Deliverable D3.2: Audio-visual speaker tracking in relevant environments

Due Date: 01/06/2022

Main Author: Luis G Camara (INRIA) and Sharon Gannot (BIU)

Contributors: Chris Reinke (INRIA) and Xavier Alameda-Pineda (INRIA)

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	D3.2: Audio-visual speaker tracking in relevant environments
Responsible Partner	BIU
Work Package	WP3: Robust Audio-visual Perception of Humans
Task	T3.1: Audio-visual Speaker Detection & Tracking
Version & Date	01/06/2022
Dissemination	Public Deliverable

CONTRIBUTORS AND HISTORY

Version	Editor	Date	Change Log
1	BIU	05/05/2022	First Draft
2	INRIA	03/06/2022	Second Draft
3	BIU	04/06/2022	Third Draft
4	BIU	27/07/2022	Final Version

APPROVALS

Authors/editors	Luis G Camara (INRIA), Sharon Gannot (BIU), Chris Reinke (INRIA), Xavier Alameda-Pineda (INRIA)
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

Executive Summary **3**

1 Introduction **5**

2 Visual Localisation and Tracking **6**
2.1 Training models on the fisheye camera . . . 6
2.1.1 Transforming datasets to the fisheye perspective . . . 6
2.1.2 Manually annotating an ARI-specific dataset . . . 7
2.1.3 Model training . . . 7
2.1.4 Tracking results . . . 8
2.1.5 Video tracker implementation . . . 9
2.2 Audio-visual fusion . . . 9
2.3 Conclusions . . . 10

3 Audio-based Concurrent Speakers Localisation and Tracking **11**
3.1 Time-frequency features . . . 11
3.2 FCN for DOA estimation . . . 12
3.3 Training phase . . . 12
3.4 Experimental Study . . . 13
3.4.1 Experimental setup . . . 13
3.4.2 Speaker localization results . . . 14
3.5 Conclusions and Next Steps . . . 16

4 Conclusions **18**

Bibliography **19**

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

Executive Summary

Deliverable 3.2 reports the progress on task T3.1 on Audio-visual Speaker Detection & Tracking, which is part of WP3: Robust Audio-visual Perception of Humans. The goal of task 5.3 is developing the multi-party conversational system that will be deployed on ARI, the robotic platform designed by PAL Robotics for the SPRING project. This deliverable provides the preliminary software package for multi-party automatic speech recognition (ASR) with speech enhancement algorithms T3.2 & T3.3, and conversational system. The code can be found at the project's repository,[1] and will be made available up to 4 years after the end of the project.

The main achievements reported in this document are:

1. Visual localisation and tracking of humans adapted to the data acquired by ARI's fisheye camera, specifically at Broca Hospital.
2. Model trained using fisheye images and encapsulated in a docker container. The code is also available as a robot operating system (ROS) package.
3. Detection of active sources in a visual scene (speakers that utter sound) and projecting the results of a naïve sound localiser, Open embeddeD Audition System (ODAS), on the image.
4. Audio-only concurrent speaker localisation and tracking algorithm, implemented in Python and successfully tested with moving human speakers at Bar-Ilan University (BIU) acoustic lab in adverse reverberation conditions.

[1] `https://gitlab.inria.fr/spring/wp3_av_perception`

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 context, the current deliverable D3.2 is an upgrade of D3.1, which described the methods and the software used for "Audio-visual speaker tracking in realistic environments." The reader is referred to D3.1 for background content, as D3.2 is meant to include only the most recent developments.

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

2 Visual Localisation and Tracking

As a reminder, the goal of this module is the detection, identification and tracking of people over time using visual data. In D3.1, a state-of-the-art multi-person visual tracker known as fair multi-objective tracking (FairMOT) [23] was introduced. In D3.2, some of the original FairMOT models based on the residual neural network (ResNet34) [12] architecture have been compared with newly trained models that are better adapted to the non-rectilinear perspective characteristics of the fisheye camera.

2.1 Training models on the fisheye camera

The main requirement to train new models for multi-person tracking is the availability of suitable annotated images. In our case, these images should be compatible with the fisheye camera perspective. We considered two approaches to obtain them: i) transforming available annotated datasets containing wide angle panoramic images into the fisheye perspective and ii) using actual fisheye images directly recorded with ARI and manually annotated.

2.1.1 Transforming datasets to the fisheye perspective

Since a dataset with images originated in ARI's fisheye camera was not immediately available, a first attempt was made to adapt an existing annotated dataset by transforming 360° images from the JackRabbot dataset [16] into the fisheye geometry. Panoramic images should be suitable to reconstruct fisheye images in the horizontal direction, as the latter require a wide angle of roughly 180°. However, these panoramics do not cover a very wide angle in the vertical direction (see Figure 2.1).

