Semi-Supervised Adversarial Audio Source Separation applied to Singing Voice Extraction

Daniel Stoller[1], Sebastian Ewert[2], Simon Dixon[1]

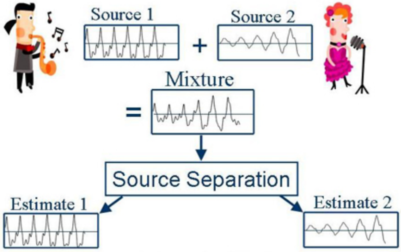
[1]Centre for Digital Music
Queen Mary University London

[2]Spotify

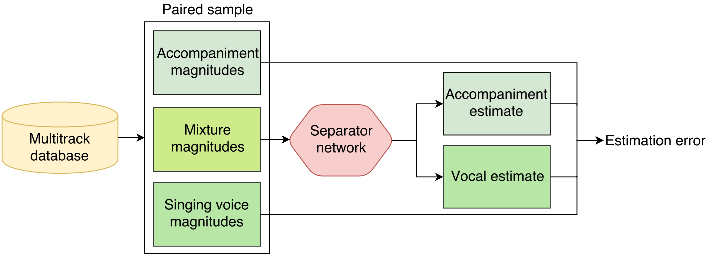
MLSP-L8: Deep Learning III
ICASSP
19.04.2018

Audio source separation

- Task: Recover sources from mixtures
- Example: Music instrument separation:

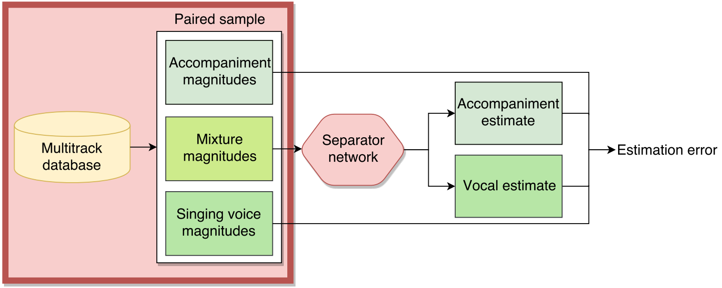


Current state of the art [5, 3, 1]



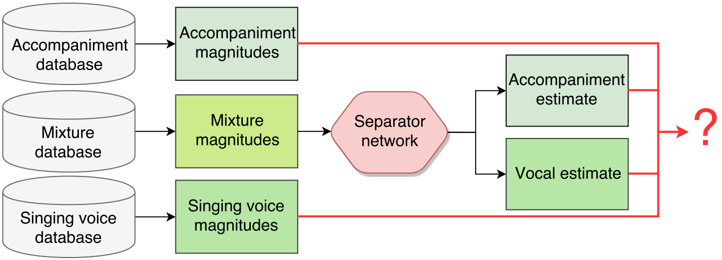
- Training on multitrack datasets
- Neural network
- Discriminative, MSE loss

Current state of the art [5, 3, 1]



- **Training on multitrack datasets (small $\Rightarrow$ overfitting!)**
- Neural network
- Discriminative, MSE loss

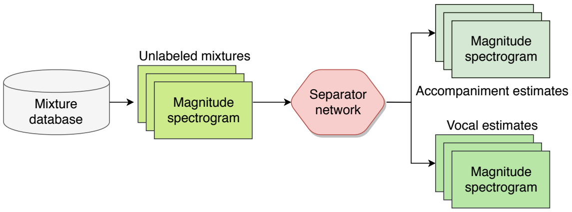
Our goal



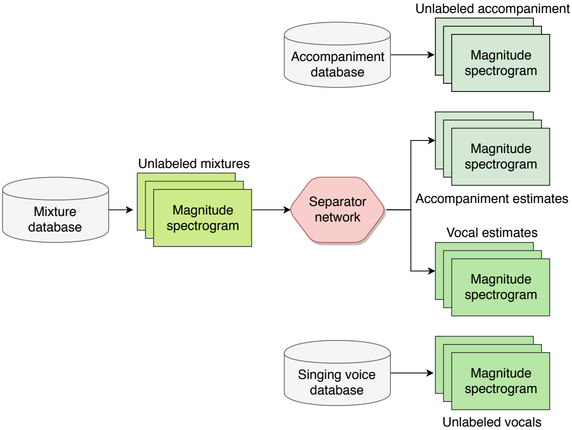
$\Rightarrow$ How to also learn from unpaired mixtures and sources?

