# Multi-Microphone Speaker Localization and Tracking on Manifolds

Sharon Gannot
joint work with Bracha Laufer-Goldshtein and Ronen Talmon

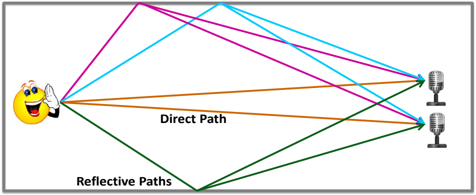
Faculty of Engineering and Data Science Institute, Bar-Ilan University, Israel

ITG Conference on Speech Communication, Oldenburg, Germany, October 10th, 2018

# Acoustic Source Localization

**Goal**

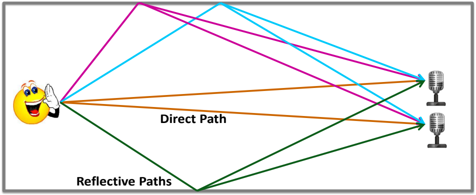
Locate a sound source given measurements of the sound field



# Acoustic Source Localization

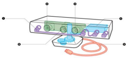
## Goal

Locate a sound source given measurements of the sound field




## Applications

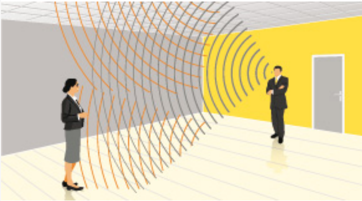
1. An essential component in speech enhancement algorithms
2. Camera steering
3. Teleconferencing
4. Robot audition
5. Surveillance
6. Smart home/clinic/car

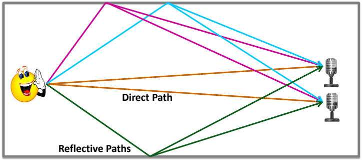


# Room Acoustics Essentials

## Acoustic propagation models

- When sound propagates in an enclosure it undergoes reflections from the room surfaces
- Reflections can be modeled as images beyond room walls and hence impinging the microphones from many directions [Allen and Berkley, 1979, Peterson, 1986]
- Statistical models for late reflections [Polack, 1993, Schroeder, 1996, Jot et al., 1997]
- Late reflections tend to be diffused, hence do not exhibit directionality [Dal-Degan and Prati, 1988, Habets and Gannot, 2007]





# Speech Processing in Acoustic Environments

- Classical multi-microphone speech processing algorithms use time difference of arrival (TDOA)-only model
- Viable speech processing solutions can only be accomplished by an accurate source propagation description, captured by the acoustic impulse response (AIR)
- Describing the wave propagation of any audio source in an arbitrary acoustic environment is, however, a cumbersome task, since:
  - No simple mathematical models exist
  - The estimation of the vast number of parameters used to describe the wave propagation suffers from large errors

## Data-driven approach

To alleviate these limitations and to infer a mathematical model that is accurate, simple to describe and simple to implement, we propose a data-driven approach

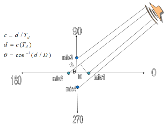
# How to Model the Acoustic Environment?

How to model the acoustic path?

**Classical speech processing**

TDOA-only

Too simplified



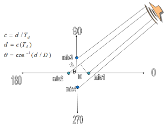
# How to Model the Acoustic Environment?

**How to model the acoustic path?**

**Classical speech processing**

TDOA-only

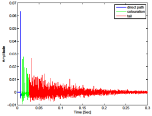
Too simplified




**To cope with reverberations**

Entire Impulse Response

Perfect but Complicated: No simple Model



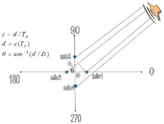
# How to Model the Acoustic Environment?

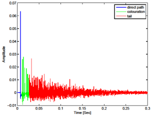
How to model the acoustic path?

**Classical speech processing** → TDOA-only → Too simplified

→ Manifold Learning → Simple to describe and to implement

**To cope with reverberations** → Entire Impulse Response → Perfect but Complicated: No simple Model

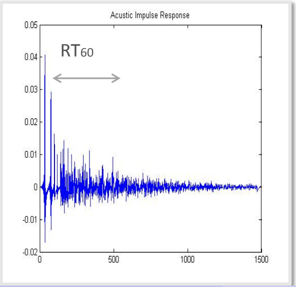


# Main Claims

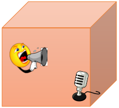
- The acoustic response can serve as a fingerprint for source localization
- The variability of the acoustic response in specific enclosures depends only on a small number of parameters
- The intrinsic degrees of freedom in acoustic responses are limited to a small number of variables

$\Rightarrow$ manifold learning approaches may improve localization ability




**Controlling Parameters**

- room dimensions
- reverberation time
- microphone position
- source position
- ...



# Outline

# Outline

# Data Model: The Two Microphone Case

## Microphone signals:

The measured signals in the two microphones:

$$y_1(n) = a_1(n) * s(n) + u_1(n)$$
$$y_2(n) = a_2(n) * s(n) + u_2(n)$$

- $s(n)$ - the source signal
- $a_i(n),\ i = \{1, 2\}$ - the acoustic impulse responses relating the source and each of the microphones
- $u_i(n),\ i = \{1, 2\}$ - noise signals, independent of the source

# Data Model: The Two Microphone Case

## Microphone signals:

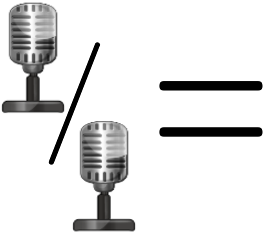
The measured signals in the two microphones:

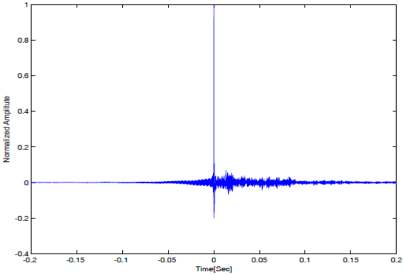
$$y_1(n) = a_1(n) * s(n) + u_1(n)$$
$$y_2(n) = a_2(n) * s(n) + u_2(n)$$

- $s(n)$ - the source signal
- $a_i(n),\ i = \{1, 2\}$ - the acoustic impulse responses relating the source and each of the microphones
- $u_i(n),\ i = \{1, 2\}$ - noise signals, independent of the source

Find a feature vector representing the characteristics of the acoustic path (a fingerprint) and independent of the source signal!

# Relative Transfer Function (RTF) [Gannot et al., 2001]






## RTF:

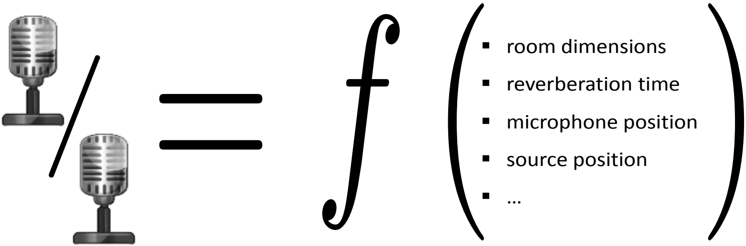
- Defined as the ratio between the transfer functions relating the source and the two mics:

$$H_{12}(k) = \frac{A_2(k)}{A_1(k)}$$

- In the time domain: the relative impulse response (RIR) satisfies:

$$a_2(n) = h_{12}(n) * a_1(n)$$

# Relative Transfer Function (RTF) [Gannot et al., 2001]




## RTF:

- Represents the acoustic paths and is independent of the source signal
- Generalizes the TDOA
- Depends on a small set of parameters related to the physical characteristics of the environment

# Relative Transfer Function (RTF) [Gannot et al., 2001]

$= f\Big($

- room dimensions
- reverberation time
- microphone position
- **source position**
- ...

