denoiseAudio

Mathieu Lagrange

December 7, 2015

Introduction

This report documents the fourth demonstration of the use of the expLanes framework to conduct a computational experiment. The project denoiseAudio is about the use of the Non-negative Matrix Factorization (NMF) framework to remove noise background from an audio recording.

Design

The project is divided into three processing steps:

1. **mix**: generate mixture of the target sound with specific level of noise
2. **model**: model the spectrogram of the mixture in order to ease separability
3. **separate**: perform actual separation in order to remove the noise

***mix**: generate mixture of the target sound with specific level of noise*

The target sound is additively mixed with white noise:

```
% mix the source and the noise at a given snr
mixture = data.source+data.noise./10^(.05*setting.snr);
```

***model**: model the spectrogram of the mixture in order to ease separability*

Two methods are considered.First, as an oracle baseline, the Ideal Binary Mask (IBM) is used[1], where the spectrogram of the target and the noise are used as priors to generate the mask.

```
SS = abs(computeSpectrogram(data.source, ...
    config.fftlen, config.samplingFrequency));
% compute the magnitude spectrogram of the noise
SN = abs(computeSpectrogram(data.noise, ...
    config.fftlen, config.samplingFrequency));
% record where the source is dominant in the ...
    spectrogram
store.mask = SN<SS;
```

Then, a simple separation scheme based on the Non negative Matrix Factorization (NMF) algorithm[2] is used. The NMF algorithm is an iterative factorization process and we want to study the impact of the number of iterations on the quality of the separation, the number of iterations is set as a factor in the experiment. Then,

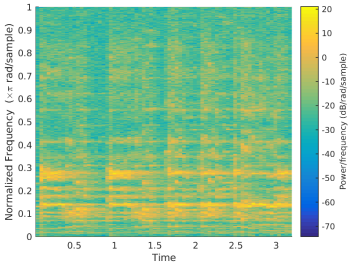

Figure 1: The spectrogram of the "Hallelujah Chorus" of Handel with 20 dB of white noise.

[1] D. Wang. On ideal binary mask as the computational goal of auditory scene analysis. In *Speech separation by humans and machines*, pages 181–197. Springer, 2005

[2] P. Smaragdis. Convolutive speech bases and their application to supervised speech separation. *Audio, Speech, and Language Processing, IEEE Transactions on*, 15(1):1–12, 2007

in order to gain computation time, this factor is set to be sequential, so that settings with increasing number of iterations are done sequentially. The Euclidean distance is considered in this example:

```
% perform nmf optimization
for k=1:nbIterations
    % Euclidean multiplicative updates
    H = H.*(W'*SM)./((W'*W)*H+eps);
    W = W.*(H*SM')'./(W*(H*H')+eps);
end
```

The quality of fit of the NMF model can measured using the Spectral Signal to Reconstruction Ratio (SSRR):

```
% record spectral signal to reconstruction ratio
obs.ssrr = ...
    log(sum(sum((SM).^2))/sum(sum((SM-W*H).^2)));
```

Then, the elements of the dictionary are sorted against a flatness indicator computed as follows:

```
% compute flatness of the dictionary
flatness = (mean(W)-median(W))./mean(W);
[null, order] = sort(flatness, 'descend');
```

This way, peaky spectra, likely to relate to the signal of interest, are modeled by elements of the dictionary which are sorted first. On contrary, flat spectra are modeled by elements of the dictionary with high flatness sorted last.

***separate**: perform actual separation in order to remove the noise*

In the case of the IBM, the generation of the separated spectrogram is:

```
% estimate the magnitude spectrogram of the source
SE = SM.*data.mask;
```

For the NMF scheme, the less spectrally flat elements of the dictionary are selected in order to generate the separated spectrogram:

```
% select the most salient components
nbkept = ...
    floor((100-setting.pruning)/100*setting.dictionarySize);
W = data.W(:, 1:nbkept);
H = data.H(1:nbkept, :);
% estimate the magnitude spectrogram of the source
SE = W*H;
```

Then, from the separated magnitude spectrogram to the sound signal:

```
% generate estimated spectrogram of the source
se = SE.*exp(1i*angle(sm));
% generate estimated signal of the source
e = ispecgram(se, config.fftlen, config.samplingFrequency, ...
    config.fftlen/2, config.fftlen/4);
```

```
e = e(1:length(data.source));
```

Standard evaluation metrics are computed[3], and the widely used Signal to Degradation Ratio (SDR) is considered in the remaining of the report.

[3] E. Vincent, R. Gribonval, and C. Févotte. Performance measurement in blind audio source separation. *Audio, Speech, and Language Processing, IEEE Transactions on*, 14(4):1462–1469, 2006

Definition of factors

Those factors and their corresponding modalities are defined in the file named `deauFactors.txt` whose content is the following:

```
method =2:== {'ibm', 'nmf'}
snr === −40:10:40
nbIterations =2:s=1/2= (1:10)*50
dictionarySize =2:=1/2= 10:10:80
pruning =3=1/2= 0:10:90
```

Most the factor design discussed above is compactly displayed in Figure 2.

