance is obtained. The dual-path recurrent neural network (DPRNN) algorithm, also applied in the time-domain, was presented in [30]. The mixing signal is split into overlapped chunks and processed using intra-chunk and inter-chunk Recurrent Neural Networkss (RNNs). The performance of the algorithm was evaluated with the WSJ0-2mix database. In [7] an algorithm that can successfully process mixtures with a larger number of speakers is presented. Moreover, it can also work in noisy and reverberant scenarios. The algorithm comprises a multi-head architecture, jointly trained with a gate. Each head is responsible to separate a different number of speakers. The gate, which classifies the number of speakers in the mixture, determines which of the heads should be applied to the mixture.

While these algorithms demonstrate promising results, they suffer from excess computational complexity, due to the need to train the feature extraction stage, rather than to utilize the traditional time frequency (TF) representation. Additionally, the permutation problem [24] must be taken into consideration during training. Finally, additional information regarding the specific characteristics of the desired speaker may facilitate its extraction.

Recently, different architectures were proposed to extract the desired speaker given a reference signal. They can be roughly split into TF-domain methods and time-domain methods. In the TF-domain, most models are using masking operation on the mixed signal [20,29,42,43,45,50]. In [43] a pre-trained d-vector [41] is utilized as an embedding of the reference signal. A mask is then estimated given the mixed signal and the reference embedding vector. Finally, the short time Fourier transform (STFT) of the noisy signal is multiplied with the mask to extract the desired speaker. Note that the phase of the mixed signal is not processed, and only the spectrogram of the desired speaker is extracted.

The permutation problem is not an issue in the problem of speaker extraction, as a prior information on the desired signal is available. Yet, since applied in the TF domain, these methods use the Mean Square Error (MSE) loss as their training objective rather than the time-domain SI-SDR loss, which is perceptually more meaningful.

These drawbacks, namely the use of the MSE loss and of the noisy phase, have led to a series of models applied directly in the time-domain [9,10,15,19,46,47,49]. Inspired by the time-domain Blind Source Separation (BSS) algorithms, the architecture of these methods comprises an encoder block, a separation block (usually based on a proven BSS architecture) and a decoder block to implement the inverse-transform of the desired signal back to the time-domain.

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

These methods, similar to the time-domain BSS algorithms, also utilize the SI-SDR loss function [28] as their objective. Unfortunately, similar to the time-domain BSS methods, these extraction methods suffer from two main drawbacks: 1) thay are not easy to implement, and 2) they ignore the specific TF patterns of the speech signal.

In this work, we propose a Siamese-Unet architecture that uses both representations, the TF features as input and output features to the network along with the time-domain representation for computing the SI-SDR loss. Our model is constructed with a two-head encoder, one for the reference signal and the other for the mixed signal. The estimated embeddings are then concatenated as input to the decoder, to extract the desired speaker. Similar to other speech enhancement methods, e.g. [21], the Real-Imaginary (RI) components of the STFT are utilized as inputs, while the waveform is utilized as the target of the network. The RI components are inverse-transformed to the time domain and the SI-SDR loss is used to train the model. A comprehensive simulation study using common databases demonstrates the benefits of the proposed scheme.

2.2 Problem Formulation

Let $x(t)$ be a mixture of I concurrent speakers captured by a single microphone:

$$x(t) = \sum_{i=1}^{I} \{s_i * h_i\}(t) + n(t) \quad t = 0, 1, \ldots, T-1\,, \tag{2.1}$$

where $s_i(t)$ represents the signal of the i-th speaker, $h_i(t)$ represents the Room Impulse Response (RIR) between the i-th speaker and the microphone, and $n(t)$ represents the additive noise. Note that in a noiseless and anechoic enclosure, $h_i(t) = \delta(t)$, $i = 1, \ldots, I$ and $n(t) = 0$. In the STFT domain (2.1) can be reformulated as,

$$x(l,k) = \sum_{i=1}^{I} s_i(l,k) \cdot h_i(l,k) + n(l,k), \tag{2.2}$$

where $l \in \{0, \ldots, L-1\}$ and $k \in \{0, \ldots, K-1\}$ are the time-frame and the frequency-bin (TF) indexes, respectively. The terms L and K represent the total number of time-frames and frequency bands, respectively.

For simplicity, we address in this work the $I = 2$ case and denote the desired speaker as $s_d(l,k)$, the reference signal as $s_r(l,k)$, and the interference speaker as $s_i(l,k)$. The output of the proposed algorithm is $\hat{s}_d(l,k)$, an estimate of $s_d(l,k)$ given the mixed signal (2.2) and a *reverberant* reference signal $s_r(l,k) \cdot h_r(l,k)$.

2.3 Proposed Model

In this section we introduce the proposed novel Siamese-Unet architecture for desired speaker extraction, given a reference recording.

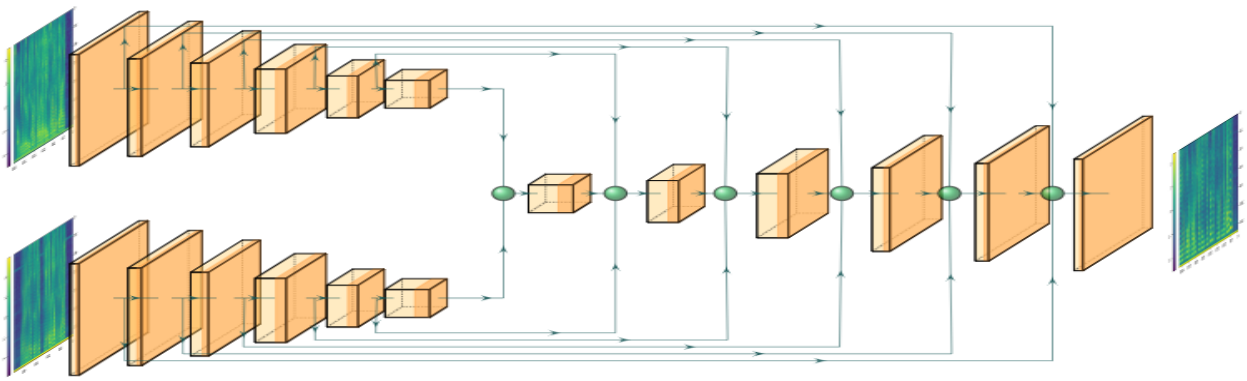


Figure 2.1: The proposed architecture. The green circles stands for the concatenation operation. To calculate the SI-SDR loss, inverse short-time Fourier transform (ISTFT) is applied to the model's output. The inputs and outputs features of the network are the RI components of the mixture, reference and estimated signals, respectively.

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

2.3.1 Architecture

The proposed Siamese-Unet model comprises a two-head encoder and a decoder. Skip connections are concatenated between the layers of the encoders and the decoder sections. The proposed architecture is summarized in Fig. 2.1.

In our approach, the reference signal and the mixed signal are first projected to the same latent space by the two-head encoder. The encoder's architecture is constructed with seven convolution layers, each layer followed by a two-dimensional batch normalization and a 'Relu' function. The decoder architecture is similar to the encoder architecture, but instead of the convolution layers, transpose-convolution layers are applied. Denote $\text{CBR}_{i,o}$ and $\text{TCBR}_{i,o}$ as the Convolution-BatchNormalization-Relu and the Transpose-Convolution-BatchNormalization-Relu layers, respectively, where i and o are the number of the input and output channels, respectively. The size of the filters in each layer is set to 4, the stride size is set to 2 and the padding value is set to 1.

The encoder path is given by: $\text{CBR}_{64,128} \rightarrow \text{CBR}_{128,256} \rightarrow \text{CBR}_{256,512} \rightarrow \text{CBR}_{512,512} \rightarrow \text{CBR}_{512,512} \rightarrow \text{CBR}_{512,512} \rightarrow \text{CBR}_{512,512}$ and the decoder path is given by: $\text{TCBR}_{1024,512} \rightarrow \text{TCBR}_{1536,512} \rightarrow \text{TCBR}_{1536,512} \rightarrow \text{TCBR}_{1536,256} \rightarrow \text{TCBR}_{768,128} \rightarrow \text{TCBR}_{384,64} \rightarrow \text{TCBR}_{192,2}$

Finally, an additional convolution-layer is applied to obtain the desired signal estimate.

Different alternatives for integrating the information from the reference signal are described in [50]. Two comments regarding the implementation are in place: 1) when using the skip connections in the U-net architecture, concatenating the encoder layers and the decoder layers, rather than multiplying them, yields better results, and 2) concatenating all intermediate layers of the reference encoder using skip connections (rather than only the bottleneck layer) in parallel to the skip connections of the mixture encoder, improves separation performance. While the majority of STFT-domain algorithms are applying a mask to the mixed signal, our proposed network is directly trained to estimate the TF representation of the target source.

2.3.2 Features

As mentioned above, the gist of this work is the utilization of both the TF and the time-domain information. Most of the approaches applied in the STFT domain use the noisy phase for calculating the inverse-transform back to the time domain, since estimating the phase is a cumbersome task. Unfortunately, the performance of such approaches is limited even if the spectrogram is perfectly estimated, especially in reverberant environment. Instead, we propose to use the RI components as both the input features and the model's target. In this way, we circumvent the inaccuracies that result from applying the inverse-STFT with the noisy phase.

2.3.3 Objectives

To train the proposed Siamese-Unet for the extraction task, the time-domain SI-SDR loss function, which was found to be most appropriate for BSS tasks, is used. The loss is formulated as,

$$\text{SI-SDR}\left(s, \hat{s}\right) = 10\log_{10}\left(\frac{\left\|\frac{\langle \hat{s}, s\rangle}{\langle s, s\rangle} s\right\|^2}{\left\|\frac{\langle \hat{s}, s\rangle}{\langle s, s\rangle} s - \hat{s}\right\|^2}\right) \tag{2.3}$$

where $\langle , \rangle$ is the inner product, s is the target speaker in the time domain and $\hat{s}$ is the estimated speaker.