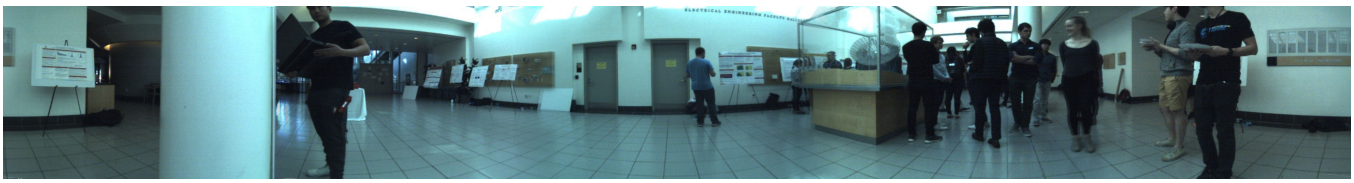


Figure 2.1: Example of a 360° panoramic image from the JackRabbot dataset.

Figure 2.2: Left: attempt to transform the central part of the panoramic image of Fig 2.1 into the fisheye geometry. Right: image captured from ARI's front fisheye camera.

Not having enough vertical coverage poses a problem because people in the scene at short distances from ARI are arguably the most interesting to track accurately. However, they will most likely appear cut down at the bottom

and the top in images from this dataset. Left image of Figure 2.2 shows an attempt to convert the central part of the panoramic image in Figure 2.1 to the fisheye geometry. We used the package OmniCam[1] to do the transformation. As can be seen when compared with an actual ARI fisheye image on the right, there is a lot of missing information on the top and bottom and therefore this approach was considered not viable at this point.

2.1.2 Manually annotating an ARI-specific dataset

Our second approach consisted on recording sequences of images that are specific to the ARI robot. In particular, sequences from the front fisheye camera were recorded using several instances of ARI, i.e., from Inria, from Heriot-Watt University (HWU), and from the robot used at Broca hospital in Paris in a data collection campaign that took place in April, 2022.

A total of 6 sequences were considered for training (see Figure 2.3 for examples). The number of images per sequence was in the range 1000-2000. Some training sequences contained only one person, others only two and yet others more than 4-5 people. The training dataset was split in two subsets, one containing two sequences with a total of 2502 images recorded at Broca hospital and another containing 4 sequences and 5605 images, recorded in several rooms and labs at the premises of the different partners involved. The illumination conditions between the sequences were significantly different.

In addition to the training dataset, another sequence consisting of 1225 images was used for testing the trained models. It was a rather challenging sequence due to the fact that both the robot and the people in the scene were constantly moving, the latter often crossing their paths as well as leaving and entering the scene several times. There was also a significant number of blurred images due to the movement. The sequence was recorded at the waiting room of the Broca hospital, which is the room where ARI is expected to operate once deployed.

(a) Inria (b) HWU (c) HWU

(d) HWU (e) Broca (waiting room) (f) Broca (hall)

Figure 2.3: Example of fisheye camera images from the sequences used for training.

All sequences were annotated at Inria frame by frame, using the CVAT (`https://cvat.org`) software. An example of an annotated image from the testing sequence is depicted in Figure 2.4, which also shows the CVAT working environment.

2.1.3 Model training

Initially, the image resolution utilised from the fisheye camera was 1280x960. After some tests, however, it became clear that reducing the image size to 640x480 did not affect tracking performance while significantly speeding up