$\Big)$

## RTF:

- Represents the acoustic paths and is independent of the source signal
- Generalizes the TDOA
- Depends on a small set of parameters related to the physical characteristics of the environment
- In a static environment the source position is the only varying degree of freedom

# Relative Transfer Function (RTF)

## RTF-based feature vector:

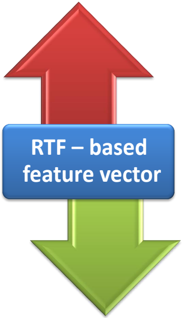
- Estimated based on PSD and cross-PSD
  (alternatively [Markovich-Golan and Gannot, 2015, Koldovsky et al., 2014]):

$$\hat{H}_{12}(k) = \frac{\hat{S}_{y_2 y_1}(k)}{\hat{S}_{y_1 y_1}(k)} \simeq \frac{A_2(k)}{A_1(k)}$$

- Define the feature vector:

$$\mathbf{h} = \left[\hat{H}_{12}(k_1), \ldots, \hat{H}_{12}(k_D)\right]^T$$

- $D \propto$ length of the relative impulse response (time domain)



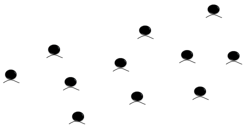

**High dimensional representation - due to reverberation**

**Controlled by one dominant factor - source position**

# Outline

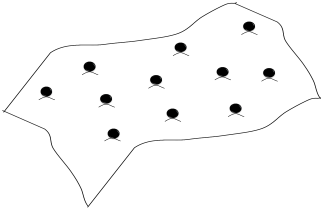
# How to Measure the Affinity between RTFs? [Laufer-Goldshtein et al., 2015, Laufer-Goldshtein et al., 2016b]

- The RTFs are represented as points in a high dimensional space
- Small Euclidean distance of high dimensional vectors implies proximity
- Large Euclidean distance of high dimensional vectors is meaningless



# How to Measure the Affinity between RTFs? [Laufer-Goldshtein et al., 2015, Laufer-Goldshtein et al., 2016b]

- The RTFs are represented as points in a high dimensional space
- Small Euclidean distance of high dimensional vectors implies proximity
- Large Euclidean distance of high dimensional vectors is meaningless

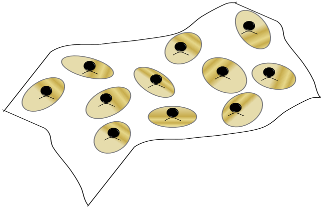


## Acoustic manifold

- They lie on a low dimensional nonlinear manifold $\mathcal{M}$

# How to Measure the Affinity between RTFs? [Laufer-Goldshtein et al., 2015, Laufer-Goldshtein et al., 2016b]

- The RTFs are represented as points in a high dimensional space
- Small Euclidean distance of high dimensional vectors implies proximity
- Large Euclidean distance of high dimensional vectors is meaningless

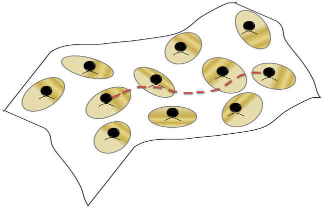


## Acoustic manifold

- They lie on a low dimensional nonlinear manifold $\mathcal{M}$
- Linearity is preserved in small neighbourhoods

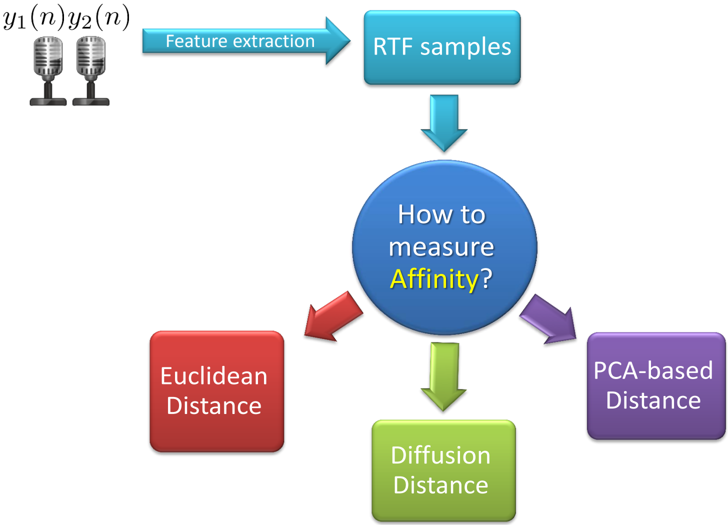
# How to Measure the Affinity between RTFs? [Laufer-Goldshtein et al., 2015, Laufer-Goldshtein et al., 2016b]

- The RTFs are represented as points in a high dimensional space
- Small Euclidean distance of high dimensional vectors implies proximity
- Large Euclidean distance of high dimensional vectors is meaningless



## Acoustic manifold

- They lie on a low dimensional nonlinear manifold $\mathcal{M}$
- Linearity is preserved in small neighbourhoods
- Distances between RTFs should be measured along the manifold



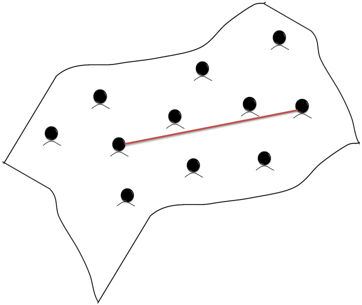

Each distance measure relies on a different hidden assumption about the underlying structure of the RTF samples

# Euclidean Distance

## The Euclidean distance between RTFs

$$D_{\mathrm{Euc}}(\mathbf{h}_i, \mathbf{h}_j) = \|\mathbf{h}_i - \mathbf{h}_j\|$$

- Compares two RTFs in their original space
- Does not assume an existence of a manifold
- Respects flat manifolds



A good affinity measure only when the RTFs are uniformly scattered all over the space, or when they lie on a flat manifold

# PCA-Based Distance

## PCA algorithm

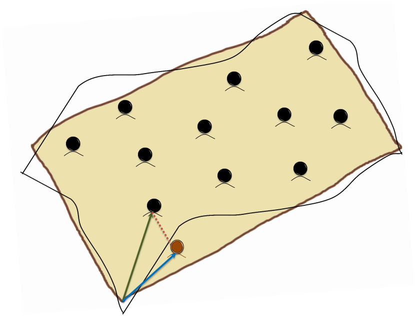
- The principal components - the $d$ dominant eigenvectors $\{\mathbf{v}_i\}_{i=1}^{d}$ of the covariance matrix of the data
- The RTFs are linearly projected onto the principal components:

$$\boldsymbol{\nu}(\mathbf{h}_i) = [\mathbf{v}_1, \ldots \mathbf{v}_d]^T (\mathbf{h}_i - \boldsymbol{\mu})$$