To further improve the training, we used, for each training sample, the same mixture and swapped the desired and interference signals. The corresponding reference signal was used for each of the extracted sources. The two losses are then averaged,

$$L_{\text{SISDR}} = 0.5 \cdot \left[\text{SISDR}\left(s_1, \hat{s}_1\right) + \text{SISDR}\left(s_2, \hat{s}_2\right)\right]. \tag{2.4}$$

We also add the MSE loss as a regularization term to the SI-SDR loss, considering the RI features,

$$L_{\text{MSE}} = 0.5 \cdot \left[\text{MSE}(\text{RI}_1, \widehat{\text{RI}_1}) + \text{MSE}(\text{RI}_2, \widehat{\text{RI}_2})\right]. \tag{2.5}$$

Our final training loss is a weighted sum of the main loss and its regularization term:

$$L = \beta_{\text{SISDR}} \cdot L_{\text{SISDR}} + \beta_{\text{MSE}} \cdot L_{\text{MSE}} \tag{2.6}$$

with $\beta_{\text{SISDR}} + \beta_{\text{MSE}} = 1$.

In this work we aim to combine both time and time-frequency representations in the training of the network. By doing so, we preserve the TF patterns of the speech signal, while still optimizing the perceptually meaningful SI-SDR loss. As a byproduct, we found that this method is easier to implement and that its training time is faster than the respective training time of algorithms with only time-domain loss.

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

2.4 Experimental study

In this section, we describe the experimental setup, the train and test datasets, and the obtained results. The performance of the proposed algorithm and the competing methods is reported for both clean conditions and for mild noise and reverberation conditions.

2.4.1 Datasets

To train our model, we constructed a dataset of mixed signals. Each sample in the dataset consists of a simulated mixture, two reference signals and two clean signals, used as targets to the model.

Dataset of mixed signals in clean conditions: The clean speech signals were randomly drawn from the LibriSpeech corpus [34] and the WSJ0 corpus [35]. Each corpus was randomly split, with 80% of the speakers taken for the training dataset, 10% for validation dataset and 10% for test. The signals in the training phase were randomly truncated or zero-padded to a duration of 2-8 seconds. The duration is kept fixed for each batch and may vary between batches. In the test phase, the mixed signals were not truncated. If the reference utterance is shorter than the mixed signal, it is repeated until it fits the duration of the mixture. A total of 50,000 training samples, 10,000 validation samples and 3000 test samples were simulated, with the gender of the speakers uniformly selected. Finally, the signals are summed up to generate the mixing signal.

Dataset of mixed signals in mild noise and reverberation conditions: We also constructed a noisy and reverberant dataset. Signals were randomly drawn from the LibriSpeech and WSJ0, following a similar procedure to the construction of the clean dataset. Each signal was convolved with a simulated RIR using the RIR generator tool [16], with randomly chosen acoustic conditions, such as room dimensions, microphone and speaker positions, and reverberation level. The parameters controlling the acoustic conditions can be found in Table 2.2. The noise signals were drawn from the WHAM! corpus [44], which consist of babble noises from different environments (such as restaurants, cafés, bars, and parks) and added to the clean mixtures with random Signal To Noise Ratio (SNR) in the range of [10,25] dB. Note that these acoustic conditions represent low reverberation conditions and relatively high SNR.

2.4.2 Algorithm Settings

The speech and noise signals are downsampled to 8 [KHz]. The frame-size of the STFT is 256 samples with 75% overlap. Due to the symmetry of the Discrete Fourier Transform (DFT) only the first half of the frequency bands is used. β_{SISDR} was empirically set to 0.75 to give a preference to the SI-SDR loss.

In the training procedure, we used the Adam optimizer [22]. The learning rate was set to 0.001 and the training batch size to 16. The weights are randomly initialized, and the lengths of the signals were randomly changed at each batch.

2.4.3 Evaluation Measures

To evaluate the proposed algorithm we use three evaluation measures: 1) the SI-SDR, as mentioned above, 2) the signal to interference (SIR), and 3) the signal direction recognition (SDR). These evaluation measures are widely used for BSS tasks. Note that for the SI-SDR measure we present the improvement (SI-SDRi). The proposed algorithm is compared to the commonly used VoiceFilter [43] algorithm and to a recently proposed BSS method [7] that demonstrates high performance even in reverberant and noisy conditions. Additionally, we tested a variant of the proposed method in which, rather than the RI features, only the log-spectrum is estimated while the noisy phase is used. Finally, an oracle solution was generated by using the target log-spectrogram and the noisy phase. This oracle solution provides the best achievable performance by the masking-based procedure.

2.4.4 Results

Clean conditions: The SI-SDRi results for the clean test dataset are depicted in Fig. 2.2. First, it is easy to verify that the reference signal indeed assists the extraction task. Second, it is clear that the proposed approach outperforms the VoiceFilter algorithm. Finally, it can be deduced that the RI features are more suitable to the task at hand than the log-spectrum (LS) features.

The SIR and SDR values are presented in Table 2.1. It is evident that the proposed algorithm (RI variant) and the algorithm in [7] perform similarly in terms of SIR measures. However, in the SDR measure, the proposed algorithms clearly outperform the algorithm in [7], implying lower distortion.

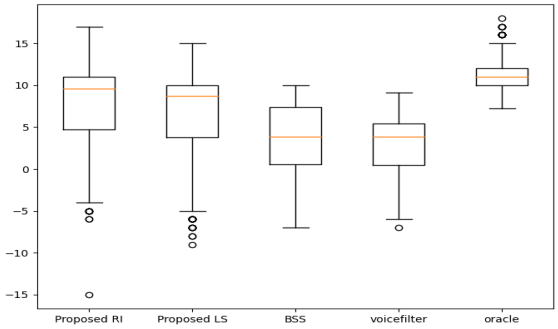
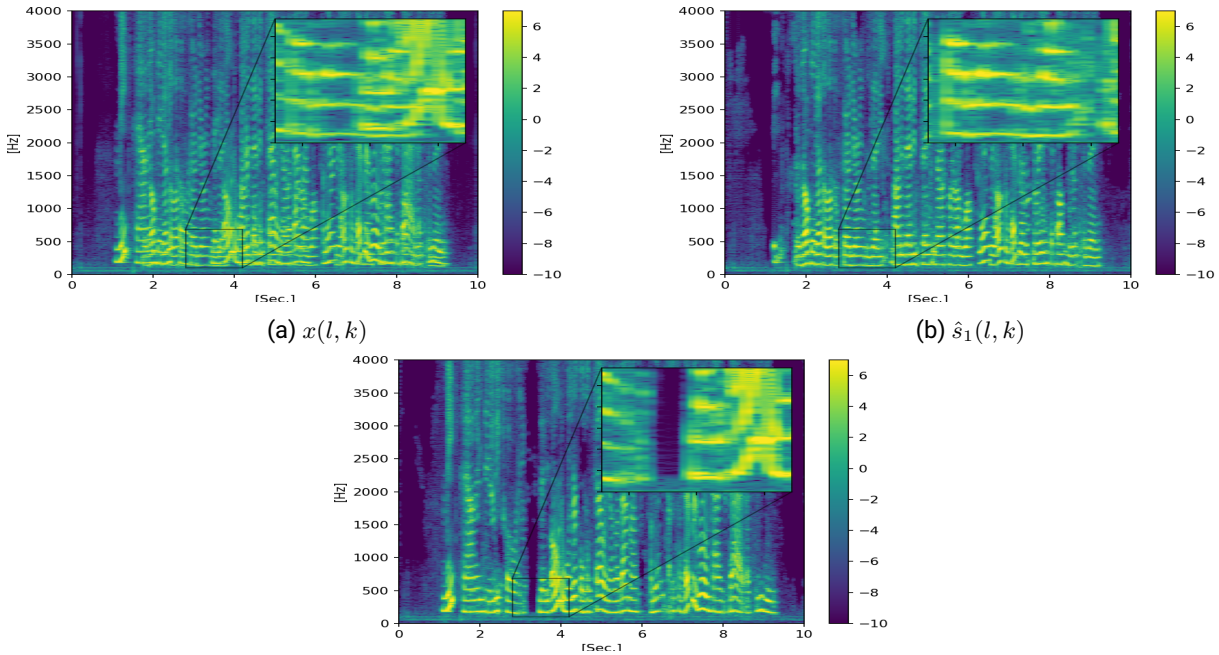


Figure 2.2: SI-SDRi comparison between the models for the clean test dataset. Two variants of the proposed method, (real-imaginary (RI) and log-spectrogram (LS) features) are compared to VoiceFilter and to the Oracle masking-based method.

Table 2.1: SIR and SDR results (the higher the better).

Model	Mixture	BSS [7]	Proposed (LS)	Proposed (RI)
SIR	0.1	**15.5**	14.7	15.4
SDR	0.1	5.73	7.9	**8.22**



(a) $x(l,k)$ (b) $\hat{s}_1(l,k)$

(c) $\hat{s}_2(l,k)$

Figure 2.3: Real mixture recording and the extraction of each speaker given its own reference signal.

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

Table 2.2: Noisy reverberant data specification.

Room dim. [m]	L_x	$U[4, 8]$
	L_y	$U[4, 8]$
	L_z	$U[2.5, 3]$
Reverb. time [sec]	T_{60}	$U[0.16, 0.2]$
Mic. Pos. [m]	x	$\frac{L_x}{2} + U[-0.5, 0.5]$
	y	$\frac{L_y}{2} + U[-0.5, 0.5]$
	z	1.5
Sources Pos. [°]	θ	U[0,180]
Sources Distance [m]		$1 + U[-0.5, 0.5]$

Noisy and reverberant conditions: Table 2.3 depicts the results of the proposed algorithm in comparison with the BSS algorithm trained on noisy and reverberant conditions, and to the oracle log-spectrogram combined with the noisy phase. First, it is clear that the noisy phase with the oracle spectrogram deteriorates the extraction capabilities. It is worth noting that this is the maximum score that can be obtained by the masking-based approach. For this reason, the performance of the VoiceFilter is not reported in this section. Second, our method outperforms the BSS method.

Table 2.3: SI-SDRi for noisy-reverberant data.