[1] https://gitlab.inria.fr/robotlearn/omnicam

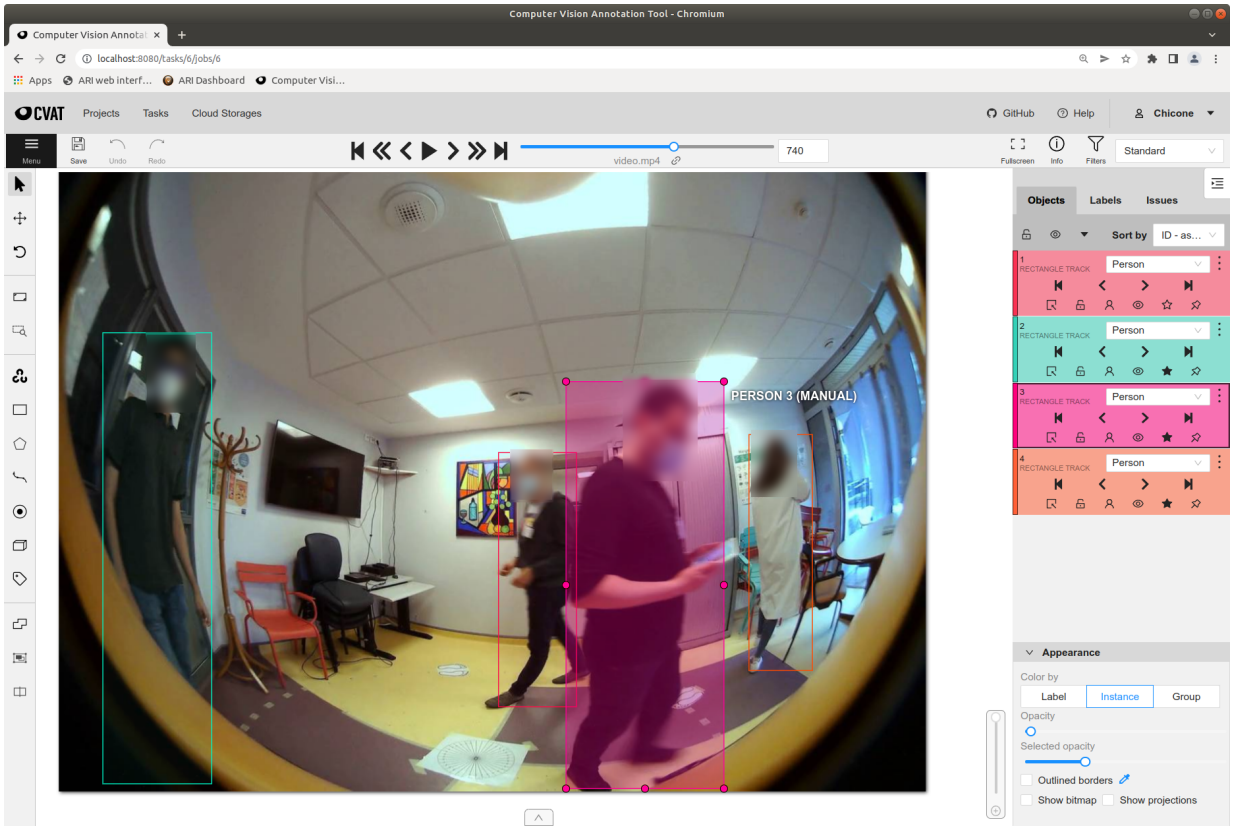


Figure 2.4: Example of an image from the testing sequence while annotating using the tool CVAT.

both training and tracking. Since the original FairMOT models were trained with images at a much larger resolution of 1088x608, we opted for retraining them using the same datasets as in the original paper (CrowdHuman (CH) [21] and MIX datasets [23]) but at a smaller resolution of 800x448, hence maintaining the aspect ratio. We obtained in this fashion two baseline models, one for CH and one for MIX. The latter was trained using the former to initialize weights. Training run during 30 epochs for each model using a batch size of 128.

Another important modification during training was the increase in the dimensionality of the Re-Identification (ReID) branch from 128 as in the original paper to 512. This improved the robustness of the tracker, especially to identity switches and in situations where people were leaving the scene and re-entering a while after.

2.1.4 Tracking results

Table 2.1 shows multiple object tracking accuracy (MOTA) performance metrics [2] results in the test sequence for the different trained models. This metric is widely used to measure the performance of a tracker with a single number. *Baseline* correspond to models whose training images are not of the fisheye type (CH and MIX). Models in the column *Labs* were fine-tuned from the respective baselines using fisheye sequences *a-d* in Figure 2.3. Column *Broca* contains results for models fine-tuned on sequences *e-f* whereas the last column are for models fine-tuned on all training sequences.

Table 2.1: MOTA/epochs/detection-threshold on the test sequence for the different trained models.

Pretrained	Baseline	Labs	Broca	Labs+Broca
CH	65.6/30/0.35	72.9/15/0.4	75.4/15/0.35	**76.2**/10/0.35
CH+MIX	68.4/30/0.35	73.9/10/0.45	71.6/10/0.45	74.3/10/0.35

It is clear from the table that fine-tuning on ARI's fisheye images represent a major improvement in tracking performance with respect to training on images from more standard cameras, as is the case for CH and MIX datasets. The fisheye perspective introduces heavy distortions that are extreme towards the edges of the image. The tracking

models trained on CH and MIX work reasonably well towards the center of the fisheye images but they start to fail as we depart from these central locations.

Considering the small size of the newly annotated dataset, whose number of images are 5605 and 2502 for the *Labs* and *Broca* subsets, respectively, we took care not to overfit during training. Therefore, we closely monitored the performance as a function of the number of epochs and stopped the training at an optimal point. This is given by the central number in the cells of Table 2.1, after the MOTA values. The last number of each cell provides the detection threshold used during tracking.

Notice from the table that the largest increase in tracking performance as compared with the baselines is due to the use of fisheye images in the training (using *Labs* subset), and to a lesser extent the involvement of the same robot and environment during training and during testing (*Broca* subset). The best MOTA measure was achieved when fine-tuning the CH baseline using both subsets.

Some visual examples of the tracker on the test sequence for the best model are shown in Figure 2.5. One can clearly see in (b) and (c) that even people located on the edges of the images can be detected by the tracker, something we did not observe using the models trained on the CH and MIX datasets.

(a) (b) (c)

Figure 2.5: Tracking examples of the best performing model on the test sequence.