- Random mixing ignores source correlations [4, 2]

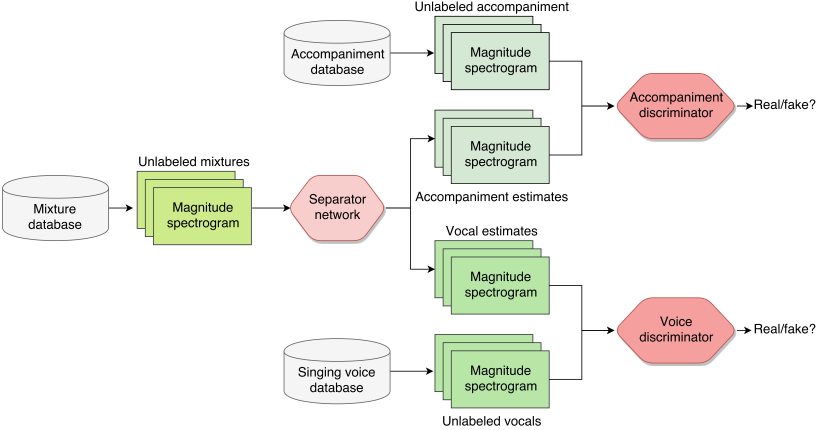
Intuition


Mixture database
Unlabeled mixtures
Magnitude spectrogram
Separator network
Magnitude spectrogram
Accompaniment estimates
Vocal estimates
Magnitude spectrogram

Intuition


Unlabeled accompaniment
Accompaniment database
Magnitude spectrogram
Magnitude spectrogram
Accompaniment estimates
Unlabeled mixtures
Mixture database
Magnitude spectrogram
Separator network
Vocal estimates
Magnitude spectrogram
Singing voice database
Magnitude spectrogram
Unlabeled vocals

Intuition


Unlabeled accompaniment
Accompaniment database
Magnitude spectrogram
Accompaniment discriminator
Real/fake?
Magnitude spectrogram
Accompaniment estimates
Unlabeled mixtures
Mixture database
Magnitude spectrogram
Separator network
Vocal estimates
Magnitude spectrogram
Voice discriminator
Real/fake?
Singing voice database
Magnitude spectrogram
Unlabeled vocals

Derivation of unsupervised loss

- For optimal separator: $q_\phi(s^k|m) = p(s^k|m)$

Derivation of unsupervised loss

- For optimal separator: $q_\phi(s^k|m) = p(s^k|m)$

$$\begin{array}{rcl} E_{m\sim p_{\text{data}}}\ q_\phi(s^k|m) & = & E_{m\sim p_{\text{data}}}\ p(s^k|m) \\ \text{Overall separator output} & = & \text{Source distribution} \end{array}$$

Derivation of unsupervised loss

- For optimal separator: $q_\phi(s^k|m) = p(s^k|m)$

$$\begin{aligned} E_{m\sim p_{\text{data}}}\ q_\phi(s^k|m) &= E_{m\sim p_{\text{data}}}\ p(s^k|m) \\ {}^{\text{out}}q_\phi^k &= p_s^k \end{aligned}$$

Derivation of unsupervised loss

- For optimal separator: $q_\phi(s^k|m) = p(s^k|m)$

$$\begin{aligned} E_{m\sim p_{\text{data}}}\ q_\phi(s^k|m) &= E_{m\sim p_{\text{data}}}\ p(s^k|m) \\ {}^{\text{out}}q_\phi^k &= p_{\text{s}}^k \end{aligned}$$

- Necessary condition for optimal separator
- Loss: Minimise divergence between source outputs: $L_{\text{u}} = \sum_{k=1}^{K} D[{}^{\text{out}}q_\phi^k || p_{\text{s}}^k]$

Overall approach

- Supervised loss: MSE between estimate and ground truth

Overall approach

- Supervised loss: MSE between estimate and ground truth
- Unsupervised loss:
  - $L_\mathrm{u} = \sum_{k=1}^{K} D[^{\mathrm{out}} q_{\phi}^{k} || p_{\mathrm{s}}^{k}]$
  - L_add: MSE between sum of source estimates and mixture

Overall approach

- Supervised loss: MSE between estimate and ground truth
- Unsupervised loss:
  - $L_{\mathrm{u}} = \sum_{k=1}^{K} D[{}^{\mathrm{out}}q_{\phi}^{k} || p_{\mathrm{s}}^{k}]$
  - L_{add}: MSE between sum of source estimates and mixture
- Total loss:
  $L = L_{\mathrm{s}} + \alpha L_{\mathrm{u}} + \beta L_{\mathrm{add}}$

Divergence minimization with GANs

- Discriminator estimates divergence D between generator and real distribution
- Generator minimises divergence D

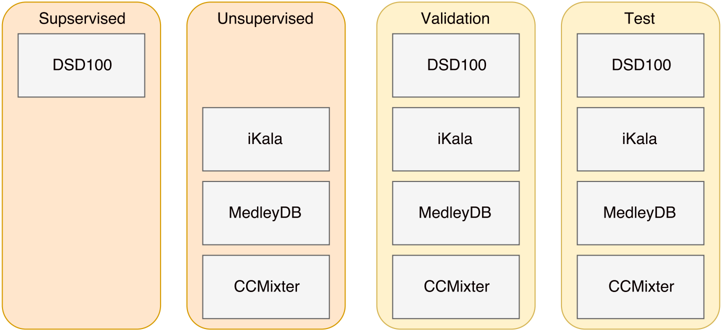
Divergence minimization with GANs

- Discriminator estimates divergence D between generator and real distribution
- Generator minimises divergence D

