Separation Performance Evaluation

Table of Contents

Introduction

This script reproduces the result of the experiment described in Section 5.2. of our paper [1].

Download dataset

Please download the 'Impulse_response_Acoustic_Lab_Bar-Ilan_University_(Reverberation_0.160s)_3-3-3-8-3-3-3.zip' [2] and place it in your current directory.

Note: because of the file's large size, you will need to access the link above and download the file **manually**.

Run

```
% unzip dataset
if ~exist("Impulse_response_Acoustic_Lab_Bar-
Ilan_University_(Reverberation_0.160s)_3-3-3-8-3-3-3","dir")
    mkdir Impulse_response_Acoustic_Lab_Bar-
Ilan_University_(Reverberation_0.160s)_3-3-3-8-3-3-3
    unzip("Impulse_response_Acoustic_Lab_Bar-
Ilan_University_(Reverberation_0.160s)_3-3-3-8-3-3-3.zip",...
          "Impulse_response_Acoustic_Lab_Bar-
Ilan_University_(Reverberation_0.160s)_3-3-3-8-3-3-3")
end

% install BSS-eval
if ~exist("bss_eval-master","dir")
    unzip("https://gitlab.inria.fr/bass-db/bss_eval/-/archive/master/bss_eval-
master.zip")
end
addpath(fullfile("bss_eval-master","v3.0"))

% run
addpath(genpath("IVA"));
util_loadSampleMixture; % download SiSEC
SDRi = nan(4*56,6);
mixtureIdx = 0;
count = 0;
```

```
wb = waitbar(count);
for mixingSystemIdx = 1:4
    for sourcePairIdx = 1:56
        mixtureIdx         = mixtureIdx + 1;
        [x,s,fs]           = util_reproduceExperiment1Mixture(mixingSystemIdx,
sourcePairIdx);

        SDRi(mixtureIdx,1) = util_calcSDRi(s,x,run_IVA_FastADMM(x));
        SDRi(mixtureIdx,2) = util_calcSDRi(s,x,run_IVA_ADMM(x));
        SDRi(mixtureIdx,3) = util_calcSDRi(s,x,run_IVA_PDS(x));
        SDRi(mixtureIdx,4) = util_calcSDRi(s,x,run_IVA_AuxIP2(x));
        SDRi(mixtureIdx,5) = util_calcSDRi(s,x,run_IVA_AuxISS(x));
        SDRi(mixtureIdx,6) = util_calcSDRi(s,x,run_IVA_AuxIP(x));

        waitbar(count/224,wb,count+" / 224");
        count = count + 1;

        boxchart(SDRi);
        xticklabels(["FastADMM","ADMM","PDS","IP","ISS","IP2"]);
        ylabel("SDRi [dB]");
        drawnow
    end
end
close(wb)

% visualize the result
boxchart(SDRi);
xticklabels(["FastADMM","ADMM","PDS","IP","ISS","IP2"]);
ylabel("SDRi [dB]");
```

Reference

1. H. Watarai, K. Matsumoto, and K. Yatabe, "Fast and flexible algorithm for determined blind source separation based on alternating direction method of multipliers," Acoustical Science and Technology, **47**(1), pp. XX–XX (2026).
2. E. Hadad, F. Heese, P. Vary, and S. Gannot, “Multichannel audio database in various acoustic environments”, in International Workshop on Acoustic Signal Enhancement (2014).