2.1.5 Video tracker implementation

The best performing model and the visual tracking code have been encapsulated in a docker container[2] that can run on an external computer (basestation), provided ARI is publishing the required images from the fisheye. The tracker was successfully tested and code is also available as a ROS package.[3]

A recommendation is made at this point to set the fisheye publishing frame rate in ARI to more than the default 5 fps. A frame rate of 20 fps was successfully evaluated yielding much smoother tracking capabilities, while still meeting real-time constraints.

2.2 Audio-visual fusion

We are currently working on a first version of the audio-visual fusion module, and in particular on *speaker diarisation*, namely determining "who speaks when". Specifically, we are aiming at the task of detecting the active speaker(s) in a scene based on a combination of the audio and visual modalities. We started to integrate the visual tracker with a simple audio tracker known as ODAS [9], specifically implemented for robot audition tasks. This module will be substituted in the near future with a convolutional neural network (CNN)-based system developed by BIU [11] (see Chapter 3). ODAS implements sound source localisation algorithm, which combines the classical steered response power with phase transform (SRP-PHAT) method, enhanced by hierarchical search with directivity model and automatic calibration (HSDA), followed by a tracking algorithm supported by a Kalman filter. The package can be used out-of-the-box for ARI's microphone array. Nevertheless, it required some software development to share the hardware (specifically, to be able to use the microphones simultaneously by other sound processing modules), and ROS integration. Tracked sound sources are given by ODAS as unit vectors pointing to them (i.e. direction-only), in the microphone frame. Since we do not actually know the distance of the sound source but only its direction, we have to set an arbitrary distance (2 or 3 m, for example) in order to obtain an approximated 3D position of the sound source in the microphone frame.

[2] Docker can be pulled with: "docker pull registry.gitlab.inria.fr/spring/dockers/wp3_tracker:lowres"
[3] https://gitlab.inria.fr/spring/wp3_av_perception/multi-person_visual_tracker

Projecting the location of the sound sources into the image plane, necessitates a modification of the sound frame to the camera frame. For that, we used the calibration parameters of the camera, and projected the point into the plane image using standard ROS functions (cameraModel.project3dToPixel).

Figure 2.6 depicts an example of this fusion process, where three speakers took turns to speak, with no overlap, from left to right in successive order in the three images. Each sound source is denoted by a blue circle with a number inside indicating to which bounding box the source is assigned. We used the following assignment procedure. Based

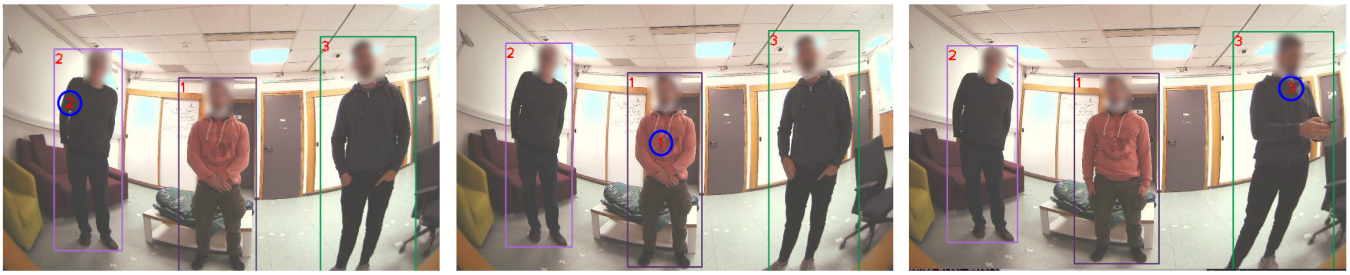


Figure 2.6: Example of audio-visual fusion between sound source localisation provided by the ODAS algorithm (blue circles) and the visual tracking module (bounding boxes).

on the location of the sound source (blue circles), a distance cost matrix is calculated with respect to the upper-central part of the detected bounding boxes, where the head of the speaker is expected to be. Then, the sound source is assigned to the most likely person (the number inside the circles).

This is a preliminary implementation that nonetheless sets a framework that can be used in the future to seamlessly add more properties to the calculation of the cost matrix, for instance, the power of the audio signal.

2.3 Conclusions

As seen in this document, and in the context of multi-person visual tracking, models trained using images from standard cameras with more or less rectilinear perspective can be used to some extent on images with fisheye perspective. However, the high distortion present in fisheye images have a detrimental effect on the tracking performance, especially towards the edges of the images.

In order to counterbalance this problem, we have recorded and annotated several thousand fisheye images, and fine-tuned models based on rectilinear perspective to achieve a much higher level of tracking performance, improving MOTA by more than 10% on the tested dataset. We expect that further extension of our fisheye dataset, especially from the Broca hospital environment, will allow us to improve even more the robustness of the tracker.

We have also started work on an audio-visual fusion approach that will serve as a framework to combine information from both the video tracker and the audio processing modules.