Model	BSS	Oracle	Proposed
Value	5.4	3.7	**5.7**

Real recording To further examine the capabilities of the proposed method, we recorded 2 speakers in a $3 \times 3 \times 2.5$, relatively quiet, enclosure. Both speakers are standing close to the microphone while uttering English sentences. Additionally, each participant was separately recorded to be used as the reference signal. Figure 2.3 depicts the sonograms of the experiment. The upper figure depicts the mixture recording. The center figure depicts the output of the model with the first reference. It is clear that the model accurately extracts the first speaker (denoted $\hat{s}_1$). To better understand the role of the reference signal embedding, the reference of the second speaker was recorded in Hebrew, which has a different phoneme structure than in English. The lower figure demonstrates the extraction capabilities of the second speaker (denoted $\hat{s}_2$). It is easy to verify that the algorithm is still capable of extracting this speech signal, despite the use of a reference signal in a different language. This may imply that the embedding focuses on the speaker's characteristics rather than the content of the reference utterance. The results are available for listening on our website.[1]

2.5 Conclusions

A novel combined time and time-frequency model was presented. This architecture enables the exploitation of the TF patterns of the speech while utilizing the time-domain SI-SDR loss. We also show that the RI features are beneficial for clean and for noisy and reverberant conditions and achieve better results than the LS features, which use the noisy phase for the reconstruction of the wave signal. Experiments show that our model outperforms BSS algorithms as well as common speaker extraction models.

[1] https://sharongannot.group/audio/

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

3 Simultaneous Speakers Detector and Localization of Multi-Sources for Separation and Noise Reduction

A blind and online speaker separation problem using a microphone array is addressed. When using the LCMV beamformer, the relative transfer function (RTF) of the various speakers is required. A library of RTFs associated with a grid of direction of arrivals (DOAs) is therefore blindly collected using a multi-task learning approach, to jointly identify single-speaker frames and classify the DOAs. A control mechanism, implemented as a convolutional neural network (CNN), uses these frames to estimate the RTFs by applying subspace estimation methods. In the experimental study, we demonstrate the performance benefits of the multi-task approach. The proposed scheme was evaluated using both simulated and real-life recordings in both static and dynamic scenarios. Interestingly, training the control mechanism with simulated data while testing it with real-life data, still provides good results. The proposed scheme was compared with Independent Low-Rank Matrix Analysis (ILRMA) algorithm (in static scenarios) and demonstrated perpetual improvements.

3.1 Introduction

In recent years, the usage of a multi-microphone array for speech processing becomes more widespread compared to the usage of a single microphone due to quality and stability considerations. Speech signals are often deteriorated by ambient noise, reverberation, and competing speakers. Therefore, separating and enhancing the desired speech from a mixture is a major challenge in the field of speech processing. Various methods for separating speakers and improving speech can be found in the following articles. A comprehensive survey of state-of-the-art, multi-channel audio separation methods can be found in [14, 32, 40].

Many speaker-separation methods use beamforming, such as minimum variance distortionless response (MVDR) [38] and LCMV [12, 39], to extract the desired speech. The RTF-based MVDR [13] and LCMV-beamformers (BFs) [17, 33] show great success in separating speakers, but require prior knowledge of the speakers RTFs. A multi-speaker LCMV-BF was proposed in [36] to simultaneously extract all individual speaker signals. In [33], the authors presented a method to estimate the RTFs, based on the generalized eigenvalue decomposition (GEVD) of the power spectral density (PSD) matrices of the received signals and the background noise.

Time frames dominated by a single speaker can facilitate the estimation of the RTF for each speaker of interest. However, the classification task of identifying these frames and associating them with the various active speakers is a cumbersome task. In [8], the authors based their technique on the assumption that the sources do not become simultaneously active. That is, they address the challenge of estimating the RTF of a single speech source while assuming that the RTFs of all other active sources in the environment were previously estimated in an earlier stage. Still, the activity of the speakers should be estimated in order to initiate a new RTF estimation procedure.

In [33, 36], the RTFs were estimated using time intervals comprising each of the desired speakers separately, assuming a static scenario. Practically, these time intervals need to be detected from data and cannot be assumed to be known. In [2], time frames dominated by each of the speakers were identified by estimating the DOA for each frame using clustering of a time series of steered response power (SRP) estimates. In [37], the activity of the speakers was estimated by introducing a latent variable that takes $N+1$ possible discrete states for a mixture of N speech signals plus additive noise. The activity is estimated using spatial cues from the observed signals modeled with a Gaussian-mixture-like model. In [25, 26], these frames were identified by exploiting convex geometry tools on the recovered simplex of the speakers' probabilities or the correlation function between frames [27].

Recently, deep neural network (DNN) methods have been widely used to separate speakers using microphone arrays. In [4], a neural-network-based, concurrent speaker detector was presented to detect single-speaker frames using only a single-microphone data, while spatial information of the overall microphone array was not taken into account.

In [5], a multi-microphone extension of a DNN-based classifier (dubbed multi-channel concurrent speakers detector (MCCD)) is presented, to achieve a more successful estimation of the speakers' activity (single speaker, multiple

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

speakers, or speakers absence). A library of RTFs was collected for constructing an LCMV-BF. The RTFs library was collected using single speaker frames. Since these frames can be associated with speakers from many possible locations, a new RTF was added to the library only when the cosine similarity between a new RTF and the RTFs at the library is relatively low. This stage can be unreliable since the typical cosine similarity between RTFs is unknown in advance.

In our work, we design NN model that, besides the concurrent speaker detector (CSD), also estimates the DOA of the speaker in each single speaker frame, thus constructing a library of RTFs associated with each DOA.

The task of DOA estimation using NN models was addressed in the literature [1, 3, 6, 11, 18, 48]. In [18], the speech sparsity in the STFT domain was utilized to track the DOAs of multiple speakers, using a CNN applied to the instantaneous RTF estimates. In our work, a similar DOA estimation procedure is added to the CSD estimation in a single NN model. Additionally, in [18], a TF mask (based on the DOA estimation per frequency) was obtained to perform speaker separation. While speech sparsity is not always guaranteed, speaker separation capabilities may not be always satisfactory. We stress, that in our work, the DOA estimates are only utilized for clustering of the single speaker frames and RTFs library construction, while the separation is carried out by the RTF-based LCMV-BF.

We now summarize the main stages of our contribution. A dual-task CNN-based model is presented. The activity of the speakers (single speaker, multiple speakers, or speakers absence) and the DOAs (mainly for single speaker frames) are simultaneously estimated. In the experimental study, we show that working with such a multi-tasking model actually improves the performance of each individual task. The inputs to the CNN consist of the log-spectrum of the current frame along with the current RTF estimate. Using these inputs, spectral and spatial information are used for both the CSD and the DOA classification tasks. The RTF is estimated using the GEVD method using some past and future frames. The log-spectrum and RTF are fed to the CNN model and the outputs are the CSD and DOA. At the training stage, the overall loss is the weighted sum of the losses of the DOA and CSD, while these marginal losses are calculated using categorical cross-entropy. A control system manages the speaker separation using the CNN outputs as further explained. Focusing on single-speaker frames (relying on the CSD output of the CNN), a library of DOA-based RTFs (namely an RTF for each possible DOA) is aggregated and updated. Then, in multi-speaker frames, these RTFs are used to separate the speaker using the LCMV beamformer. Since the identity of the speakers is unknown in multi-speaker frames, the latest DOAs from the latest single-speaker frames are assumed to be active and the relevant RTFs are used. The proposed CNN model was examined using simulated and real microphone recordings with static and dynamic speakers. The experimental study is divided into three parts: 1) CSD results, 2) DOA estimation results and 3) speaker separation performance using our control system and the LCMV beamformer. It is shown that CSD and DOA estimates are improved while concurrently estimating them. Finally, our separation performance is compared with ILRMA-algorithm [23] and shows perpetual improvement by measurements and by listening.

3.2 Problem Formulation and Basic Solution

In this work, the case of concurrent static or dynamic speakers acquired by a microphone array in a reverberant and noisy environment is considered. The signal received at the m-th microphone is represented in the STFT domain by

$$y_m(n,k) = \sum_{j=1}^{J(n)} g_{m,j}(n,k) s_j(n,k) + v_m(n,k) \tag{3.1}$$

where n, k denote the frame-index and frequency-index, respectively, $y_m(n,k)$ denotes the m-th microphone signal, $g_{m,j}(n,k)$ denotes the RTF relating microphone m and the reference microphone associated with speaker j, and $v_m(n,k)$ denotes the ambient noise. The individual speaker signals as received by the reference microphone are denoted by $s_j(n,k)$. The variable $J(n)$ is the number of active speakers at frame n which is not known in advance. The signal model in (3.1) can be recast in a vector form,

$$\mathbf{y}(n,k) = \sum_{j=1}^{J(n)} \mathbf{g}_j(n,k) s_j(n,k) + \mathbf{v}(n,k) = \mathbf{G}(n,k)\mathbf{s}(n,k) + \mathbf{v}(n,k) \tag{3.2}$$

where

$$\mathbf{y}(n,k) = (y_1(n,k), \ldots, y_M(n,k))^\top, \tag{3.3a}$$

$$\mathbf{g}_j(n,k) = (g_{1,j}(n,k), \ldots, g_{1,M}(n,k))^\top, \tag{3.3b}$$

$$\mathbf{G}(n,k) = (\mathbf{g}_1(n,k), \ldots, \mathbf{g}_M(n,k))^\top, \tag{3.3c}$$

$$\mathbf{v}(n,k) = (v_1(n,k), \ldots, v_M(n,k))^\top \tag{3.3d}$$

$$\mathbf{s}(n,k) = \left(s_1(n,k), \ldots, s_{J(n)}(n,k)\right)^\top \tag{3.3e}$$

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

Hereafter, the indexes n and k are omitted for brevity. The purpose of this work is to extract the individual speaker signals s_j, namely to propose a speaker separation and noise reduction method. As elaborated in the introduction, the linearly constrained minimum variance beamformer (LCMV-BF) [36] is widely used for the exploitation of spatial diversity between the microphones and extraction of individual speakers from a mixture. Hence, our main tool for speakers extraction is the LCMV-BF,

$$\widehat{\mathbf{s}} = \mathbf{W}_{\mathrm{LCMV}}^{\top}\mathbf{y} \tag{3.4}$$

where

$$\mathbf{W}_{\mathrm{LCMV}} = \Phi_{\mathbf{v}}^{-1}\mathbf{G}\left(\mathbf{G}^{H}\Phi_{\mathbf{v}}^{-1}\mathbf{G}\right)^{-1} \tag{3.5}$$

and the matrix $\Phi_{\mathbf{v}}$ denotes the noise spatial PSD matrix. The noise PSD matrix and the RTFs associated with the speaker are usually not known in advance and their blind estimation is therefore the main goal of this work. The noise spatial PSD matrix can be estimated using speech absence frames, while the RTFs can be estimated using single-speaker frames, namely frames in which only a single speaker is active.