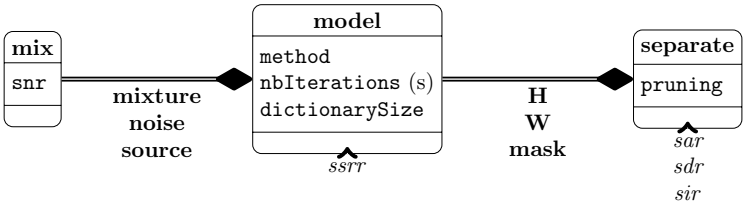

Figure 2: Factor and data flow graph.

Results

Convergence of the NMF scheme

First, in order to check the fitting capability of the NMF scheme, we consider in Figure

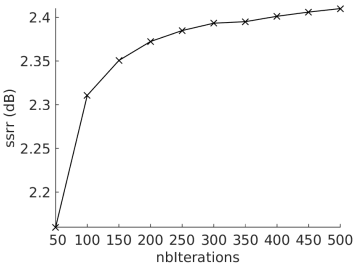

Figure 3: Spectral Signal to Reconstruction Ratio as a function of the number of iterations.

Parametrization of the NMF scheme

The NMF scheme has 2 important parameters to be set in order to ensure its best behavior. First, the number of elements of the dictionary is critical as it should be high enough to allows good reconstruction and small enough not spread parts of the source into multiple non meaningful components. Next, for the purpose of this experiment, the number of elements to be dropped is also important. As shown on Figure 5, the optimal setting is with 16 elements kept among 40 elements. With this setting, the influence of the number of iterations is relatively small, see Figure 4.

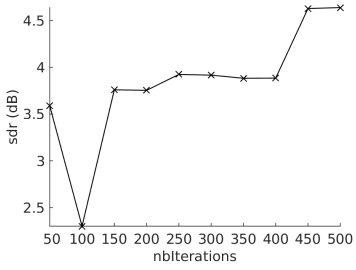

Figure 4: SDR as a function of the number of iterations with 40 % pruning and 40 elements of dictionary.

Overall

The setting selected previously is next studied for various SNRs in Figure 6 and compared to the oracle baseline. On average, the drop

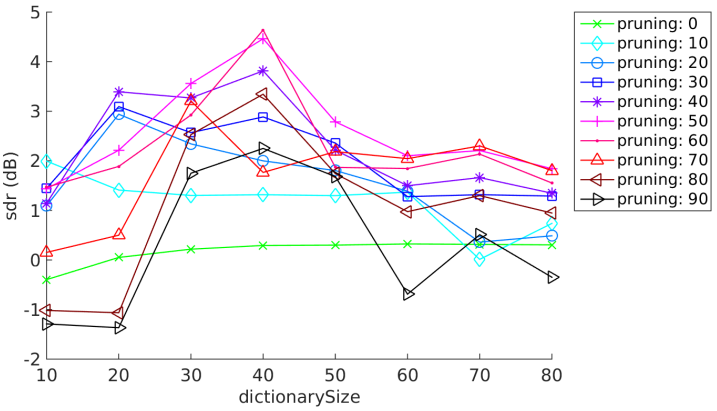

Figure 5: SDR as a function the number of elements in the dictionary and the percentage of pruning.

of performance is about 5 dB.

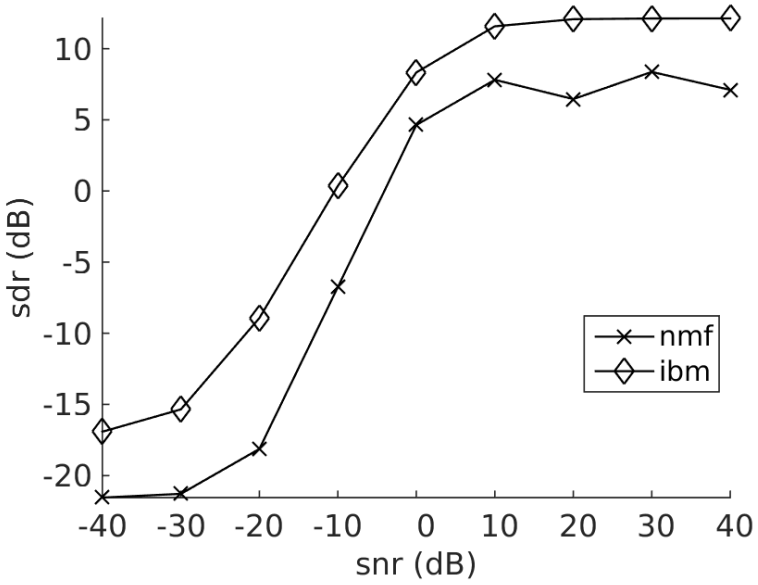

Figure 6: Performance comparison between the NMF scheme in its optimal setting with the IBM oracle baseline