- **Our separator is a conditional generator**

$\Rightarrow$ We use one discriminator per source to estimate the Wasserstein distance $W[{}^{\text{out}}q_{\phi}^{k}||p_{s}^{k}]$

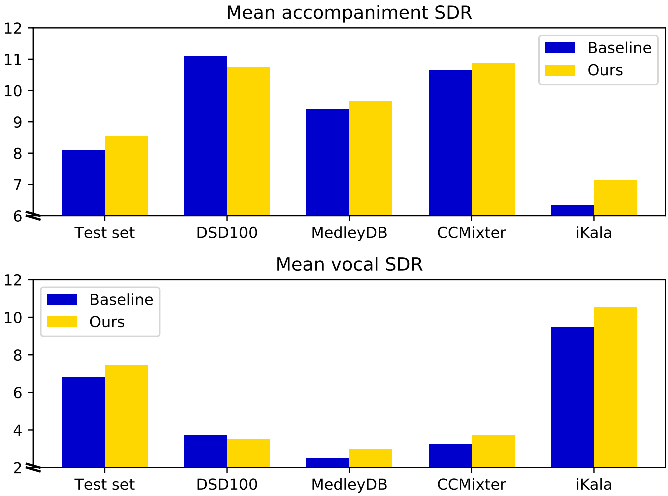
Experimental setup



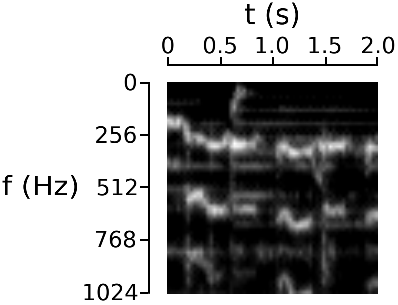
- Avoids dataset bias
- Supervised and semi-supervised training with early stopping
- U-Net as separator, DCGAN as discriminator

Results

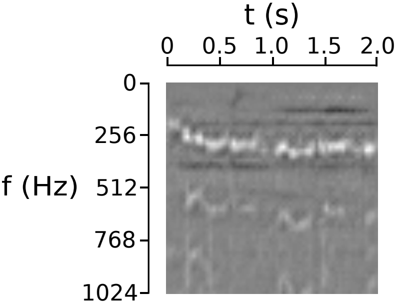
Performance



Results
Qualitative

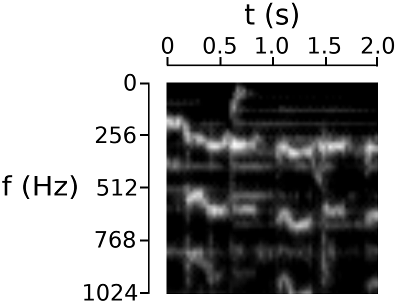


(a) Separator estimate x

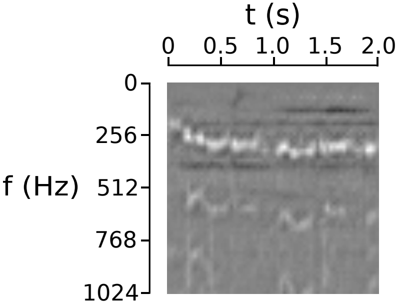


(b) $\nabla_x D(x)$

Results
Qualitative



(a) Separator estimate x



(b) $\nabla_x D(x)$

$\Rightarrow$ Discriminator appears to work

- More perceptual loss function?

Summary

- Current SotA methods only use multi-track data
- Our approach also uses solo source recordings
- Performance improvement in singing voice separation experiment
- More perceptual loss? (seeks posterior modes, not means)

End

Code available at
`https://github.com/f90/AdversarialAudioSeparation`

Thank you for your attention!

A. Jansson, E. J. Humphrey, N. Montecchio, R. Bittner, A. Kumar, and T. Weyde.
Singing voice separation with deep U-Net convolutional networks.
In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 323–332, 2017.

M. Miron, J. Janer Mestres, and E. Gómez Gutiérrez.
Generating data to train convolutional neural networks for classical music source separation.
In *Proceedings of the 14th Sound and Music Computing Conference*. Aalto University, 2017.

A. A. Nugraha, A. Liutkus, and E. Vincent.
Multichannel audio source separation with deep neural networks.
PhD thesis, Inria, 2015.

S. Uhlich, F. Giron, and Y. Mitsufuji.
Deep neural network based instrument extraction from music.
In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2135–2139. IEEE, 2015.

S. Uhlich, M. Porcu, F. Giron, M. Enenkl, T. Kemp, N. Takahashi, and Y. Mitsufuji.
Improving music source separation based on deep neural networks through data augmentation and network blending.
In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 261–265, March 2017.