Master Thesis Presentation
21 February 2023

Generative Adversarial Networks for the Generation of Microphone Array Data

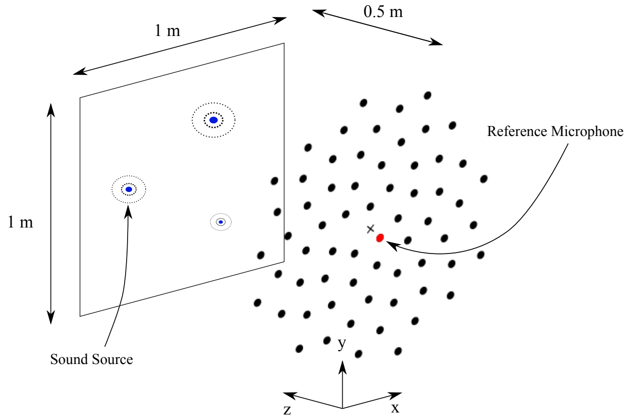
Gaspard Ulysse Fragnière

Agenda

TODO: remove this slide if not more insightful

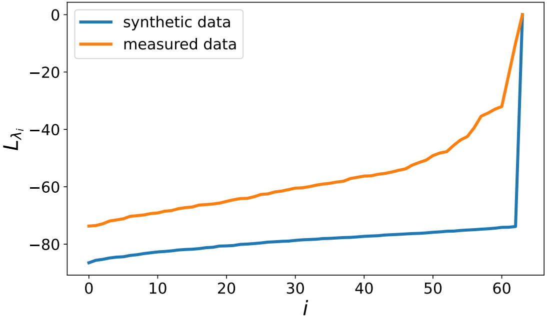
- Introduction
- Some Fundamentals
- Methods
- Results and Discussion
- Conclusion
- Future Works

Introduction: Context



- Microphones array are used for source characterization.
- Want to create DL-based algorithm to characterize sound source (in a supervised context).
- Source characterization algorithm uses the Cross Spectral Matrix for stationary signal.
- Cross Spectral Matrix: compact representation of the microphones' signals in the frequency domain

Introduction: Data



- Clear difference between measured and synthetic data
- Only limited amount of data for supervised learning (e.g. 50 millions sample required to train)
- Domain-shift

Introduction: Generating Data

- Generative Adversarial Networks (GAN): DL-based approach to generate data
- Able to learn complicated data structure.
- Could be suited to generate realistic CSMs.

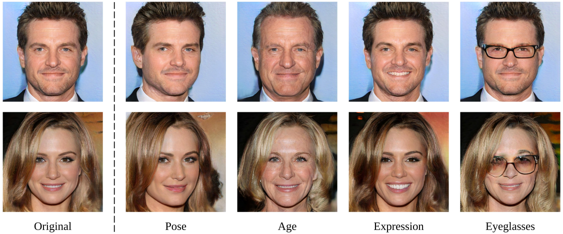


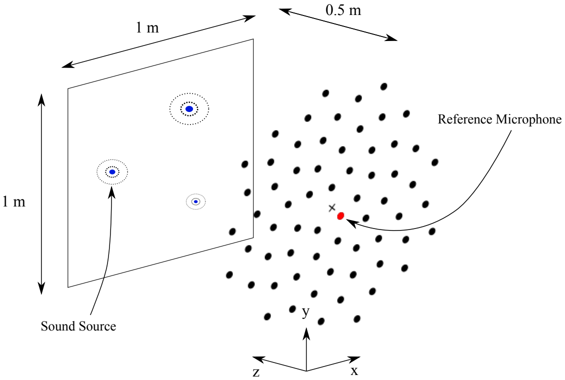
Figure 1: Faces generated with a GAN. From **TODO: add source**

Introduction: Aim

Hence the goal of this thesis is to investigate:

- if CSMs can be generated directly or indirectly, as realistically as possible, using a GAN approach.
- if training first GAN with synthetic data and fine-tune them with measurement is suited to deal with lack of data.
- Investigate data augmentation scheme to improve the realness of synthetic data.

Fundamentals: Sound Model



The sound model equation is given by:

$$\mathbf{p} = \mathbf{H}\mathbf{q} + \mathbf{n} \tag{1}$$

with

- $\mathbf{p} \in \mathbb{C}^{64}$: the pressure in the 64 microphones of the array
- $\mathbf{q} \in \mathbb{C}^{J}$: amplitudes of the J uncorrelated sources
- $\mathbf{H} \in \mathbb{C}^{(64,J)}$: transfer function from the sources to the sensors.
- $\mathbf{n}$: independent noise.