Classifying the frames of the received signals according to their activity and associating the single-speaker frames to specific speakers will facilitate the construction of the LCMV-BF (3.5). A CNN-based technique for achieving these classifications is presented in the next section.

3.3 CNN-based technique For dual CSD and DOA estimation

In this section, we present a dual-task CNN model that determines the activity of the speakers per frame, namely the CNN model classifies each frame to either 1) speech absence, 2) single-speaker activity, or 3) multi-speaker activity. Simultaneously, the CNN model classifies each single-speaker frame to a DOA candidate, chosen from a predefined set of possible DOAs.

Next, The noise spatial PSD matrix is estimated using speech absence frames (class #1). The bank of RTFs (related to each DOA) is estimated using single-speaker frames (class #2). Namely, for single-speaker frames classified as DOA candidates, an RTF that is also related to the same DOA is updated. Finally, LCMV-BF will be employed on multi-speaker frames (class #3) using the estimated noise PSD and relevant RTFs.

3.3.1 Multi Task classification CNN

As elaborated above, the proposed acCNN model has two simultaneous outputs (two outputs are supplied in each frame). The first output is the CSD for each frame n:

$$\mathrm{CSD}(n) = \begin{cases} \text{Class } \#0 & \text{Noise only } (J(n)=0) \\ \text{Class } \#1 & \text{Single-speaker activity } (J(n)=1) \\ \text{Class } \#2 & \text{Multi-speaker activity } (J(n)>1) \end{cases}. \tag{3.6}$$

The second output is the DOA which estimates the DOA in single speaker frames (namely when $\mathrm{CSD}(n)=1$). The DOA output is neglected in other cases. The possible DOA range ($0^\circ \ldots 180^\circ$ in linear array) is divided into N sub-ranges with equal depth $\rho = \frac{180^\circ}{N}$. The CNN model classifies each single-speaker frame to each sub-range

$$\mathrm{DOA}(n) = \begin{cases} \text{Class } \#1 & \mathrm{DOA} \in (0 \ldots \rho - 1) \\ \text{Class } \#2 & \mathrm{DOA} \in (\rho \ldots 2\rho - 1) \\ \vdots & \vdots \\ \text{Class } \#\mathrm{N} & \mathrm{DOA} \in ((N-1)\rho \ldots N \cdot \rho) \end{cases}. \tag{3.7}$$

The proposed two-task model has essential advantages for the CSD task. Experimentally, it is shown that a CNN model trained to do only the CSD task performs less well relative to the two-task CNN model. It seems that the additional information of the DOA labels in the training stage actually helps to increase the performance of the speakers' activity classification.

3.3.2 Training stage

In this work, the following feature vector is used for the training stage. For each frame, the log-spectrum of the observed frame $\log|y_1(n,k)|$ and an estimated *local*-RTF $\widehat{\mathbf{g}}(n)$ are fed to the CNN model. It appeared experimentally, that adding the *local*-RTFs to the models' input actually improved the DOA classification. Note that the *local*-RTF is

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

only for additional input (to the model). For the actual LCMV-BF implementation, the RTF is re-estimated using only single-speaker frames.

The log-spectrum values are normalized across the frequency index, obtaining zero mean and unity variance,

$$\mathbf{a}(n) = \text{Normalize}\left(\left[\log|y_1(n,1)|\dots\log|y_1(n,K)|\right]^\top\right). \tag{3.8}$$

when for each matrix X, $\text{Normalize}(X)$ returns X with mean 0 and standard deviation 1. The *local*-RTFs are separated to their real and imaginary components and are normalized across the frequency and microphone indexes.

$$\text{B}(n) = \text{Normalize}\left(\begin{bmatrix}[l]\text{Re}\left([\widehat{\mathbf{g}}(n,1)\dots\widehat{\mathbf{g}}(n,K)]\right)\\ \text{Im}\left([\widehat{\mathbf{g}}(n,1)\dots\widehat{\mathbf{g}}(n,K)]\right)\end{bmatrix}\right) \tag{3.9}$$

Finally, the feature matrix $\begin{pmatrix}\mathbf{a}(n) & \text{B}^\top(n)\end{pmatrix}$ is fed to the model for each frame n.

The *local*-RTFs are estimated using the GEVD-based method [33]. First, a whitened version of the PSD matrix of the observed microphones is calculated

$$\widehat{\mathbf{\Phi}}_z(n) = \sum_{m=n-m_1}^{n+m_2} w_m \mathbf{z}(m)\mathbf{z}(m)^H \tag{3.10}$$

where w_m is a weighting factor (that usually emphasizes the current frame above future and past frames). The vector $\mathbf{z}(n)$ is a whitened version of the observed microphones vector $\mathbf{z}(n) = \Phi_{\mathbf{v}}^{-H/2}\mathbf{x}(n)$, where $\Phi_{\mathbf{v}}^{1/2}$ is the square root of the noise PSD matrix, and $\Phi_{\mathbf{v}} = \Phi_{\mathbf{v}}^{H/2}\Phi_{\mathbf{v}}^{1/2}$ (obtained using e.g., Cholesky decomposition). This denotes the principal eigenvector of $\widehat{\mathbf{\Phi}}_z(n)$ as $\widehat{\psi}(n)$. The estimator for the *local* RTF is given by:

$$\widehat{\mathbf{g}}(n) = \frac{\Phi_{\mathbf{v}}^{H/2}\widehat{\psi}(n)}{\mathbf{e}_1^\top\Phi_{\mathbf{v}}^{H/2}\widehat{\psi}(n)} \tag{3.11}$$

Note that the first element of $\widehat{\mathbf{g}}(n)$ can be omitted from the input matrix since it always equals 1.

3.3.3 Database diversity

To represent various real-life scenarios, the network was trained with a wide variety of simulated recordings. The following characteristics were manifested in the simulated recordings with corresponding ranges:

1. The geometric structure includes the room size, the location of the speakers, the location and orientation of the microphone array, and the location of the directional noise. However, the number of microphones and the structure of the microphone array were fixed. Additionally, it was assumed that the speakers are at least half a meter away from each other, and the distance of the speakers from the walls is at least half a meter.
2. The reverberation time of the room ranged by $T_{60} \in \{0.2\dots0.6\}$sec.
3. The SNR ranged by $\text{SNR} \in \{10\dots20\}$dB.
4. The order of the activity of the speakers: the appearance order of the activity of the speakers (noise only, single speaker, multi-speaker) within the signals was randomly simulated to reflect real-life scenarios, where the activity of the speakers is unexpected[1].
5. Speaker activity and DOAs: for each class, the database was equally balanced, namely, the same amount of recordings was simulated for each class. Overall, the entire database was divided into three data subsets of the speakers' activity, where the subset of the single-speaker activity was divided into N data subsets for the N DOA candidates. It should be emphasized that the training data contained only static speakers while the algorithm was tested also with dynamic speakers.

[1] Many papers constructed a database where the signals always have a fixed order: noisy period, single-speaker period and multi-speaker period. In our database, we simulate the real-life variety of activities.

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

3.3.4 CNN structure

The proposed CNN model consists of three convolutional layers followed by three fully-connected (FC) layers. Finally, two separated routes for each classification are constructed 1) single FC layer with N outputs for the DOA classification, and 2) two FC layers with 3 outputs for the speaker activity classification. Each output of each layer was activated by Rectified Linear Unit (ReLU), apart from the last outputs (N for the DOA and three for the speaker activity), which were activated by the softmax function. Using max pooling operations, the convolution layers were calculates the maximum value for patches of a feature map (2X2 features), and uses it to create a downsampled (pooled) feature map. Dropout, batch normalization, and weight limitation operations were added to prevent over-fitting.

The categorical cross-entropy (CCE) loss function was used for training the model and the Adaptive Moment Estimation (ADAM) optimizer was used. To make the DOA output more efficient for the learning process, three important updates of the loss function were made.

1. Estimating the RTF in the case of two (or more) mutual speakers may be disastrous to the LCMV beamforming, which actually turns a beam to the wrong RTF. Therefore, labeling a multi-speaker activity as single-speaker activity is severely undesirable. An option to avoid it is to increase the loss function for this case and to denote pCSD as the predicted CSD and aCSD as the actual CSD. The sub-loss function of the CSD is obtained by:

$$\mathrm{Loss}_{\mathrm{CSD}} = \begin{cases} \alpha \cdot \mathrm{CCE}_{\mathrm{CSD}} & \mathrm{pCSD} = 1 \ \& \ \mathrm{aCSD} = 2 \\ \mathrm{CCE}_{\mathrm{CSD}} & \text{otherwise} \end{cases}, \tag{3.12}$$

   where $\mathrm{CCE}_{\mathrm{CSD}}$ is the CCE between the actual CSD and the model distribution and ($\alpha > 1$) is a magnification parameter.

2. Low angular errors in DOA may be tolerable versus high angular errors in DOA. Therefore, the loss for high angular errors may be increased w.r.t. the loss for low angular errors. The ordinal categorical cross-entropy is therefore adopted to enable the loss to be consistent with the Euclidean distance between the actual and the predicted DOA.

   Accordingly, the sub-loss function of the DOA is obtained by:

$$\mathrm{Loss}_{\mathrm{DOA}} = \frac{|\mathrm{pDOA} - \mathrm{aDOA}|}{N} \mathrm{CCE}_{\mathrm{DOA}} \tag{3.13}$$

3. The DOA estimates in speakers' absence or multi-speaker frames are useless for the purposes of this work. Therefore, the loss function for the DOA in these cases is set to zero ($\mathrm{Loss}_{\mathrm{DOA}} = 0$ when aCSD=0 or aCSD=2). However, since these cases cover approximately two-thirds of the training data (as elaborated in the experimental section), the model may suffer from unbalance between the CSD training and the DOA training. To compensate for this, the sub-loss function of the DOA was enhanced w.r.t. the sub-loss function of the CSD, such that the total loss is given by:

$$\mathrm{Loss} = \beta \mathrm{Loss}_{\mathrm{DOA}} + \mathrm{Loss}_{\mathrm{CSD}} \tag{3.14}$$

   where ($\beta > 1$) is a magnification parameter.

3.4 RTFs and noise PSD estimation

In the following sections, a procedure for estimating the ambient noise PSD and the speakers RTFs using the outputs of the CNN model is described. Next, a procedure for determining the current speakers' activity is described.