## PCA-based distance between RTFs

$$D_{\mathrm{PCA}}(\mathbf{h}_i, \mathbf{h}_j) = \|\boldsymbol{\nu}(\mathbf{h}_i) - \boldsymbol{\nu}(\mathbf{h}_j)\|$$

- A global approach - extracts principal directions of the entire set
- Linear projections - the manifold is assumed to be linear/flat



# PCA-Based Distance

## PCA algorithm

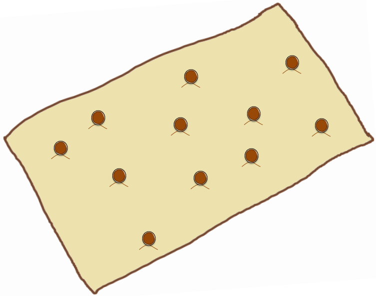
- The principal components - the $d$ dominant eigenvectors $\{\mathbf{v}_i\}_{i=1}^{d}$ of the covariance matrix of the data
- The RTFs are linearly projected onto the principal components:

$$\boldsymbol{\nu}\left(\mathbf{h}_i\right) = \left[\mathbf{v}_1, \ldots \mathbf{v}_d\right]^T \left(\mathbf{h}_i - \boldsymbol{\mu}\right)$$

## PCA-based distance between RTFs

$$D_{\mathrm{PCA}}(\mathbf{h}_i, \mathbf{h}_j) = \|\boldsymbol{\nu}(\mathbf{h}_i) - \boldsymbol{\nu}(\mathbf{h}_j)\|$$

- A global approach - extracts principal directions of the entire set
- Linear projections - the manifold is assumed to be linear/flat



# PCA-Based Distance

## PCA algorithm

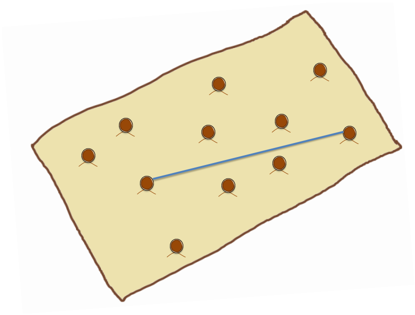
- The principal components - the $d$ dominant eigenvectors $\{\mathbf{v}_i\}_{i=1}^{d}$ of the covariance matrix of the data
- The RTFs are linearly projected onto the principal components:

$$\boldsymbol{\nu}(\mathbf{h}_i) = [\mathbf{v}_1, \ldots \mathbf{v}_d]^T (\mathbf{h}_i - \boldsymbol{\mu})$$

## PCA-based distance between RTFs

$$D_{\mathrm{PCA}}(\mathbf{h}_i, \mathbf{h}_j) = \|\boldsymbol{\nu}(\mathbf{h}_i) - \boldsymbol{\nu}(\mathbf{h}_j)\|$$

- A global approach - extracts principal directions of the entire set
- Linear projections - the manifold is assumed to be linear/flat

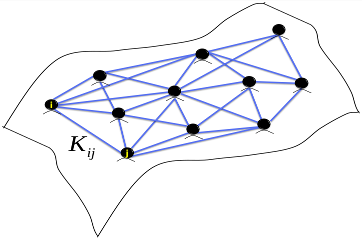


# Diffusion Maps

## Discretization of the manifold

- The manifold can be empirically represented by a graph:
  - The RTF samples are the graph nodes
  - The weights of the edges are defined using a kernel function:

$$K_{ij} = k(\mathbf{h}_i, \mathbf{h}_j) = \exp\left\{-\frac{\|\mathbf{h}_i - \mathbf{h}_j\|^2}{\varepsilon}\right\}$$

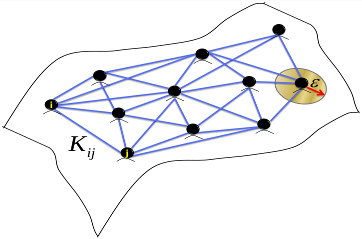


# Diffusion Maps

## Discretization of the manifold

- The manifold can be empirically represented by a graph:
  - The RTF samples are the graph nodes
  - The weights of the edges are defined using a kernel function:

$$K_{ij} = k(\mathbf{h}_i, \mathbf{h}_j) = \exp\left\{-\frac{\|\mathbf{h}_i - \mathbf{h}_j\|^2}{\varepsilon}\right\}$$



# Diffusion Maps

## Discretization of the manifold

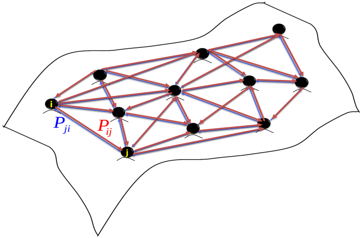
- The manifold can be empirically represented by a graph:
  - The RTF samples are the graph nodes
  - The weights of the edges are defined using a kernel function:

$$K_{ij} = k(\mathbf{h}_i, \mathbf{h}_j) = \exp\left\{-\frac{\|\mathbf{h}_i - \mathbf{h}_j\|^2}{\varepsilon}\right\}$$

- Define a Markov process on the graph by the transition matrix:

$$p(\mathbf{h}_i, \mathbf{h}_j) = P_{ij} = K_{ij} / \sum_{r=1}^{N} K_{ir}$$

which is a discretization of a diffusion process on the manifold

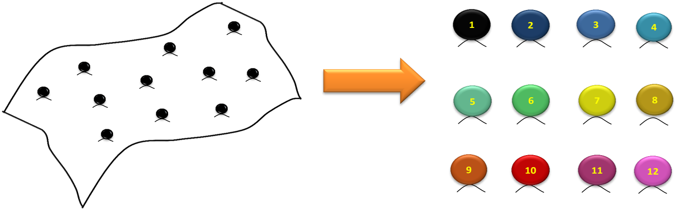


# Diffusion Maps

## Diffusion mapping [Coifman and Lafon, 2006]

- Apply eigenvalue decomposition (EVD) to the matrix $P$ and obtain the eigenvalues $\{\lambda_j\}$ and right eigenvectors $\{\varphi_j\}$.
- A nonlinear mapping into a new low-dimensional Euclidean space:

$$\mathbf{\Phi}_d : \mathbf{h}_i \mapsto \left[\lambda_1 \varphi_1^{(i)}, \ldots, \lambda_d \varphi_d^{(i)}\right]^T$$




The mapping provides a parametrization of the manifold and represents the latent variables - here, the position of the source

# Diffusion Distance

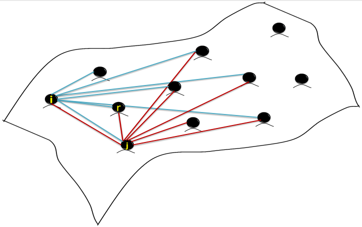
## Diffusion distance between RTFs

The distance along the manifold is approximated by the diffusion distance:

$$D^2_{\mathrm{Diff}}(\mathbf{h}_i, \mathbf{h}_j) = \sum_{r=1}^{N} \left(p\left(\mathbf{h}_i, \mathbf{h}_r\right) - p\left(\mathbf{h}_j, \mathbf{h}_r\right)\right)^2 / \phi_0^{(r)}$$