Fundamentals: CSM, Eigendecomposition and Rank I CSM

- The Cross Spectral Matrix (CSM) is a representation of the sound pressure in the different microphones. Using Welch's method, it can be approximated as:

$$\hat{\mathbf{C}} = \frac{1}{B}\sum_{b=1}^{B} \mathbf{p}\mathbf{p}^H \tag{2}$$

- A Cross spectral matrix $\hat{\mathbf{C}}$ of dimension $M \times M$ can be representated by its eigenvalues and eigenvectors using:

$$\hat{\mathbf{C}} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^H \tag{3}$$

- Using only one eigenvector $\mathbf{v}_i \in \mathbf{V}$ and the corresponding eigenvalue $\mathbf{\Lambda}_{ii}$, the Rank I CSM can be computed as:

$$\hat{\mathbf{C}}_i = \mathbf{v}_i \mathbf{\Lambda}_{ii} \mathbf{v}_i^H \tag{4}$$

Fundamentals: GAN and WGAN-GP

- GAN consists of two competing networks: generator and discriminator.
- Discriminator: determine real from fake samples.
- Generator: produce data realistic enough to fool the discriminator.
- Wasserstein GAN with Gradient Penalty (WGAN-GP): improved GAN, using the Wasserstein distance as loss function.

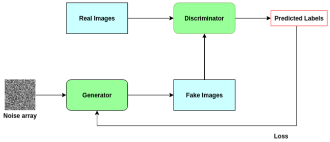


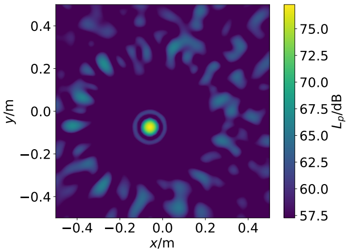
Figure 2: GAN Structure. From **TODO: add source**

Methods: Introduction

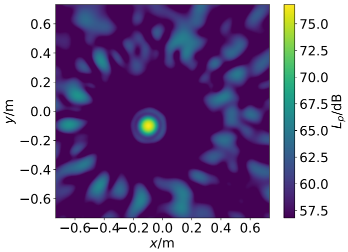
- This thesis investiguates the generation of CSMs through their eigendecomposition.
- CSMs are complex and hermitian matrices, hence have complicated distribution to learn.
- Eigendecomposition is an insightful representation of source:
  - Eigenvalues: represent the sources' strength
  - Eigenvectors: represent the sources' position

Methods: Data

- Synthetic data follows a complex Wishart distribution
- Measurements performed with single loudspeaker in an anechoic chamber
- Synthetic and measurement have same position and Helmotz number $He = 16$



(a) Synthetic



(b) Measurement

Methods: Generating Eigenvalues

The first approach consisted in generating the scaled eigenvalues $[\lambda_0, \ldots, \lambda_{63}] \in]0,1]^{64}$, with $\lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_{63}$. This approach allowed to scale generated eigenavalues, before feeding them to the discriminator, in order to improve performances.

Layer	Output Shape	Number of Parameters
InputLayer	128	0
Dense	256	32768
LeakyReLU	256	0
BatchNormalization	256	1024
Dense	512	131584
LeakyReLU	512	0
Dense	1024	525312
LeakyReLU	1024	0
Dense	64	65600

Table 1: Generator

Layer	Output Shape	Number of Parameters
InputLayer	64	0
Dense	512	33280
LeakyReLU	512	0
Dense	256	131328
LeakyReLU	256	0
Dense	1	257

Table 2: Discriminator

Methods: Generating Eigenvalues

The second approach to generate the eigenvalues consisted in generating their level representation $L_{\lambda_0}, \ldots, L_{\lambda_{63}}$ defined as:

$$L_{\lambda_i} = 10 \log_{10}\left(\frac{\lambda_i}{\lambda_{63}}\right) \tag{5}$$

Layer	Output Shape	Number of Parameters
InputLayer	128	0
Dense	256	32768
LeakyReLU	256	0
BatchNormalization	256	1024
Dense	512	131584
LeakyReLU	512	0
Dense	1024	525312
LeakyReLU	1024	0
Dense	64	65600
ReLU	64	0
Multiply	64	0

Table 3: Generator

Layer	Output Shape	Number of Parameters
InputLayer	64	0
Multiply	64	0
Dense	512	33280
LeakyReLU	512	0
Dense	256	131328
LeakyReLU	256	0
Dense	1	257

Table 4: Discriminator

Methods: Generating Eigenvectors

- As a proof of concept, it was decided to start by only generating the strongest eigenvector, defined as main eigenvector.
- The main eigenvector contains all the information about the position of the source.