3.4.1 Noise PSD estimation

Only in those frames that the CSD implies on speech absence is the noise PSD matrix updated by

$$\Phi_{\mathbf{v}}(n,k) = \gamma_n \Phi_{\mathbf{v}}(n-1,k) + (1-\gamma_n)\mathbf{y}(n,k)\mathbf{y}^{\mathbf{H}}(n,k), \tag{3.15}$$

when γ is the learning rate factor that will be adjusted according to the rate of noise diversity.

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

3.4.2 RTFs estimation

The LCMV-BF in (3.5) is designed to separate and enhance up to M speakers. Thus, the RTFs associated with the dominant M (or less) present speakers are required. Two procedures are hereafter described: RTFs estimation and LCMV design. The RTFs of the speakers may be estimated using the GEVD-based method (3.11) applied to frames with a single speaker. However, when the speakers are in motion, the RTFs are changed across time and therefore need to be continually re-estimated. Another possibility is estimating individual RTF for each optional DOA by storing a library of DOA-based PSD matrices. To do so, we use the CNN model outputs (CSDs and DOAs) to locate single-speaker frames for each possible DOA. Then, in such a frame when $\mathbf{DOA} = j$, the relevant PSD matrix is updated by

$$\Phi_{\mathbf{j}}(n,k) = \frac{1}{\delta_j}\Phi_{\mathbf{j}}(n-1,k) + \left(1 - \frac{1}{\delta_j}\right)\mathbf{y}(n,k)\mathbf{y}^{\mathbf{H}}(n,k), \tag{3.16}$$

where $j = 1 : N$ and δ_j is the learning rate factor. In frames when $\mathbf{DOA} = j$, the PSD matrix of the j-th DOA is reduced by $\Phi_{\mathbf{j}}(n,k) = \frac{1}{\delta_j}\Phi_{\mathbf{j}}(n-1,k)$ in order to diminish its effect the next time $\Phi_{\mathbf{j}}$ will be updated (when the speaker will again appear in DOA j).

3.4.3 LCMV designing

Additionally, the LCMV can satisfy M channels of speakers at the same time; the fewer channels there are, the higher the quality of the separation will be for each output channel. Note that when a single speaker is active, the LCMV-BF degenerates to the MVDR-BF, and in noise-only frames, no activation of any BF is needed. It is therefore important to apply the LCMV with the RTFs of the current active speakers in order to achieve better separation ability and low computational burden.

In each frame, the LCMV consists of the RTFs associated with the DOAs in group Θ where $|\Theta| < M$. For each $j \in \Theta$, the relevant RTF is estimated using Φ_j by the GEVD technique. The procedure for determining Θ for each frame is based on the following principles:

1. In case of a single-speaker frame (CSD=1) and $\mathbf{DOA} = j$, the DOA j is added to Θ. In case there is already DOA which is angularly closed to the new DOA, the older DOA is removed. If Θ already consists of M angularly far DOAs, the most veteran DOA is removed from Θ.
2. In case of multi-speaker frames (CSD=2), Θ is untouched.
3. A DOA is removed from Θ after Q frames attributed with only noise or another single speaker DOA (further called 'expiry date').

The overall processing flow, from the noisy inputs up to the final outputs (the enhanced and separated dominant speakers) can be seen in Algorithm 1 and is illustrated in Figure 3.1. For a better understanding of the processing flow, an example of real-time processing of input signals is given in Figure 3.2. In the figure, one can track across time after the oracle speakers activity and DOA, the CNN outputs (CSD and DOA), the content of Θ, and the number of beams of the LCMV..

Algorithm 1: Processing Flow

Input: Noisy signal in the STFT domain $\mathbf{z}(n,k)$
while *observing new frame* n **do**
  CNN stage:
  Inputting feature matrix $\left[\mathrm{a}(n) \quad \mathrm{B}^{\top}(n)\right]$ to the CNN model and obtaining $\mathrm{CSD}(n)$ and $\mathrm{DOA}(n)$.
  Update stage:
  if $\mathrm{CSD}(n) = 0$ **then**
    Updating the noise PSD $\Phi_{\mathbf{v}}$ using (3.15)
  else if $\mathrm{CSD}(n) = 1$ *& DOA=j* **then**
    Updating the DOA based PSD Φ_j using (3.16)
  Determining the active speakers:
  Update Θ using the principles at Sec. 3.4.3
  RTF estimation stage:
  Estimate the RTFs using the PSDs matrices attributed to the DOAs in Θ using the GEVD technique (3.11).
  BF stage:
  Apply LCMV/MVDR using the RTFs.

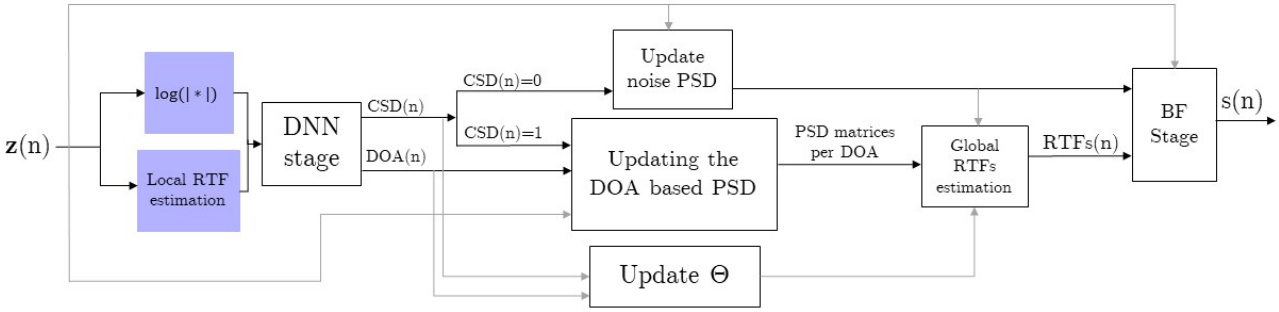


Figure 3.1: Processing Flow

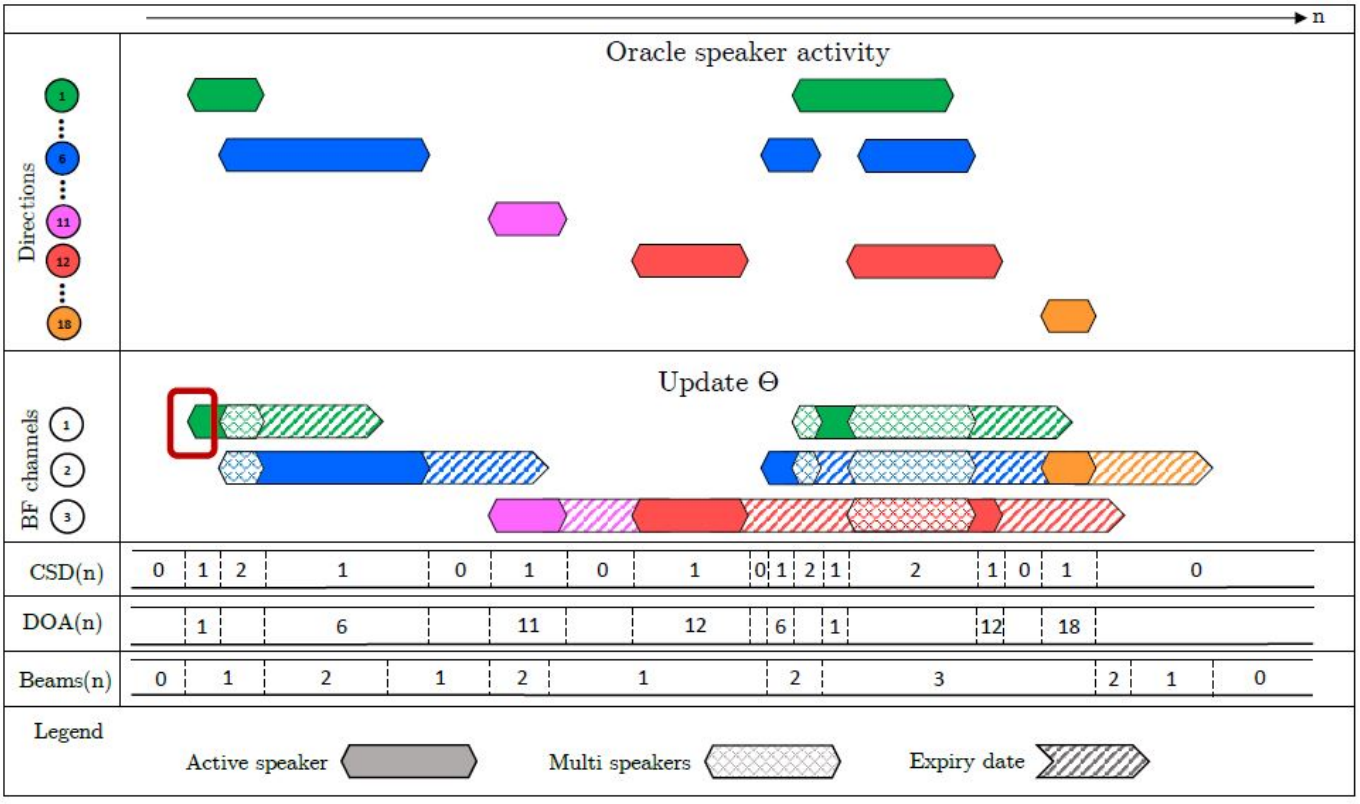


Figure 3.2: Update Θ using the principles at Sec. 4.3. For illustration, a resolution of 10 degrees was chosen ($\rho = 10°$).

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

3.5 Experiment

In this section, the performance of our technique is compared with another baseline technique for the three basic parts: CSD estimation, DOA estimation, and speaker separation and enhancement. First, the database used for training and testing is described. Then the aforementioned parts are evaluated, using the testing database in separate sections.

3.5.1 Database description

The overall database consists of 500 simulated signals (each 40sec length), generated using open-source software [2]. For the training database, signals from the TIMIT database were convolved by synthetic acoustic transfer function (ATF)s. In our experiments, the case of a maximum of two concurrent speakers was examined. Each signal is a summation of two single-speaker signals, such that frames with 0,1,2 concurrent, activated speakers are obtained. Since the speakers are in various gains, the SIR between the speakers was $\in 0-5$dB. The evaluated array is $M=4$ microphones, equally placed in a half-circle structure with a 10 cm diameter. The DOA resolution of our model is 10^o ($N=18$ labels), where each signal contains speakers from arbitrary DOAs ($0..180^o$). To keep the balanced database from a DOA point of view, each signal contains speakers from all possible DOA labels with arbitrary order (each speaker was static in its DOA. However, for the test database dynamic, speakers were used.