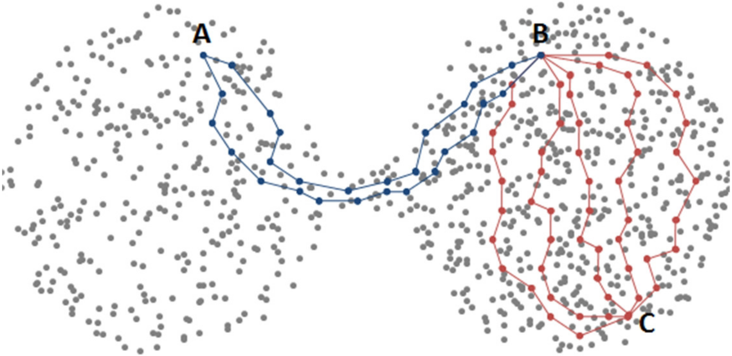
- Two points are close if they are highly connected in the graph
- The diffusion distance can be well approximated by the Euclidian distance in the embedded domain:

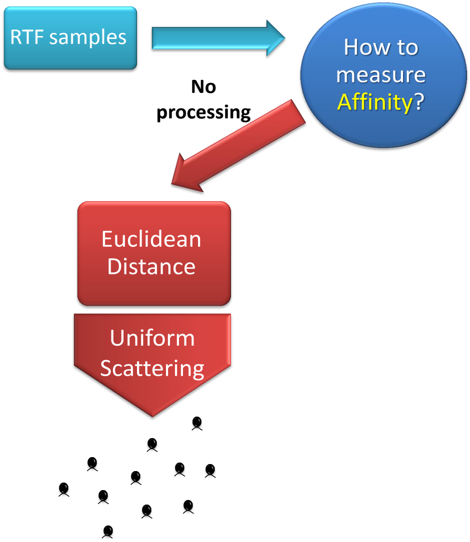
$$D_{\mathrm{Diff}}(\mathbf{h}_i, \mathbf{h}_j) \cong \|\mathbf{\Phi}_d(\mathbf{h}_i) - \mathbf{\Phi}_d(\mathbf{h}_j)\|$$

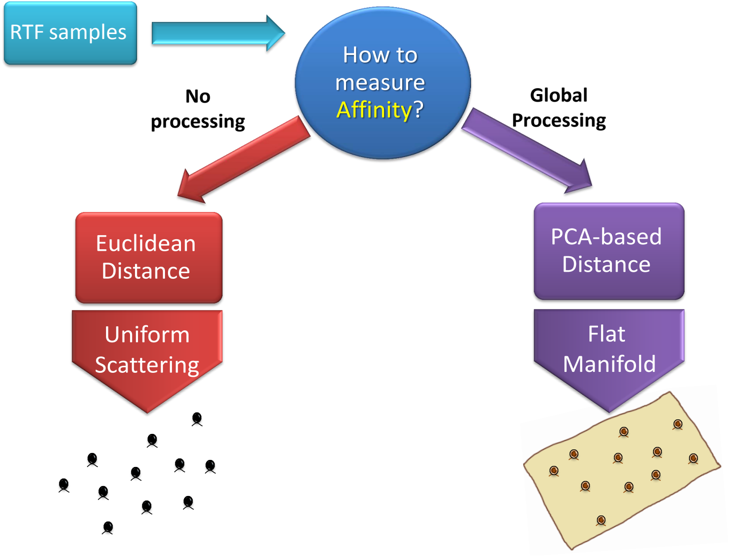


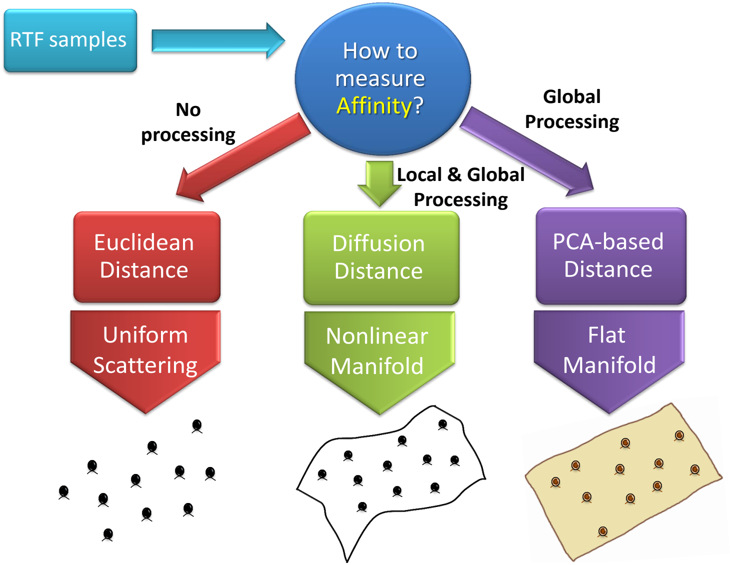
# Diffusion Distance

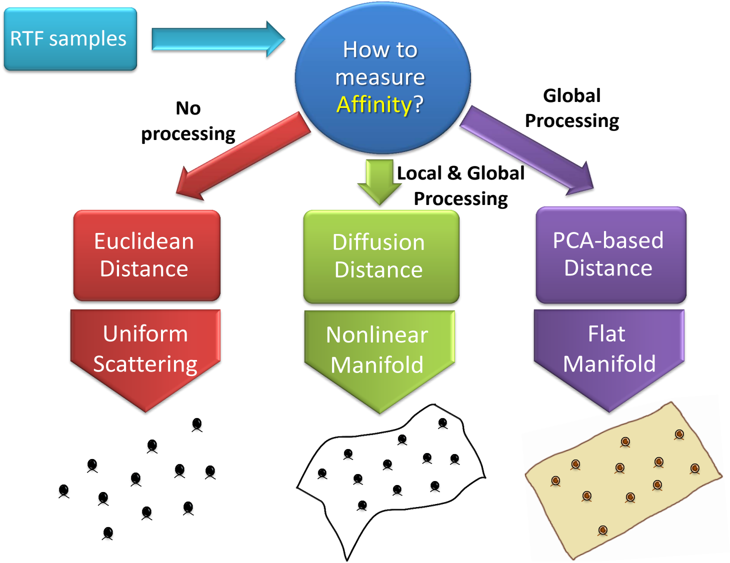
## Illustration




RTF samples
No processing
How to measure Affinity?
Euclidean Distance
Uniform Scattering


RTF samples
How to measure Affinity?
No processing
Global Processing
Euclidean Distance
Uniform Scattering
PCA-based Distance
Flat Manifold


RTF samples
How to measure Affinity?
No processing
Local & Global Processing
Global Processing
Euclidean Distance
Diffusion Distance
PCA-based Distance
Uniform Scattering
Nonlinear Manifold
Flat Manifold



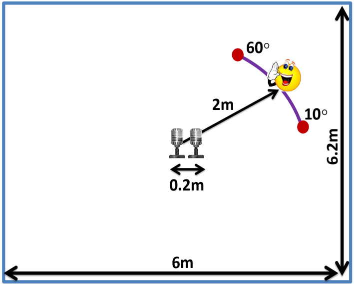

Which of the distance measures is proper?
What is the true underlying structure of the RTFs?