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

3 Audio-based Concurrent Speakers Localisation and Tracking

We present a deep neural network-based online multi-speaker localisation algorithm. Following the W-disjoint orthogonality (WDO) principle in the spectral domain, each time-frequency (TF) bin is dominated by a single speaker, and hence by a single direction-of-arrival (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. The algorithm was evaluated using recordings of moving human speakers. Deploying the algorithm in ARI necessitates re-training.

3.1 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:

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

where h_m^i is the room impulse response (RIR) relating the i-th speaker and the m-th microphone. In the short-time Fourier transform (STFT) domain, (3.1) can be written as (provided that the frame-length is sufficiently large with respect to (w.r.t.) the filter length):

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

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

The STFT (3.2) 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 relative transfer function (RTF) [8] 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 $z_{\text{ref}}(l,k)$:

$$\text{iRTF}(m,l,k) = \frac{z_m(l,k)}{z_{\text{ref}}(l,k)}. \tag{3.3}$$

Note, that the reference microphone is arbitrarily chosen. Reference microphone selection is beyond the scope of this chapter (see [22] for a reference microphone selection method). The input feature set extracted from the recorded signal is thus a 3D tensor $\mathcal{R}$:

$$\mathcal{R}(m,l,k) = [\mathfrak{Re}(\text{iRTF}(m,l,k)), \mathfrak{Im}(\text{iRTF}(m,l,k))]. \tag{3.4}$$

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 [18] 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 each TF bin from the given mixed recorded signal.

3.2 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 random variable (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:

$$p_{l,k}(\theta) = p(D_{l,k} = \theta|\mathcal{R}), \quad \theta \in \Theta. \tag{3.5}$$

For this task, we use a deep neural network (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. A fully convolutional network (FCN) is used to compute (3.5) 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 [19]. We dub our algorithm time-frequency direction-of-arrival net (TF-DOAnet).

The input to the network is the feature matrix $\mathcal{R}$ (3.4). 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 $P = 2M - 2$ with M 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 (3.3), 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:

$$\text{VAD}(l, k) = \begin{cases} 1 & |z_{\text{ref}}(l, k)| \geq \epsilon \\ 0 & \text{o.w.} \end{cases}, \tag{3.6}$$

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

$$p_l(\theta) = \frac{1}{K'} \sum_{k=1}^{K} p_{l,k}(\theta)\text{VAD}(l, k). \tag{3.7}$$

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 block diagram.

3.3 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 [1]. 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 3.1. 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 Wall Street Journal 1 (WSJ1) database [17] 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 at 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.

The contributions of the two sources were then summed with a random signal to interfering ratio (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

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

Table 3.1: 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				

Table 3.2: 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	

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. 3.1. 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:

$$\text{dominant speaker} \leftarrow \underset{i}{\operatorname{argmax}} |s^i(l,k) h^i_{\text{ref}}(l,k)|. \tag{3.8}$$

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 [13]. 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.

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 [20]. 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.

3.4 Experimental Study

3.4.1 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.

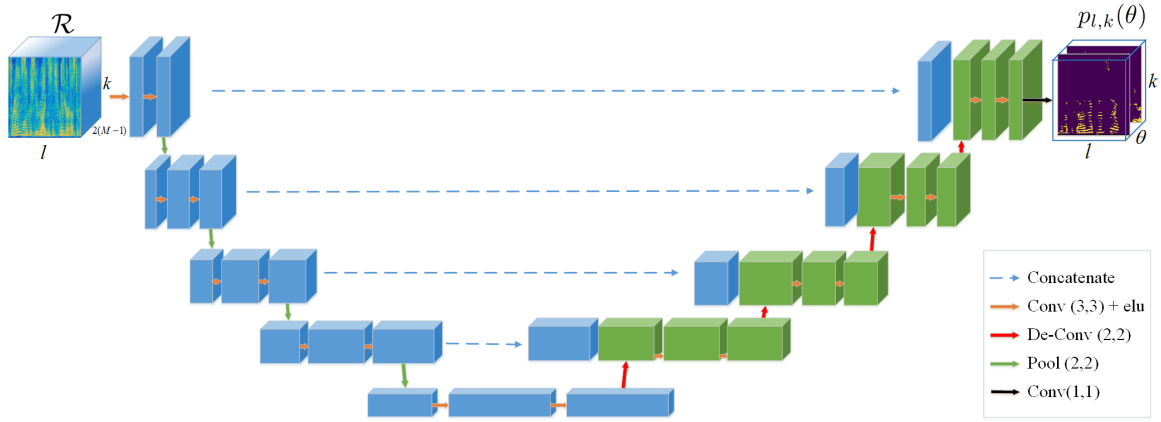


Figure 3.1: U-net architecture for DOA-mask speech separation. The blue blocks depict the encoder and the green blocks depict the decoder.