The various arbitrary parameters are summarized in Table /ref. Three background noises were added: 1) directional

Parameter	Range	Remarks
T_{60}	0.3..0.55ms	
Array orientation	0..360	
Array Position	all over the room	at least 0.5m from the walls
Speakers position	all over the room	at least 0.5m between speakers
Directional noise position	all over the room	at least 2m from array
Diffuse noise SNR	10-20dB	
Speaker to mic distance	1-1.5m	

Table 3.1: Table to test captions and labels

noise with constant SNR of 20dB and arbitrary positions, 2) diffuse noise with arbitrary SNR in the range of 10-20dB SNR, and 3) sensors noise with constant SNR of 30dB. As for the test database, the test recordings database of TIMIT was used. Two moving speakers were generated using open-source software.[3] The first speaker moves between DOA $0°$ to $140°$ and the second speaker moves between $160°$ to $180°$.

3.5.2 CSD performance

In this section, the performance of the proposed CSD is compared with the multi-channel current speakers activity detector (MCCSD). There are two main differences between the proposed model and MCCSD: 1) the input to MCCSD contains the log spectrum $\log|\mathbf{y}(n,k)|$, forming past and future frames (w.r.t. the current frame) from all microphones, while in our model, only the current log spectrum from a single microphone is inputted with an estimation of the RTF (using the same past and future frames); overall, the input vector of our model is smaller than in MCCSD. 2) In the proposed model, there is also a clustering of the DOA of the speakers in single-speaker frames (Note that the proposed model is only 10% more complex than the MCCSD).

Tables 3.5.2, III and IV depict the confusion matrices of MCCSD, CSD without the DOA clustering and CSD with the DOA clustering, respectively.

Table 3.2: Confusion matrix of MCCSD [percentage]

Estimated \ True	Class 0	Class 1	Class 2
Class 0	88.3	12.5	0.4
Class 1	9.9	75.4	15.8
Class 2	1.8	12.1	83.8

Comparing Tables II and III, it can be seen that the accuracy for Class 2 is better with the proposed model even without the DOA clustering.

[2] https://www.audiolabs-erlangen.de/fau/professor/habets/software
[3] https://www.audiolabs-erlangen.de/fau/professor/habets/software/signal-generator

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

Table 3.3: Confusion matrix of CSD without DOA classifier [percentage]

Estimated \ True	Class 0	Class 1	Class 2
Class 0	90.4	3.7	0.4
Class 1	8.6	78.3	9.4
Class 2	1.0	18.0	90.2

Table 3.4: Confusion matrix of CSD with DOA classifier [percentage]

Estimated \ True	Class 0	Class 1	Class 2
Class 0	91.1	1.9	0.0
Class 1	8.3	85.9	4.7
Class 2	0.6	12.2	95.3

Comparing Tables III and IV, it can be seen that the accuracy for Class 1 is better with the DOA clustering relative to the model without the DOA clustering. It may be deduced that the classification of the DOA within the model helps in identifying the activity of a single speaker. In our model, additional characteristics are given for a single speaker (its optional DOAs) whereas, for the multi-speaker case, the DOA is irrelevant. Note that the common weights of our model are updated by our special integrated (CSD and DOA) loss function. Therefore, an inter-channel effect (between the CSD and DOA channels) happens by activating the common back-propagation operation. Therefore, since single-speaker frames with dominant DOA have additional information strengthening the hypothesis of single-speaker, our model may have an advantage in identifying single-speaker frames versus multiple-speaker frames.

3.5.3 Performance of the DOA

In this section, the performance of the proposed DOA estimator is compared versus the SRP algorithm. As described, the DOA output of the DNN model is valid only while the CSD implies a single speaker frame. Signals in which a single speaker moves along an arc were examined for different reverberation and SNR levels. The baseline algorithm is the SRP-PHAT, denoted by $\hat{\theta}_S = \theta_{\hat{j}}$, where:

$$\hat{j}(n) = \text{argmax}_j \sum_{q_1=1}^{M} \sum_{q_2=q_1+1}^{M} \sum_{k} \frac{\hat{\Phi}_{\mathbf{y},q_1q_2}(n,k)}{\left|\hat{\Phi}_{\mathbf{y},q_1q_2}(n,k)\right|} \frac{G^*_{j,q_1}}{G_{j,q_2}}$$

the coherence matrix of the input data $\hat{\Phi}_{\mathbf{y}}$ is defined by

$$\hat{\Phi}_{\mathbf{y}}(n,k) = \sum_{m=n-m_1}^{n+m_2} w_m \mathbf{y}(m)\mathbf{y}(m)^H$$

and $\hat{\Phi}_{\mathbf{y},q_1q_2}$ is the the coherence between the q_1 and q_2 microphone signals.

The main purpose of the DOA estimation is to employ speaker separation. Therefore, the various errors in DOA estimates are divided into three classes (Note that the resolution of the DOA estimates is 10^o):

1. Successful prediction: the estimator succeeds in predicting the actual DOA label.
2. Low error prediction: the estimator fails to predict the DOA up to 2 labels ($\leq 20^o$). Such an error may cause damage when the same actual DOA is classified into two different DOAs and divides the total number of frames of single speakers into two different PSD matrices per the two DOAs estimates. In such a case, the PSD matrixes may lose relevant frames. However, there is not much of a risk for associating the frames of one speaker with another speaker, since different speakers usually do not stand close to each other.
3. High error prediction: the estimator fails to predict the DOA more than 2 labels ($> 20^o$). This error can cause much damage, since the PSD matrices may be updated by frames associated with distant DOAs w.r.t. to the true DOA; As a result the RTFs may have high estimation errors.

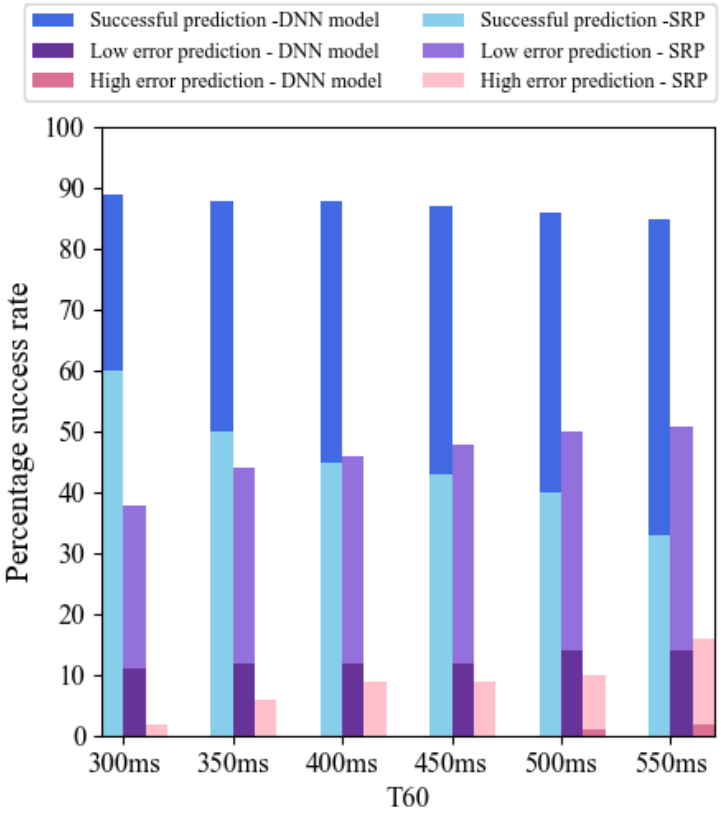


Figure 3.3: DOA performance for different reverberation time

Figure 3 depicts the DOA performance in different reverberation times for the three different error types. It can be verified that the performance of the proposed DOA estimator is approximately constant for all T_{60} levels and outperforms the performance of the SRP-PHAT.

The performed test for different SNR values in the range of 10dB to 20dB with a T_{60} of 300 msec and the subsequent results show that there is no change following the increase in noise for the two algorithms and that the results are similar to what is shown for 20dB.

3.5.4 Performance of the LCMV with the processing flow

In this section, the performance of the proposed full processing flow (namely employing online the CSD, DOA and LCMV BF) is analyzed versus the performance of ILRMA algorithm [23].

The performance is analyzed along three conditions: SNR, reverberation time, and the SIR between the desired speaker and the interfering speaker. Two scenarios are examined: static speakers and moving speakers (while the movements are only within single talk periods). The speakers' velocity is set to 0.3 meters per second. While ILRMA can be applied only for static speakers, the performance of ILRMA is presented only for this scenario. Two speakers were simulated when the first speaker starts speaking alone; the second speaker then speaks alone and then the two speakers mutually speak. Ten signals with a 40-second duration are examined. For the static case, the two speakers are placed at a random angle in each signal. As for the dynamic case, the first speaker starts at an angle of 0^o and goes towards another angle of 140^o, and returns toward an angle of 0^o. The second speaker alternately moves between angles 160^o and 180^o. Three measurements were calculated: Short Term Objective Intelligibility (STOI), Perceptual Evaluation of Speech Quality (PESQ), and output SIR. The measurements were taken only from the double-talk periods.

In Figure 3.4, the measurements of the proposed algorithm and ILRMA are depicted for a static scenario. It can be seen that the proposed algorithm outperforms the results of ILRMA. In Figure 3.5, the measurement of the proposed algorithm is depicted for a dynamic scenario. It can be seen that the proposed algorithm improves the results of the input signals. Moreover, the results are similar to the results of the static scenario, despite the dynamic behavior of the speakers. Figure 3.6 depicts example signals with SNR=10dB and SIR=0dB.