# Simulation Results

## Room setup

Simulate a reverberant room using the image method [Allen and Berkley, 1979]:

- Room dimension $6 \times 6.2 \times 3$m
- Microphones at: $[3, 3, 1]$ and $[3.2, 3, 1]$
- The source is positioned at 2m from the mics, the azimuth angle in $10^\circ \div 60^\circ$.
- $T_{60} = 150/300/500$ ms
- SNR$=$ 20 dB

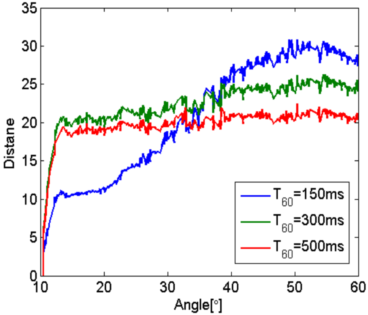



## Test

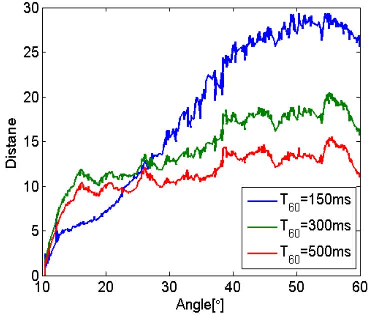
Measure the distance between each of the RTFs and the RTF corresponding to $10^\circ$:

- If monotonic with respect to the angle - proper distance
- If not monotonic with respect to the angle - improper distance

# Euclidean Distance & PCA-based Distance [Laufer-Goldshtein et al., 2015]




(a) Euclidean Distance

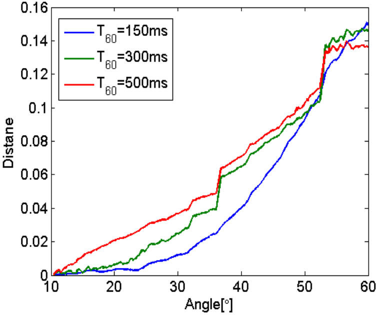



(b) PCA-based Distance

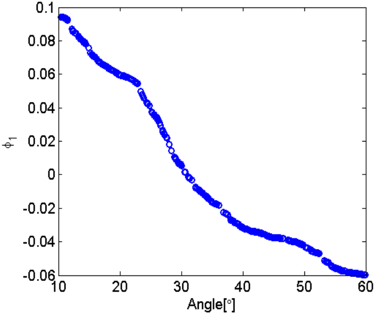
**For both distance measures:**

- Monotonic with respect to the angle only in a limited region
- This region becomes smaller as the reverberation time increases
- They are inappropriate for measuring angles' proximity

# Diffusion Maps




(c) Diffusion Distance




(d) Diffusion Mapping

## The diffusion distance:

- Monotonic with respect to the angle for almost the entire range
- It is an appropriate distance measure in terms of the source DOA
- Mapping corresponds well with angles - recovers the latent parameter

# Outline

# Semi-Supervised Learning

Mixed of supervised (attached with known locations as anchors) and unsupervised (unknown locations) learning

# Semi-Supervised Learning

Mixed of supervised (attached with known locations as anchors) and unsupervised (unknown locations) learning

## Why using unlabeled data?

1. Localization - training should fit the specific environment of interest
   - Cannot generate a general database for all possible acoustic scenarios
   - Generating a large amount of labelled data is cumbersome/impractical
   - Unlabelled data is freely available - whenever someone is speaking

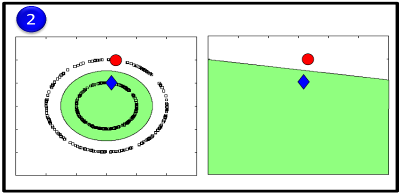
1

# Semi-Supervised Learning

Mixed of supervised (attached with known locations as anchors) and unsupervised (unknown locations) learning

## Why using unlabeled data?

1. Localization - training should fit the specific environment of interest
   - Cannot generate a general database for all possible acoustic scenarios
   - Generating a large amount of labelled data is cumbersome/impractical
   - Unlabelled data is freely available - whenever someone is speaking
2. Unlabelled data can be utilized to recover the manifold structure

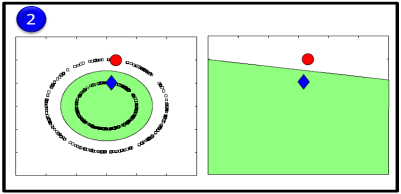



# Semi-Supervised Learning

Mixed of supervised (attached with known locations as anchors) and unsupervised (unknown locations) learning

## Why using unlabeled data?

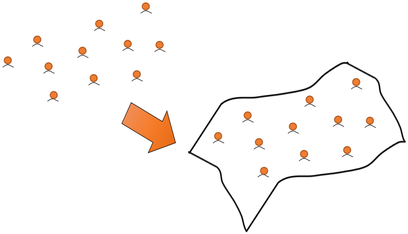
1. Localization - training should fit the specific environment of interest
   - Cannot generate a general database for all possible acoustic scenarios
   - Generating a large amount of labelled data is cumbersome/impractical
   - Unlabelled data is freely available - whenever someone is speaking
2. Unlabelled data can be utilized to recover the manifold structure
3. Semi-supervised learning is the natural setting for human learning


# Semi-Supervised Learning

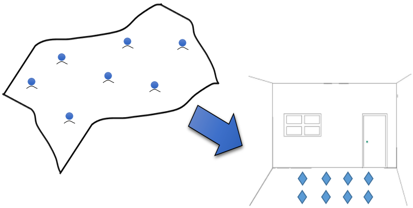
**Unlabelled Samples**

Recover the Manifold Structure

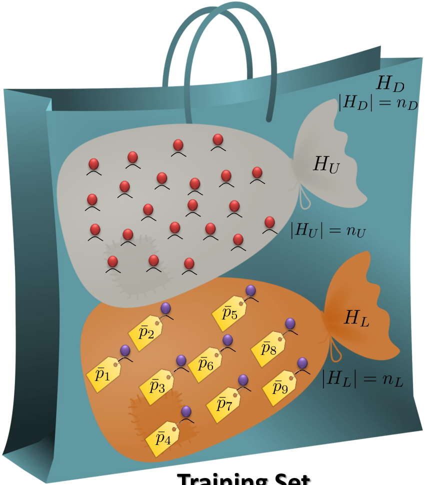


**Labelled Samples**

Anchor Points – Translate RTFs to Positions



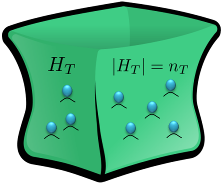
# Semi-Supervised Learning




**Training Set**

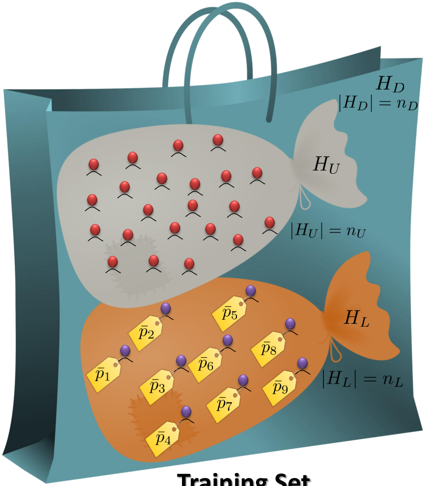
## Data:

- $H_L = \{\mathbf{h}_i\}_{i=1}^{n_L}$ - $n_L$ labelled samples
- $P_L = \{\bar{p}_i\}_{i=1}^{n_L}$ - labels/positions
- $H_U = \{\mathbf{h}_i\}_{i=n_L+1}^{n_D}$ - $n_U$ unlabelled samples
- $H_D = H_L \cup H_U$ - entire training set
- $H_T = \{\mathbf{h}_i\}_{i=n_D+1}^{n}$ - $n_T$ test samples




**Test Set**

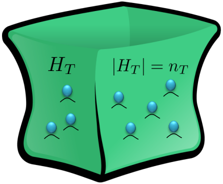
# Semi-Supervised Learning




**Training Set**

## Data:

- $H_L = \{\mathbf{h}_i\}_{i=1}^{n_L}$ - $n_L$ labelled samples
- $P_L = \{\bar{p}_i\}_{i=1}^{n_L}$ - labels/positions
- $H_U = \{\mathbf{h}_i\}_{i=n_L+1}^{n_D}$ - $n_U$ unlabelled samples
- $H_D = H_L \cup H_U$ - entire training set
- $H_T = \{\mathbf{h}_i\}_{i=n_D+1}^{n}$ - $n_T$ test samples




**Test Set**

Goal: Recover a (component-wise) function $p = f(\mathbf{h})$ which transforms an RTF to position

# Optimization and Manifold Regularization

Optimization in a reproducing kernel Hilbert space (RKHS) [Belkin et al., 2006]:

$$f^* = \underset{f \in \mathcal{H}_k}{\operatorname{argmin}} \frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2 + \gamma_k \|f\|^2_{\mathcal{H}_k} + \gamma_M \|f\|^2_{\mathcal{M}}$$

# Optimization and Manifold Regularization

Optimization in a reproducing kernel Hilbert space (RKHS) [Belkin et al., 2006]:

$$f^* = \underset{f \in \mathcal{H}_k}{\operatorname{argmin}} \frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2 + \gamma_k \|f\|_{\mathcal{H}_k}^2 + \gamma_M \|f\|_{\mathcal{M}}^2$$

**Cost function**

$\frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2$

**correspondence between function values and labels**

# Optimization and Manifold Regularization

Optimization in a reproducing kernel Hilbert space (RKHS) [Belkin et al., 2006]:

$$f^* = \underset{f \in \mathcal{H}_k}{\operatorname{argmin}} \frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2 + \gamma_k \|f\|^2_{\mathcal{H}_k} + \gamma_M \|f\|^2_{\mathcal{M}}$$

**Cost function**

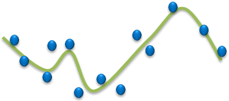
$\frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2$

**Tikhonov Regularization**

$\|f\|^2_{\mathcal{H}_k}$

**correspondence between function values and labels**

**smoothness condition in the RKHS**



# Optimization and Manifold Regularization

Optimization in a reproducing kernel Hilbert space (RKHS) [Belkin et al., 2006]:

$$f^* = \underset{f \in \mathcal{H}_k}{\operatorname{argmin}} \frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2 + \gamma_k \|f\|^2_{\mathcal{H}_k} + \gamma_M \|f\|^2_{\mathcal{M}}$$

**Cost function**

$\frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2$

**correspondence between function values and labels**

**Tikhonov Regularization**

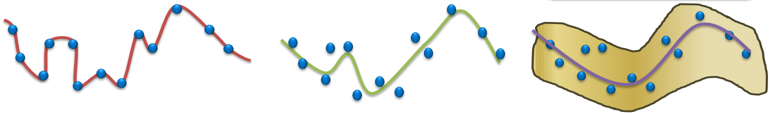
$\|f\|^2_{\mathcal{H}_k}$

**smoothness condition in the RKHS**

**Manifold Regularization**

$\|f\|^2_{\mathcal{M}}$

**smoothness penalty with respect to the manifold**



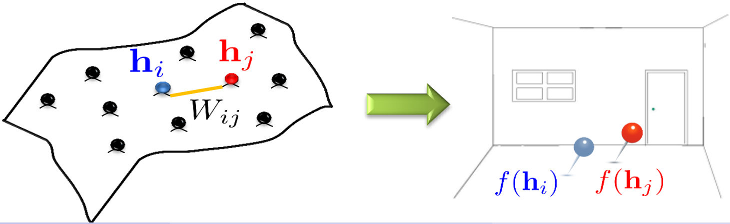
# Manifold Regularization

## Discretization of the manifold

- The manifold is empirically represented by a graph $G$ with weights:

$$W_{ij} = \begin{cases} \exp\left\{-\frac{\|\mathbf{h}_i - \mathbf{h}_j\|^2}{\varepsilon_w}\right\} & \text{if } \mathbf{h}_j \in \mathcal{N}_i \text{ or } \mathbf{h}_i \in \mathcal{N}_j \\ 0 & \text{otherwise} \end{cases}$$

  where $\mathcal{N}_j$ is a set consisting of the $d$ nearest-neighbours of $\mathbf{h}_j$
- The graph Laplacian of $G$: $\mathbf{M} = \mathbf{S} - \mathbf{W}$, with $S_{ii} = \sum_{j=1}^{n_D} \mathbf{W}_{ij}$
- Regularization: $\|f\|_{\mathcal{M}}^2 = \mathbf{f}_D^T \mathbf{M} \mathbf{f}_D = \frac{1}{2} \sum_{i,j=1}^{n_D} W_{ij} \left(f(\mathbf{h}_i) - f(\mathbf{h}_j)\right)^2$ with $\mathbf{f}_D^T = [f_1, f_2, \ldots, f_{n_D}]$ ▸ Proof



# Optimization and Manifold Regularization

The optimization problem can be recast as:

$$f^* = \underset{f \in \mathcal{H}_k}{\operatorname{argmin}} \frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2 + \gamma_k \|f\|^2_{\mathcal{H}_k} + \gamma_M \mathbf{f}_D^T \mathbf{M} \mathbf{f}_D$$

# Optimization and Manifold Regularization

The optimization problem can be recast as:

$$f^* = \underset{f \in \mathcal{H}_k}{\operatorname{argmin}} \frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2 + \gamma_k \|f\|^2_{\mathcal{H}_k} + \gamma_M \mathbf{f}_D^T \mathbf{M} \mathbf{f}_D$$

**The representer theorem:**

The minimizer over $\mathcal{H}_k$ of the regularized optimization is represented by:

$$f(\mathbf{h}) = \sum_{i=1}^{n_D} a_i k(\mathbf{h}_i, \mathbf{h})$$

where $k : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is the reproducing kernel of $\mathcal{H}_k$
with $K_{ij} = k(\mathbf{h}_i, \mathbf{h}_j) = \exp\left\{-\frac{\|\mathbf{h}_i - \mathbf{h}_j\|^2}{\varepsilon}\right\}$

