DAAP Homework #2

Space-Time-Based Source Separation

For this homework assignment, you will be exploring the topic of acoustic source separation.

You will be provided with two mixture signals $y_1(n)$ and $y_2(n)$, corresponding to files `y1.wav` and `y2.wav`, respectively. These signals have been acquired using with $M = 2$ ideal microphones placed at $d = 9$ cm from one another. Each microphone captures the mixture of $N = 3$ speech signals, i.e., $s_1(n), s_2(n),$ and $s_3(n)$, having a sampling frequency of 8 kHz.

The speakers are located at $\theta_1 = 30°$, $\theta_2 = 85°$, and $\theta_3 = -40°$, respectively. These directions of arrival (DOAs) are referred to the normal direction to the line passing through the two microphones (similarly to the ULA configuration.) All speakers are located at 75 cm from the reference microphone, i.e., microphone 1.

Each microphone is characterized by a i.i.d. Gaussian self-noise, which is uncorrelated with the source signals. For this homework, we will also assume that i) reverberation is absent, ii) the sensor noise has a standard deviation of $\sigma = 10^{-3}$, and iii) the speed of sound is $c = 340$ m/s.

You will implement source separation by applying binary masking to the short-time Fourier transform (STFT) of the mixture signal $y_1(n)$. To design the mask, you will first extract a feature vector for each of the time-frequency locations (m, ω_k) of the STFT. Then, you will apply a clustering algorithm (e.g., k-means) with $k = N = 3$ clusters. Namely, the cluster index associated to the pair (m, ω_k) will determine whether the corresponding time-frequency location in the binary mask $M_\ell(m, \omega_k)$ should be set to either 1 or 0. In other words, this corresponds to

$$M_\ell(m, \omega_k) = \begin{cases} 1 & \text{if } (m, \omega_k) \text{ belongs to cluster } \ell \\ \quad 0 & else \end{cases}$$

Having designed the three binary masks, the estimate of the source signals can be retrieved by taking the Hadamard product between the corresponding mask and the STFT of the reference microphone signal, i.e., $Y_1(m, \omega_k)$.

For this homework, we recommend the following 3-dimesional feature vector, but you are encouraged to experiment with other types of space-time features.

$$\Phi(m, \omega_k) = [A_1(m, \omega_k), A_2(m, \omega_k), P(m, \omega_k)]^T,$$

where

$$A_i(m, \omega_k) := \frac{|Y_i(m, \omega_k)|}{\sqrt{\sum_{j=1}^{M} |Y_j(m, \omega_k)|^2}}, \qquad P(m, \omega_k) := \frac{1}{2\pi} \arg \frac{Y_2(m, \omega_k)}{Y_1(m, \omega_k)}.$$

As a reference, you will be provided with the source signals (`s1.wav`, `s2.wav`, `s3.wav`), as well as exemplary separated sources (`separated_1.wav`, `separated_2.wav`, `separated_3.wav`). Please note that these files are only for reference purposes and should not be used in implementing the source separation algorithm.

Please provide and be ready to discuss the following plots:

- log-amplitude spectrograms for the mixture signals (subplot).
- log-amplitude spectrograms of the true and estimated source signals (subplots).
- Binary masks, in black and white (subplot).
- Density plot each pair of features, i.e., $[A_1(m, \omega_k), A_2(m, \omega_k)]$, $[A_1(m, \omega_k), P(m, \omega_k)]$, $[A_2(n, \omega_k), P(n, \omega_k)]$.
- Accompany the density plots with the histograms of the individual features $A_1(m, \omega_k), A_2(m, \omega_k), P(m, \omega_k)$.

Make sure to label the axes correctly, e.g., by expressing time in seconds and frequency in Hertz.

You are tasked to implement the short-time Fourier transform yourself. Do not use the built-in `stft`/`istft` functions from, e.g., MATLAB, `scipy`, or `librosa`—just to name a few. On the contrary, feel free to use whichever ready-made implementation of the clustering algorithm is available to you (e.g., `scikit-learn`). While there is no specific programming language required for this assignment, we strongly recommend using either MATLAB or Python.

Lastly, you are expected to explain and justify all your design choices and refer to the course materials if necessary.

Assignment rules:

- Groups of **at most 2 people** are allowed.
- Each group should upload the code on WeBeep as a **single zip file.** One student will submit a zip file for the entire group; **do not upload the same HW twice.**
- The zip file should be named with the surnames of all group members, e.g., Mario Rossi and Maria Bianchi will upload a file named `DAAP_HW1_Rossi_Bianchi.zip`
- Please **include all synthesized audio files** (i.e., the source signal estimates for the three speakers) in the zip folder, e.g., `Rossi_Bianchi_s1_hat.wav`, `Rossi_Bianchi_s2_hat.wav`, `Rossi_Bianchi_s3_hat.wav.`
- Shortly after the submission, you will discuss your implementation during a **short oral presentation** (about 15 minutes.)
- A bonus of **3/30** is awarded to those groups who upload the HW on WeBeep **by 11:59 pm on May 6th, 2023.**
- Only the groups that have already submitted their code are entitled to discuss the HW with the instructor.
- Oral presentations will be held either in presence or via Webex. The modality will be agreed upon on a case-by-case basis.
- Refer to the course rules and the most updated grading scheme for subsequent deadlines to avoid late-submission penalties.