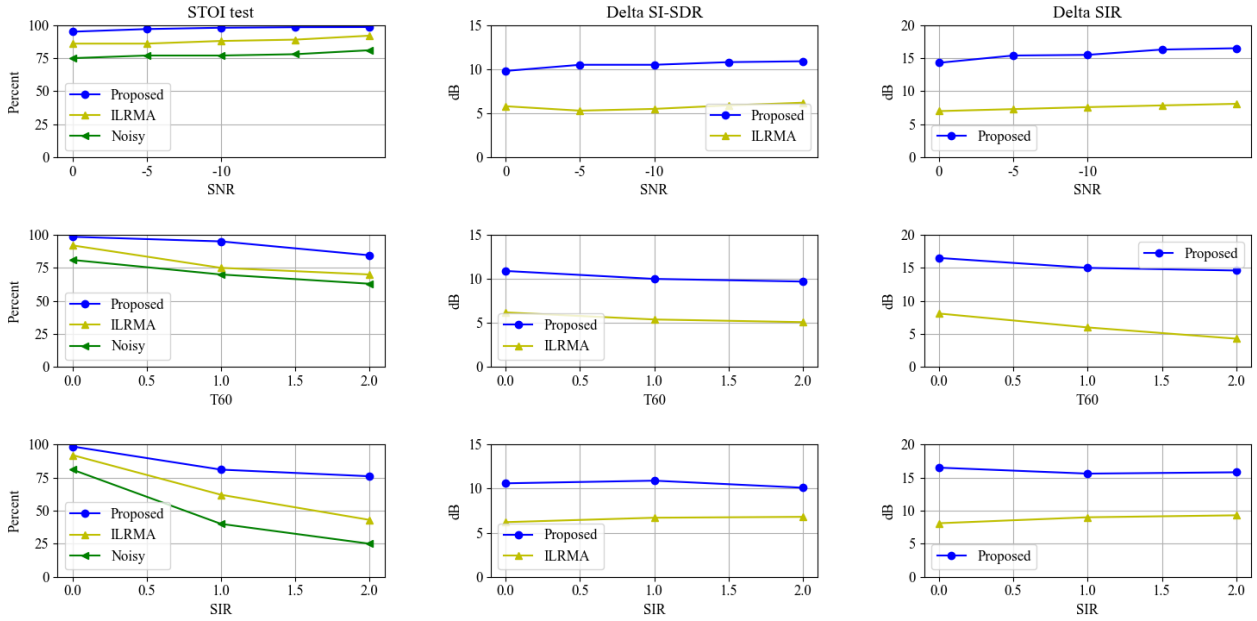


Figure 3.4: source separation results - static scenario

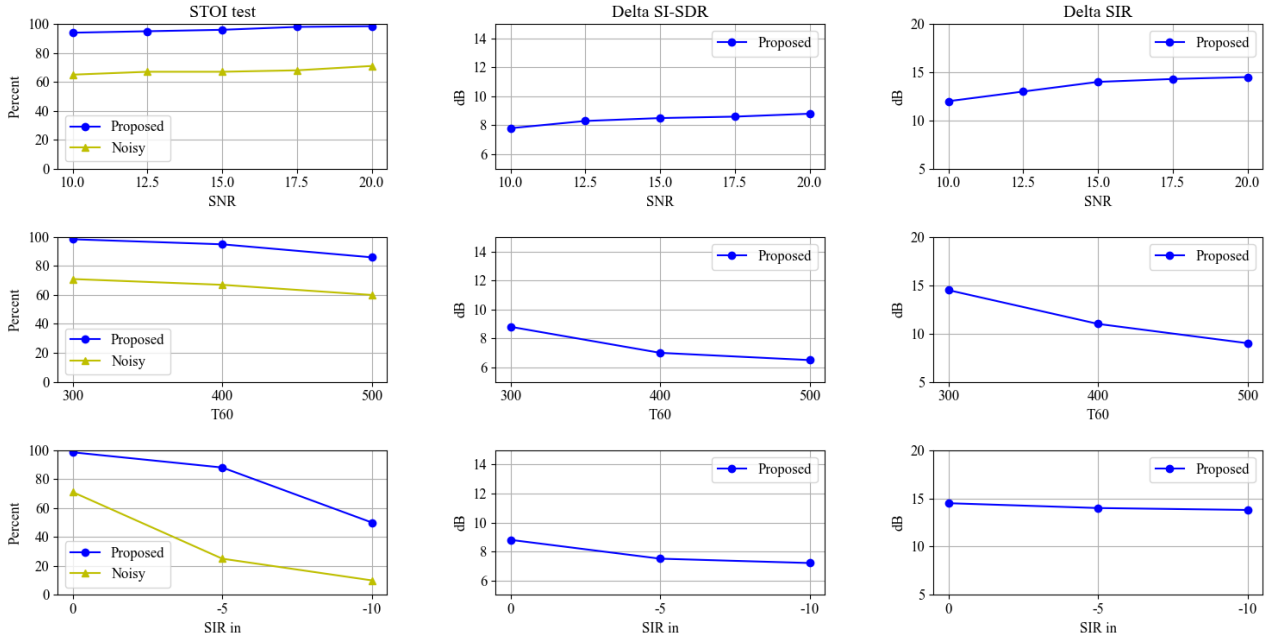


Figure 3.5: source separation results - dynamic scenario

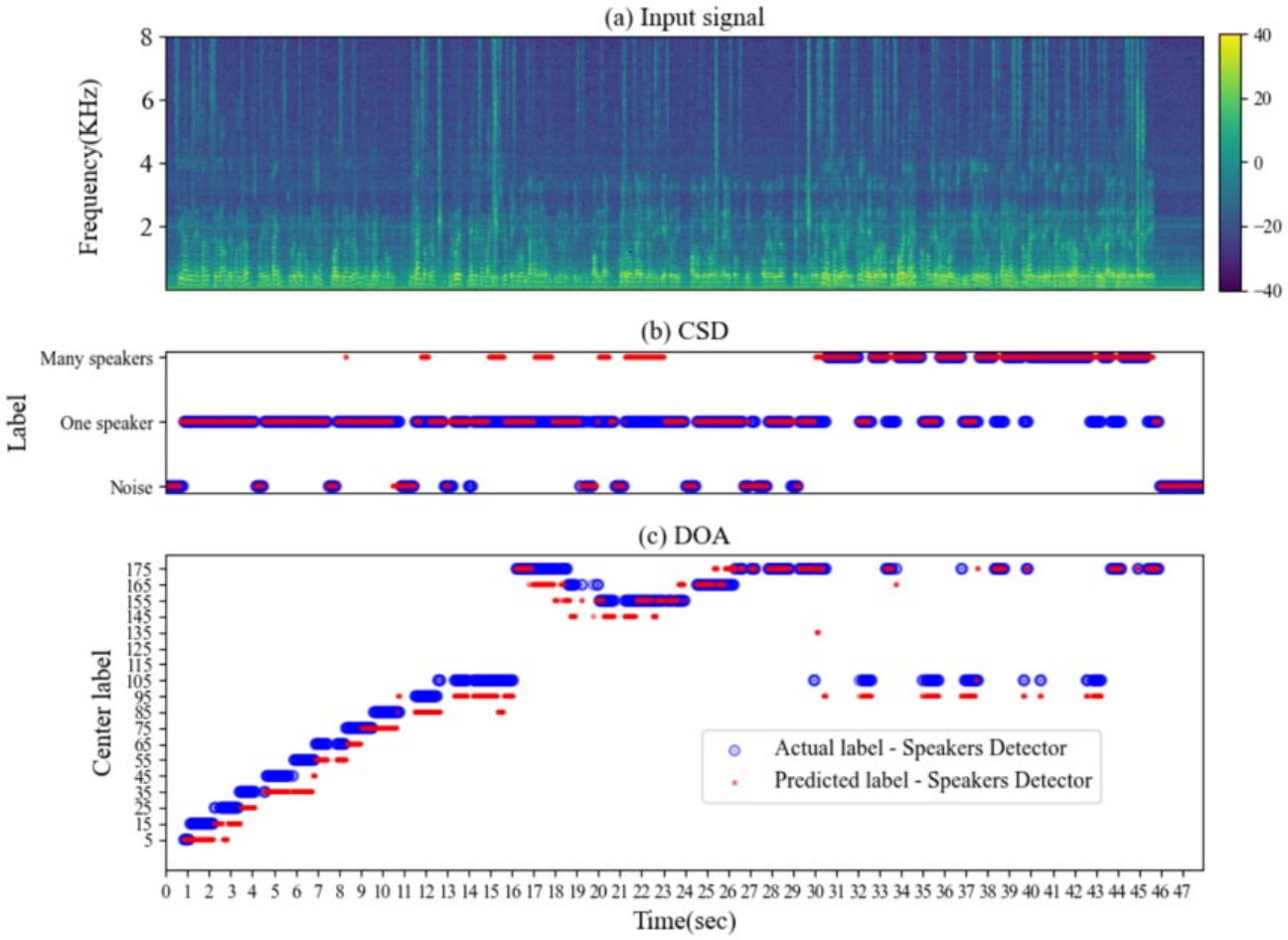


Figure 3.6: source separation results - one example

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

3.6 Conclusion

Blind and online speaker separation task using a microphone array was addressed. The LCMV beamformer was designed using the RTFs of the various speakers. A library of RTFs per DOA was blindly collected using single-speaker frames. A multi-tasking CNN, single speaker frames, and DOA were mutually identified. Using a control system, these frames were exploited to calculate the RTFs using the GEVD method. In the experimental study, it was demonstrated that working with such a multi-tasking CNN actually improves the performance for each individual task (CSD and DOA estimation). The proposed CNN model was examined using simulated and real microphone recordings with static and dynamic speakers. Comparing this to the ILRMA-algorithm shows perpetual improvement.

This project has received funding from the European Union's Horizon 2020
Research and Innovation Programme under Grant Agreement No. 871245.

4 Conclusions

This document reports the progress in the speaker separation task for a static robot. We presented two new algorithms, developed in the course of the SPRING project, one using a single microphone and the other multiple microphones. Both algorithms are implemented in Python.

The single-microphone extraction algorithm demonstrates good SIR and SDR results for low reverberation level, provided that an extract of the desired speaker is available. This may be obtained by identifying time frames in the same conversation (about 1 sec long) in which only a single speaker is active. We note that in the current audio processing architecture, two independent audio streams are simultaneously transcribed by the ASR system and transmitted to the dialogue system. To circumvent the speaker permutation phenomenon, typical to separation algorithms, we will apply a speaker identification module on the separated outputs and preserve the time-consistency of the transcribed speech signals.

The multi-microphone algorithm was extensively tested, including with moving human speakers moving in front of ARI. The algorithm is characterized by high separation capabilities and low speech distortion. Since the multi-microphone algorithm also extracts spatial information, time consistency may also be preserved using the DOA estimates.