Layer	Output Shape	Number of Parameters
InputLayer	128	0
Dense	256	32768
LeakyReLU	256	0
BatchNormalization	256	1024
Dense	512	131584
LeakyReLU	512	0
Dense	1024	525312
LeakyReLU	1024	0
Dense	128	131200
Reshape	(1, 64, 2)	0

Table 5: Generator

Layer	Output Shape	Number of Parameters
InputLayer	(1, 64, 2)	0
Flatten	128	0
Dense	512	66048
LeakyReLU	512	0
Dense	256	131328
LeakyReLU	256	0
Dense	1	257

Table 6: Discriminator

Methods: Data Augmentation

For a received synthetic CSM $\hat{\mathbf{C}}$, its eigendecomposition $\hat{\mathbf{C}} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^H$ is computed. The CSM $\hat{\mathbf{C}}$ can then be modified with

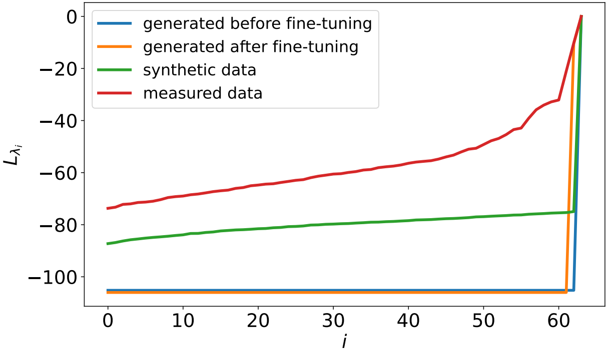
- generated eigenvalues $\hat{\mathbf{\Lambda}}$, with:

$$\hat{\mathbf{C}}^{\mathbf{\Lambda}}_{\text{augm.}} = \mathbf{V}\hat{\mathbf{\Lambda}}\mathbf{V}^H \tag{6}$$

- a generated main eigenvector $\hat{\mathbf{v}}_M$ and corresponding semi-generated eigenvectors matrix $\hat{\mathbf{V}} = [\mathbf{v}_1^T, \ldots, \hat{\mathbf{v}}_M^T]$, with:

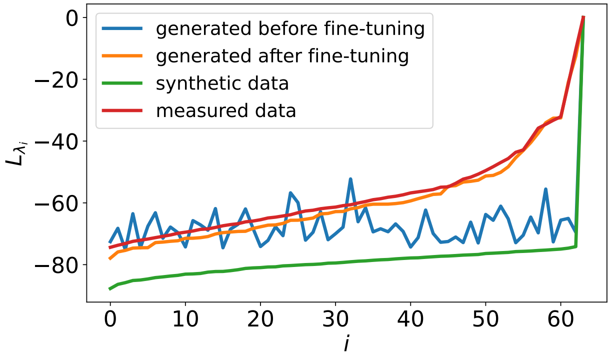
$$\hat{\mathbf{C}}^{\mathbf{V}}_{\text{augm.}} = \hat{\mathbf{V}}\mathbf{\Lambda}\hat{\mathbf{V}}^H \tag{7}$$

Results and Discussion: Generating eigenvalues from scaled values



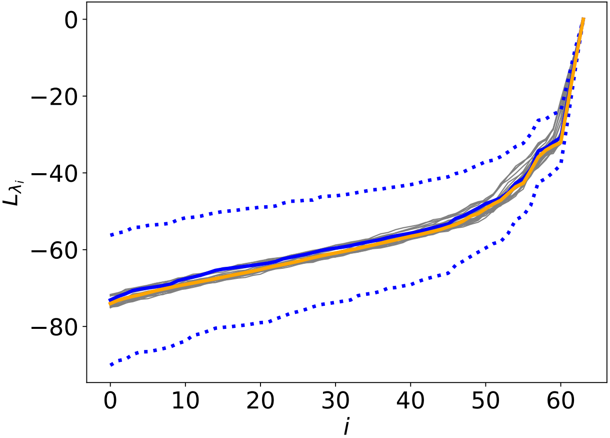
- Sudden and steep drop in value, reaching a level around 10^{-100}, both before and after fine-tuning.
- the WGAN-GP cannot capture very well small numerical variation.
- This approach is not suited.