# Optimization and Manifold Regularization

The optimization problem can be recast as:

$$f^* = \underset{f \in \mathcal{H}_k}{\operatorname{argmin}} \frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2 + \gamma_k \|f\|_{\mathcal{H}_k}^2 + \gamma_M \mathbf{f}_D^T \mathbf{M} \mathbf{f}_D$$

**The representer theorem:**

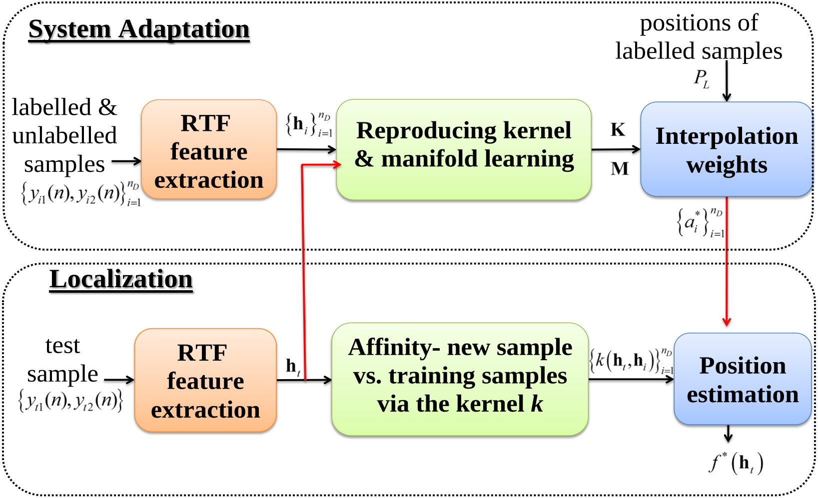
The minimizer over $\mathcal{H}_k$ of the regularized optimization is represented by:

$$f(\mathbf{h}) = \sum_{i=1}^{n_D} a_i k(\mathbf{h}_i, \mathbf{h}) \quad \Rightarrow \quad \text{closed-form solution for } \mathbf{a}^*$$

where $k : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is the reproducing kernel of $\mathcal{H}_k$

# Manifold Regularization for Localization (MRL)

[Laufer-Goldshtein et al., 2016c]




$K_{ij} = k(\mathbf{h}_i, \mathbf{h}_j)$

# Bayesian Inference on Manifold [Laufer-Goldshtein et al., 2016a][Sindhwani et al., 2007]

# Bayesian Inference on Manifold [Laufer-Goldshtein et al., 2016a][Sindhwani et al., 2007]

$$f^* = \text{argmin}_{f \in \mathcal{H}_k} \frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2 + \gamma_k \|f\|^2_{\mathcal{H}_k} + \gamma_M \mathbf{f}_D^T \mathbf{M} \mathbf{f}_D$$

**Search in RKHS defined by the kernel $k$**

**Cost Function**

$\mathcal{H}_k$ **norm**

**Manifold Regularization**

# Bayesian Inference on Manifold [Laufer-Goldshtein et al., 2016a][Sindhwani et al., 2007]

$$f^* = \mathrm{argmin}_{f \in \mathcal{H}_k} \underbrace{\frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2}_{\text{Cost Function}} + \underbrace{\gamma_k \|f\|^2_{\mathcal{H}_k}}_{\mathcal{H}_k \text{ norm}} + \underbrace{\gamma_M \mathbf{f}_D^T \mathbf{M} \mathbf{f}_D}_{\text{Manifold Regularization}}$$

**Search in RKHS defined by the kernel $k$**

$$f^* = \mathrm{argmin}_{f \in \mathcal{H}_{\tilde{k}}} \underbrace{\frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2}_{\text{Cost Function}} + \underbrace{\gamma_k \|f\|^2_{\mathcal{H}_{\tilde{k}}}}_{\mathcal{H}_{\tilde{k}} \text{ norm}}$$

**Search in RKHS defined by the kernel $\tilde{k}$**

# Bayesian Inference on Manifold [Laufer-Goldshtein et al., 2016a][Sindhwani et al., 2007]

$$f^* = \text{argmin}_{f \in \mathcal{H}_k} \frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2 + \gamma_k \|f\|^2_{\mathcal{H}_k} + \gamma_M \mathbf{f}_D^T \mathbf{M} \mathbf{f}_D$$

- $\text{argmin}_{f \in \mathcal{H}_k}$: **Search in RKHS defined by the kernel $k$**
- $\frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2$: **Cost Function**
- $\gamma_k \|f\|^2_{\mathcal{H}_k}$: **$\mathcal{H}_k$ norm**
- $\gamma_M \mathbf{f}_D^T \mathbf{M} \mathbf{f}_D$: **Manifold Regularization**

$$f^* = \text{argmin}_{f \in \mathcal{H}_{\tilde{k}}} \frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2 + \gamma_k \|f\|^2_{\mathcal{H}_{\tilde{k}}}$$

- $\text{argmin}_{f \in \mathcal{H}_{\tilde{k}}}$: **Search in RKHS defined by the kernel $\tilde{k}$**
- $\frac{1}{n_L} \sum_{i=1}^{n_L} (\bar{p}_i - f(\mathbf{h}_i))^2$: **Cost Function** → Likelihood Function
- $\gamma_k \|f\|^2_{\mathcal{H}_{\tilde{k}}}$: **$\mathcal{H}_{\tilde{k}}$ norm** → Manifold-Based Prior

$$p(f|P_L, H_L, H_U) \propto p(P_L|f, H_L) \cdot p(f|H_L, H_U)$$

- $p(f|P_L, H_L, H_U)$: **Posterior** → **$f$ is a Gaussian Process with Covariance $\tilde{k}$**
- $p(P_L|f, H_L)$: **Likelihood Function**
- $p(f|H_L, H_U)$: **Manifold-Based Prior**

# Localization

## MAP/MMSE estimator:

- Goal: estimate the function value at some test sample $\mathbf{h}_t \in \mathcal{M}$
- The training positions $\bar{\mathbf{p}}_L = \mathrm{vec}\{P_L\}$ and $f(\mathbf{h}_t)$ are jointly Gaussian ($\tilde{\boldsymbol{\Sigma}}_{HH} \Leftrightarrow \tilde{k}(\mathbf{h}_i, \mathbf{h}_j)$):

$$\begin{bmatrix} \bar{\mathbf{p}}_L \\ f(\mathbf{h}_t) \end{bmatrix} \Bigg| H_L, H_U \sim \mathcal{N}\left(\mathbf{0}_{n_L+1}, \begin{bmatrix} \tilde{\boldsymbol{\Sigma}}_{LL} + \sigma^2 \mathbf{I}_{n_L} & \tilde{\boldsymbol{\Sigma}}_{Lt} \\ \tilde{\boldsymbol{\Sigma}}_{Lt}^T & \tilde{\boldsymbol{\Sigma}}_{tt} \end{bmatrix}\right)$$

- The posterior $p(f(\mathbf{h}_t)|P_L, H_L, H_U)$ is a multivariate Gaussian with:

$$\mu_{\mathrm{cond}} = \tilde{\boldsymbol{\Sigma}}_{Lt}^T \left(\tilde{\boldsymbol{\Sigma}}_{LL} + \sigma^2 \mathbf{I}_{n_L}\right)^{-1} \bar{\mathbf{p}}_L$$