For each test scenario, we selected two speakers from the test set of the WSJ1 database [17], 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

$$\text{MAE}(°) = \frac{1}{N \cdot C} \sum_{c=1}^{C} \min_{\pi \in S_N} \sum_{n=1}^{N} |\theta_n^c - \hat{\theta}_{\pi(n)}^c|, \tag{3.9}$$

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

$$\text{Acc.}(\%) = \frac{\hat{C}_{\text{acc.}}}{C} \times 100 \tag{3.10}$$

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 multiple signal classification (MUSIC) and SRP-PHAT algorithms. In addition, we compared its performance with the CNN multi-speaker DOA (CMS-DOA) estimator [4].[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.

3.4.2 Speaker localization results

Static simulated scenario We first generated a test dataset with simulated RIRs. Two different rooms were used, as described in Table 3.2. 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.

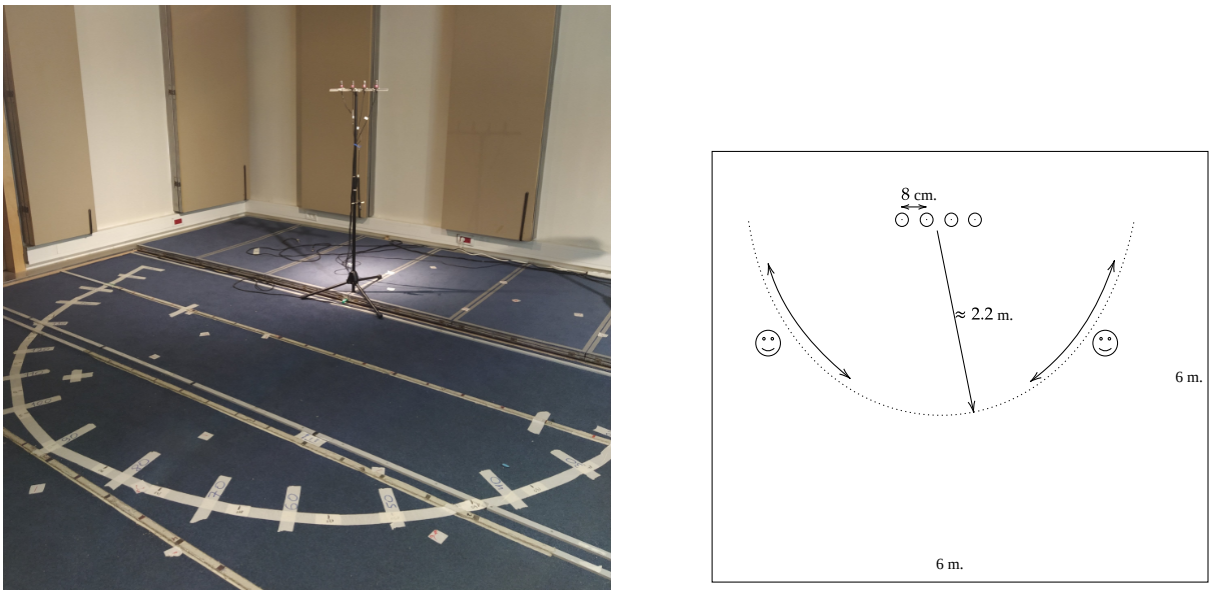
The results for the TF-DOAnet compared with the competing methods are depicted in Table 3.3. The tables shows that the deep-learning approaches outperformed the classic approaches. The TF-DOAnet achieved very high scores and outperformed the DNN-based CMS-DOA algorithm in terms of both MAE and accuracy.

[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 [10]. The database comprises RIRs measured in an acoustics lab for three different reverberation times of $\mathrm{RT}_{60} = 0.160, 0.360$, 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 3.4. 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 $\mathrm{RT}_{60} = 610$ ms, and for the 2 m case, for $\mathrm{RT}_{60} = 360$ 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 [8, 15, 7] and for speaker localization [5, 14]. 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 $\mathrm{RT}_{60} = 610$ ms was still better than the performance for $\mathrm{RT}_{60} = 160$ ms.



(a) Room view.