The final decision on which algorithm will be eventually deployed on ARI, will be made at a later stage, based on the following considerations: 1) performance (in terms of ASR accuracy) in the target acoustic environment; 2) robustness to changing conditions (multi-microphone solutions usually provide better results but are prone to significant performance drop if the acoustic conditions are rapidly changing); 3) computational load and memory requirements; and 4) system perspective, e.g. how well the algorithms fits the entire audio pipeline, especially if several enhancement algorithms should be cascaded.

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

Bibliography

[1] Sharath Adavanne, Archontis Politis, Joonas Nikunen, and Tuomas Virtanen. Sound event localization and detection of overlapping sources using convolutional recurrent neural networks. *IEEE Journal of Selected Topics in Signal Processing*, 13(1):34–48, 2018.

[2] Shoko Araki, Masakiyo Fujimoto, Kentaro Ishizuka, Hiroshi Sawada, and Shoji Makino. Speaker indexing and speech enhancement in real meetings/conversations. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 93–96, 2008.

[3] Alexander Bohlender, Ann Spriet, Wouter Tirry, and Nilesh Madhu. Exploiting temporal context in cnn based multisource doa estimation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:1594–1608, 2021.

[4] Shlomo E Chazan, Jacob Goldberger, and Sharon Gannot. DNN-based concurrent speakers detector and its application to speaker extraction with LCMV beamforming. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6712–6716, 2018.

[5] Shlomo E. Chazan, Jacob Goldberger, and Sharon Gannot. LCMV beamformer with DNN-based multichannel concurrent speakers detector. In *The 26th European Signal Processing Conference (EUSIPCO)*, Rome, Italy, September 2018.

[6] Shlomo E Chazan, Hodaya Hammer, Gershon Hazan, Jacob Goldberger, and Sharon Gannot. Multi-microphone speaker separation based on deep doa estimation. In *2019 27th European Signal Processing Conference (EUSIPCO)*, pages 1–5. IEEE, 2019.

[7] Shlomo E Chazan, Lior Wolf, Eliya Nachmani, and Yossi Adi. Single channel voice separation for unknown number of speakers under reverberant and noisy settings. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3730–3734, 2021.

[8] Dani Cherkassky and Sharon Gannot. Successive relative transfer function identification using blind oblique projection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:474–486, 2019.

[9] Marc Delcroix, Tsubasa Ochiai, Katerina Žmolíková, Kateřina, Keisuke Kinoshita, Naohiro Tawara, Tomohiro Nakatani, and Shoko Araki. Improving speaker discrimination of target speech extraction with time-domain speakerbeam. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 691–695, 2020.

[10] Chengyun Deng, Shiqian Ma, Yi Zhang, Yongtao Sha, Hui Zhang, Hui Song, and Xiangang Li. Robust speaker extraction network based on iterative refined adaptation. *arXiv preprint arXiv:2011.02102*, 2020.

[11] David Diaz-Guerra, Antonio Miguel, and Jose R Beltran. Robust sound source tracking using srp-phat and 3d convolutional neural networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:300–311, 2020.

[12] Meng Hwa Er and Antonio Cantoni. Derivative constraints for broad-band element space antenna array processors. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 31(6):1378–1393, 1983.

[13] Sharon Gannot, David Burshtein, and Ehud Weinstein. Signal enhancement using beamforming and nonstationarity with applications to speech. *IEEE Transactions on Signal Processing*, 49(8):1614–1626, 2001.

[14] Sharon Gannot, Emmanuel Vincent, Shmulik Markovich-Golan, and Alexey Ozerov. A consolidated perspective on multimicrophone speech enhancement and source separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(4):692–730, 2017.

[15] Meng Ge, Chenglin Xu, Longbiao Wang, Eng Siong Chng, Jianwu Dang, and Haizhou Li. Spex+: A complete time domain speaker extraction network. *arXiv preprint arXiv:2005.04686*, 2020.

[16] Emanuël AP Habets. Room impulse response generator. Technical report, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2014.

[17] Emanuël AP Habets, Jacob Benesty, Sharon Gannot, Patrick A Naylor, and Israel Cohen. On the application of the LCMV beamformer to speech enhancement. In *2009 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 141–144. IEEE, 2009.

[18] Hodaya Hammer, Shlomo E. Chazan, Jacob Goldberger, and Sharon Gannot. Dynamically localizing multiple speakers based on the time-frequency domain. *EURASIP Journal on Audio, Speech and Music*, March 2021.

[19] Jiangyu Han, Wei Rao, Yanhua Long, and Jiaen Liang. Attention-based scaling adaptation for target speech extraction. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 658–662, 2021.

[20] Shulin He, Hao Li, and Xueliang Zhang. Speakerfilter: Deep learning-based target speaker extraction using anchor speech. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 376–380, 2020.

[21] Yanxin Hu, Yun Liu, Shubo Lv, Mengtao Xing, Shimin Zhang, Yihui Fu, Jian Wu, Bihong Zhang, and Lei Xie. DCCRN: deep complex convolution recurrent network for phase-aware speech enhancement. *arXiv preprint arXiv:2008.00264*, 2020.

[22] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[23] Daichi Kitamura. Algorithms for independent low-rank matrix analysis, 2018.

[24] Morten Kolbæk, Dong Yu, Zheng-Hua Tan, and Jesper Jensen. Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(10):1901–1913, 2017.

[25] Bracha Laufer-Goldshtein, Ronen Talmon, and Sharon Gannot. Source counting and separation based on simplex analysis. *IEEE Transactions on Signal Processing*, 66(24):6458–6473, 2018.

[26] Bracha Laufer-Goldshtein, Ronen Talmon, and Sharon Gannot. Global and local simplex representations for multichannel source separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:914–928, 2020.

[27] Bracha Laufer Goldshtein, Ronen Talmon, and Sharon Gannot. Audio source separation by activity probability detection with maximum correlation and simplex geometry. *EURASIP Journal on Audio, Speech and Music*, 2021, January 2021.

[28] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R Hershey. SDR–half-baked or well done? In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 626–630, 2019.

[29] Tingle Li, Qingjian Lin, Yuanyuan Bao, and Ming Li. Atss-Net: Target Speaker Separation via Attention-Based Neural Network. In *Proc. Interspeech 2020*, pages 1411–1415, 2020.

[30] Yi Luo, Zhuo Chen, and Takuya Yoshioka. Dual-path RNN: efficient long sequence modeling for time-domain single-channel speech separation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 46–50, 2020.

[31] Yi Luo and Nima Mesgarani. Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 27(8):1256–1266, 2019.

[32] Shoji Makino, editor. *Audio source separation*. Signals and Communication Technology. Springer International Publishing, Cham, Switzerland, 2018.

[33] Shmulik Markovich, Sharon Gannot, and Israel Cohen. Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals. *IEEE Transactions on Audio, Speech, Language Process.*, 17(6):1071–1086, 2009.

[34] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210, 2015.

[35] Douglas B Paul and Janet Baker. The design for the Wall Street Journal-based CSR corpus. In *Speech and Natural Language Workshop*, Harriman, New York, 1992.

[36] Ofer Schwartz, Sharon Gannot, and Emanuël AP Habets. Multispeaker LCMV beamformer and postfilter for source separation and noise reduction. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(5):940–951, 2017.

[37] Mehrez Souden, Shoko Araki, Keisuke Kinoshita, Tomohiro Nakatani, and Hiroshi Sawada. A multichannel MMSE-based framework for speech source separation and noise reduction. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(9):1913–1928, 2013.

[38] Harry L Van Trees. *Optimum array processing: Part IV of detection, estimation, and modulation theory*. John Wiley & Sons, New-York, USA, 2004.

[39] Barry D Van Veen and Kevin M Buckley. Beamforming: A versatile approach to spatial filtering. *IEEE Acoustics, Speech and Signal Proc. Mag.*, 5(2):4–24, 1988.

[40] Emmanuel Vincent, Tuomas Virtanen, and Sharon Gannot. *Audio source separation and speech enhancement*. John Wiley & Sons, New-Jersey, USA, 2018.

[41] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno. Generalized end-to-end loss for speaker verification. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4879–4883, 2018.

[42] Jun Wang, Jie Chen, Dan Su, Lianwu Chen, Meng Yu, Yanmin Qian, and Dong Yu. Deep extractor network for target speaker recovery from single channel speech mixtures. In *Proc. Interspeech 2018*, pages 307–311, 2018.

[43] Quan Wang, Hannah Muckenhirn, Kevin Wilson, Prashant Sridhar, Zelin Wu, John Hershey, Rif A Saurous, Ron J Weiss, Ye Jia, and Ignacio Lopez Moreno. Voicefilter: Targeted voice separation by speaker-conditioned spectrogram masking. *arXiv preprint arXiv:1810.04826*, 2018.

[44] Gordon Wichern, Joe Antognini, Michael Flynn, Licheng Richard Zhu, Emmett McQuinn, Dwight Crow, Ethan Manilow, and Jonathan Le Roux. Wham!: Extending speech separation to noisy environments. *arXiv preprint arXiv:1907.01160*, 2019.

[45] Xiong Xiao, Zhuo Chen, Takuya Yoshioka, Hakan Erdogan, Changliang Liu, Dimitrios Dimitriadis, Jasha Droppo, and Yifan Gong. Single-channel speech extraction using speaker inventory and attention network. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 86–90, 2019.

[46] Chenglin Xu, Wei Rao, Eng Siong Chng, and Haizhou Li. Time-domain speaker extraction network. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 327–334, 2019.

[47] Chenglin Xu, Wei Rao, Eng Siong Chng, and Haizhou Li. Spex: Multi-scale time domain speaker extraction network. *IEEE/ACM transactions on audio, speech, and language processing*, 28:1370–1384, 2020.

[48] Bing Yang, Hong Liu, and Xiaofei Li. Srp-dnn: Learning direct-path phase difference for multiple moving sound source localization. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 721–725. IEEE, 2022.

[49] Zining Zhang, Bingsheng He, and Zhenjie Zhang. X-TaSNet: Robust and accurate time-domain speaker extraction network. *arXiv preprint arXiv:2010.12766*, 2020.

[50] Kateřina Žmolíková, Marc Delcroix, Keisuke Kinoshita, Tsubasa Ochiai, Tomohiro Nakatani, Lukáš Burget, and Jan Černockỳ. Speakerbeam: Speaker aware neural network for target speaker extraction in speech mixtures. *IEEE Journal of Selected Topics in Signal Processing*, 13(4):800–814, 2019.