Results and Discussion: Generating eigenvalues from levels



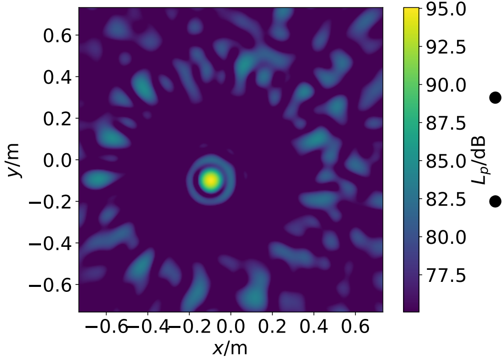
- The generated data looks more like the real data, especially after fine-tuning.
- Generating the eigenvalues from their levels produces satisfying results.

Results and Discussion: Generating eigenvalues from levels 2



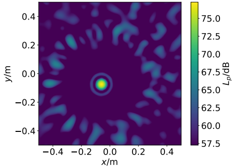
- Training the network first with synthetic data and then fine-tuning it with a single measurement allows to produce eigenvalues with a large variation.
- This approach allows to generate eigenvalues samples that are representative not only of a single measurement, but of multiple ones.

Results and Discussion: Generating strongest eigenvector

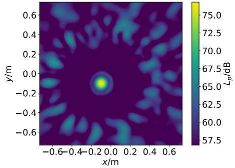


- The beamforming map computed from a Rank I CSM with generated main eigenvector is sufficiently realistic sample.
- It was observed that all the generated main eigenvector were similar.
- the WGAN-GP cannot produce a great variety of sample, but it is representative of the data it was trained with.

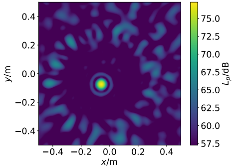
Results and Discussion: Data Augmentation



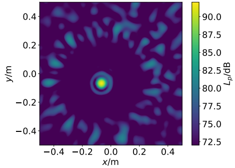
(a) Synthetic



(b) Measurement



(c) Augmented with Eigenvalues



(d) Augmented with Eigenvectors

Conclusion

- WGAN-GPs are suited to generate eigenvalues and strongest eigenvector.
- The data augmentation methods introduced allow to improve how realistic synthetic CSM are.
- The eigedecomposition is a good representation of CSM to learn their distribution.
- The finding in this thesis are an important step toward solving the unavailability of real training data for source localization or characterization.

Future Works

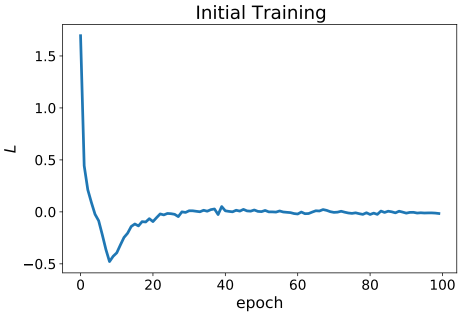
- Extending the work to be able to generate CSM for different positions.
- Investigate whether a Neural Network could be designed for generating eigenvectors corresponding to a given source position not observed in the dataset.
- Investigate how to generate the remaining weakest eigenvectors.

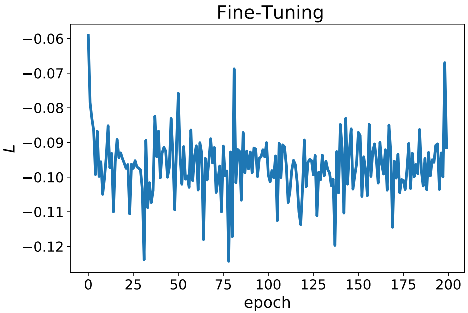
Questions?

Bibliography

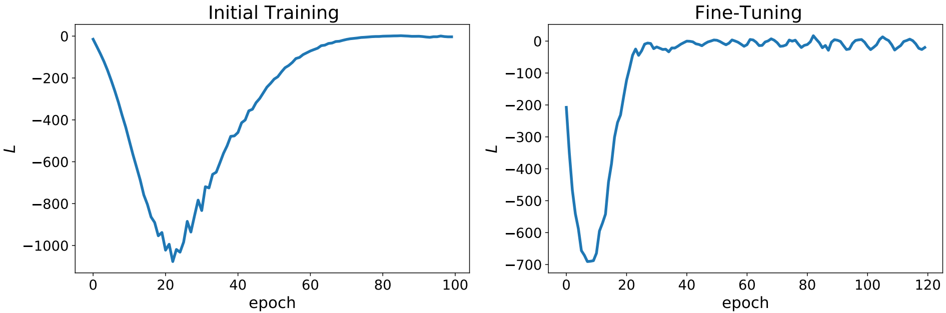
References

Appendix: loss function WGAN-GP to generate eigenvalues from scaled values





Appendix: loss function WGAN-GP to generate eigenvalues from level



Appendix: loss function WGAN-GP to generate main eigenvector