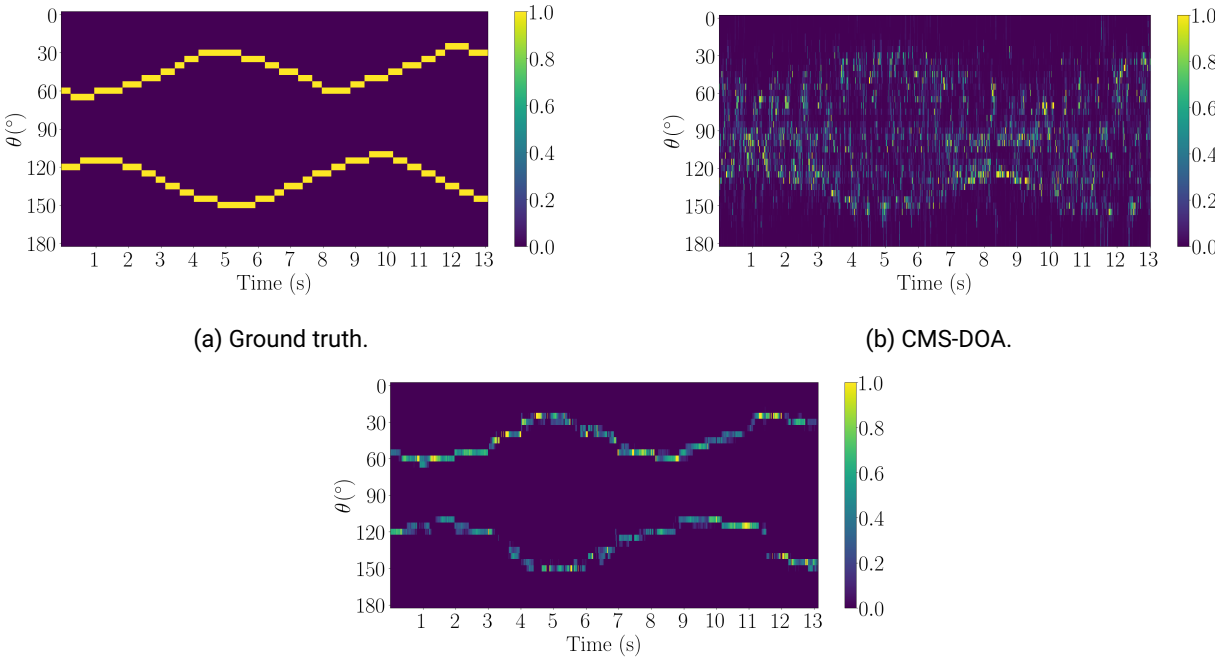
(b) Speakers' trajectory.

Figure 3.2: Real-life experiment setup.

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 3.2a depicts the real-life experiment setup and Fig. 3.2b 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

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



(a) Ground truth.

(b) CMS-DOA.

(c) The TF-DOAnet.

Figure 3.3: Real-life recording of two moving speakers in a $6 \times 6 \times 2.4$ room with $\mathrm{RT}_{60} = 720$ ms.

Table 3.3: Results for two different test rooms with simulated RIRs

Test Room	Room 1		Room 2	
Measure	MAE	Acc.	MAE	Acc.
MUSIC [6]	26.2	28.4	31.5	16.9
SRP-PHAT [3]	25.1	26.7	35.0	15.6
CMS-DOA [4]	13.1	71.1	24.0	38.1
TF-DOAnet	**0.3**	**99.5**	**1.7**	**94.3**

$100°$, respectively, turned around and walked back to their starting point. This was done several times throughout the recording. Figure 3.3 depicts the results of the this experiment for $\mathrm{RT}_{60} = 720$ ms.

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. Figure 3.4 depicts the results of the this experiment for $\mathrm{RT}_{60} = 720$ ms.

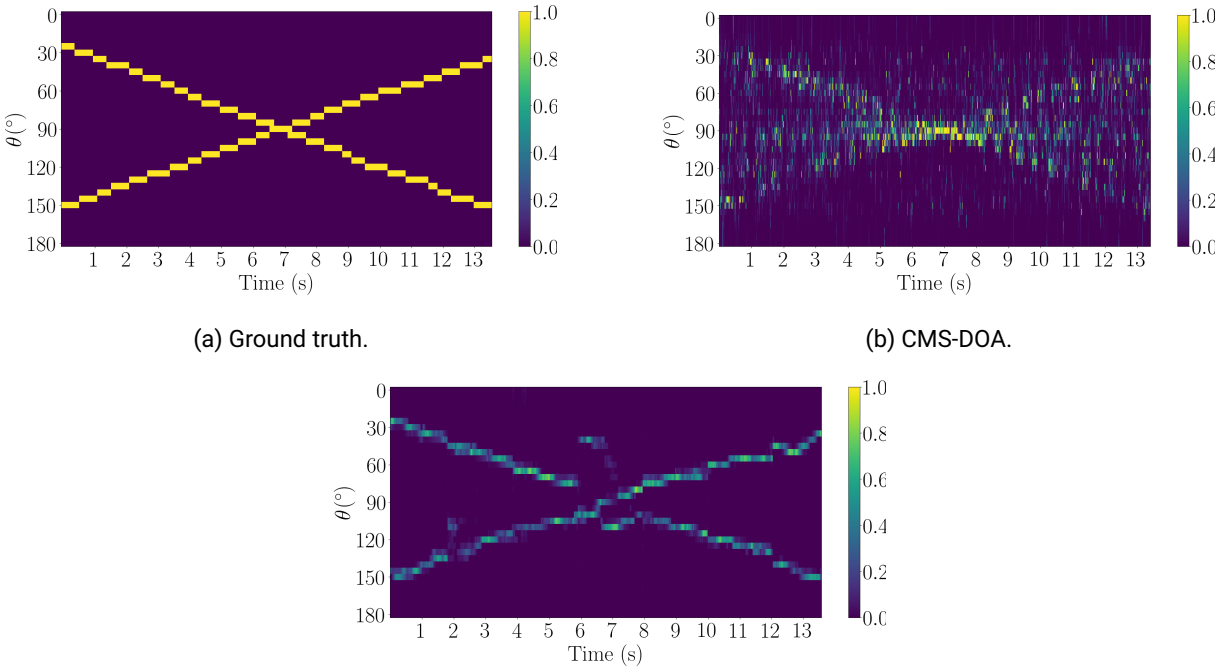
It is clear that the TF-DOAnet outperformed the CMS-DOA algorithm, especially for the high RT_{60} conditions. Whereas the CMS-DOA fluctuated rapidly, the TF-DOAnet output trajectory was smooth and noiseless.

3.5 Conclusions and Next Steps

The audio tracking algorithm is capable of tracking multiple speakers in highly reveberant environment. This was verified at BIU acoustic lab with reverberation level set to $\mathrm{RT}_{60} = 720$ ms (which is higher than the expected reverberation level measured at Broca in the room where ARI is expected to operate once deployed.

The tracking algorithm will be retrained with the data collected at BROCA (especially, the room impulse responses from sources encircling ARI and its microphone array). The BROCA environment is acoustically very challenging. Moreover, natural scenes with several people moving in an unstructured manner are known to be very challenging. Only an elaborated evaluation campaign, to take place at BROCA, can determine the tracking capabilities of the algorithm and its performance bounds.

The algorithm is implemented in Python. It will be shortly migrated to ROS and tested on ARI. As retraining may be required, we will use the RIRs recorded at Broca, during the recent data collection, to generate suitable training data.



(a) Ground truth.

(b) CMS-DOA.

(c) The TF-DOAnet.

Figure 3.4: 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.

Table 3.4: 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**

Integration with the visual tracker will follow the framework described in Sec. 2.2.

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 both visual-based and audio-based localization and tracking modules. The visual tracker has already been implemented under ROS. The audio tracker is, currently, only implemented as an independent package in Python.

Moreover, we also report preliminary results on audio-visual fusion. By projecting the direction of arrival of audio sources onto the visual tracker images, we were able to assign a person to each audio source.

In the next steps, we will work on the integration of the TF-DOAnet audio tracker with the multi-person visual tracker.

Special attention will be given to complex scenes, with moving and concurrently speaking speakers, desired and undesired speakers uttering speech out of the visual scene, and static interfering noise.

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

Bibliography

[1] Jont B. Allen and David A. Berkley. Image method for efficiently simulating small-room acoustics. *The Journal of the Acoustical Society of America*, 65(4):943–950, 1979.

[2] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008.

[3] Michael S. Brandstein and Harvey 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.

[4] Soumitro Chakrabarty and Emanuël A. P. Habets. Multi-speaker DOA estimation using deep convolutional networks trained with noise signals. *IEEE Journal of Selected Topics in Signal Processing*, 13(1):8–21, 2019.

[5] Antoine Deleforge, Florence Forbes, and Radu Horaud. Acoustic space learning for sound-source separation and localization on binaural manifolds. *International journal of neural systems*, 25(01):1440003, 2015.

[6] Jacek P. Dmochowski, Jacob Benesty, and Sofiene 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.

[7] Ivan Dokmanić, Robin Scheibler, and Martin Vetterli. Raking the cocktail party. *IEEE journal of selected topics in signal processing*, 9(5):825–836, 2015.

[8] 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.

[9] François Grondin, Dominic Létourneau, Cédric Godin, Jean-Samuel Lauzon, Jonathan Vincent, Simon Michaud, Samuel Faucher, and François Michaud. Odas: Open embedded audition system. *arXiv preprint arXiv:2103.03954*, 2021.

[10] Elior Hadad, Florian Heese, Peter Vary, and Sharon Gannot. Multichannel audio database in various acoustic environments. In *International Workshop on Acoustic Signal Enhancement (IWAENC)*, 2014.

[11] 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 Processing*, 2021(1):1–10, 2021.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

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

[14] Bracha Laufer-Goldshtein, Ronen Talmon, and Sharon Gannot. Semi-supervised sound source localization based on manifold regularization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(8):1393–1407, 2016.

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

[16] Roberto Martin-Martin, Mihir Patel, Hamid Rezatofighi, Abhijeet Shenoi, JunYoung Gwak, Eric Frankel, Amir Sadeghian, and Silvio Savarese. JRDB: A dataset and benchmark of egocentric robot visual perception of humans in built environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2021.

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

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

[18] Scott Rickard and Ozgiir Yilmaz. On the approximate w-disjoint orthogonality of speech. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2002.

[19] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 2015.

[20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

[21] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018.

[22] Sebastian Stenzel, Jürgen Freudenberger, and Gerhard 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.

[23] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *arXiv e-prints*, pages arXiv–2004, 2020.