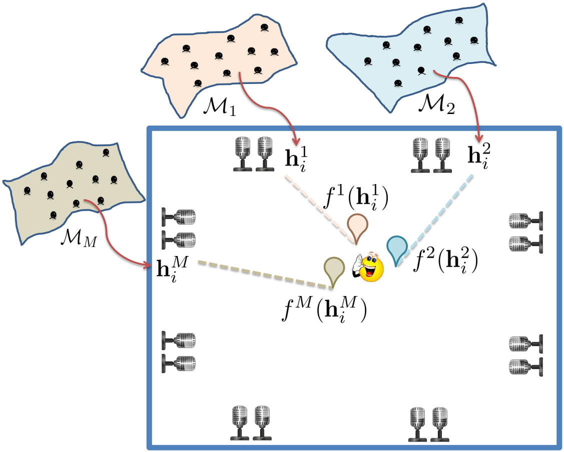
$$\sigma^2_{\mathrm{cond}} = \tilde{\Sigma}_{tt} - \tilde{\boldsymbol{\Sigma}}_{Lt}^T \left(\tilde{\boldsymbol{\Sigma}}_{LL} + \sigma^2 \mathbf{I}_{n_L}\right)^{-1} \tilde{\boldsymbol{\Sigma}}_{Lt}$$

The MAP/MMSE estimator of $f(\mathbf{h}_t)$ is given by:

$$\hat{f}(\mathbf{h}_t) = \mu_{\mathrm{cond}} = \tilde{\boldsymbol{\Sigma}}_{Lt}^T \left(\tilde{\boldsymbol{\Sigma}}_{LL} + \sigma^2 \mathbf{I}_{n_L}\right)^{-1} \bar{\mathbf{p}}_L$$

# Outline

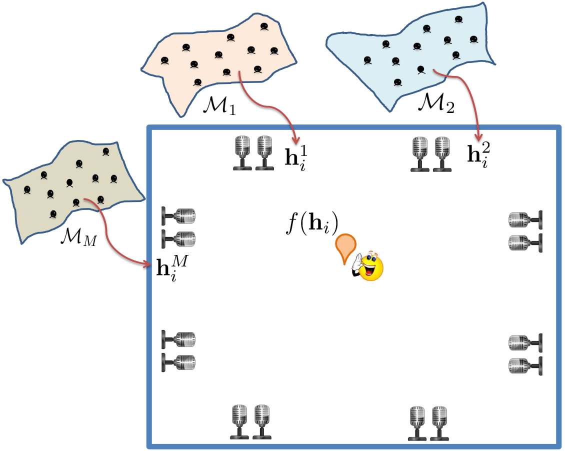
# Source Localization with Ad Hoc Array [Laufer-Goldshtein et al., 2016d]




## Each node:

- Represents a different point of view on the same acoustic event
- Induces relations between RTFs w.r.t. the associated manifold

# Source Localization with Ad Hoc Array [Laufer-Goldshtein et al., 2016d]

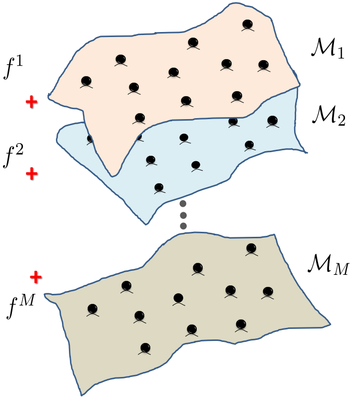



$$\mathbf{h}_i = [(\mathbf{h}_i^1)^T, (\mathbf{h}_i^2)^T, \ldots, (\mathbf{h}_i^M)^T]^T \in \cup_{m=1}^M \mathcal{M}_m$$

How to **fuse** the different views in a unified mapping $f : \cup_{m=1}^M \mathcal{M}_m \mapsto \mathbb{R}$ ?

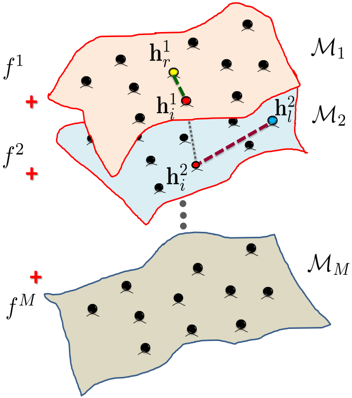
# Multiple Manifold Gaussian Process (MMGP)

Define the average process $f = \frac{1}{M}(f^1 + f^2 + \ldots + f^M) \sim \mathcal{GP}(0, \tilde{k})$



# Multiple Manifold Gaussian Process (MMGP)

Define the average process $f = \frac{1}{M}(f^1 + f^2 + \ldots + f^M) \sim \mathcal{GP}(0, \tilde{k})$

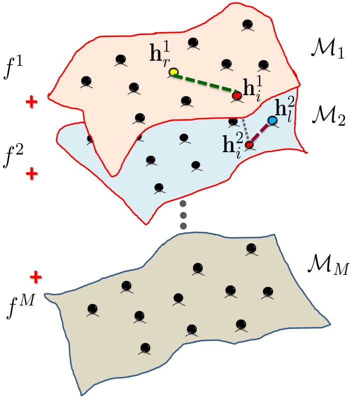



**The covariance between $p_r = f(\mathbf{h}_r)$ and $p_l = f(\mathbf{h}_l)$**

$$\text{cov}(f(\mathbf{h}_r), f(\mathbf{h}_l)) \equiv \tilde{k}(\mathbf{h}_r, \mathbf{h}_l) = \frac{1}{M^2} \sum_{q,w=1}^{M} \sum_{i=1}^{n_D} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q) k_w(\mathbf{h}_l^w, \mathbf{h}_i^w)$$

# Multiple Manifold Gaussian Process (MMGP)

Define the average process $f = \frac{1}{M}(f^1 + f^2 + \ldots + f^M) \sim \mathcal{GP}(0, \tilde{k})$

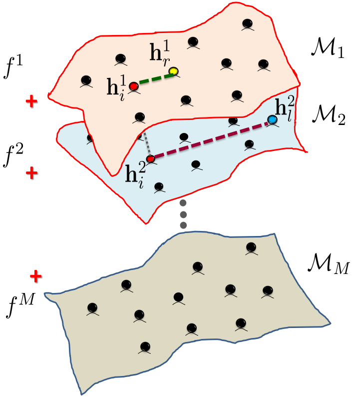



**The covariance between $p_r = f(\mathbf{h}_r)$ and $p_l = f(\mathbf{h}_l)$**

$$\mathrm{cov}(f(\mathbf{h}_r), f(\mathbf{h}_l)) \equiv \tilde{k}(\mathbf{h}_r, \mathbf{h}_l) = \frac{1}{M^2} \sum_{q,w=1}^{M} \sum_{i=1}^{n_D} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q) k_w(\mathbf{h}_l^w, \mathbf{h}_i^w)$$

# Multiple Manifold Gaussian Process (MMGP)

Define the average process $f = \frac{1}{M}(f^1 + f^2 + \ldots + f^M) \sim \mathcal{GP}(0, \tilde{k})$

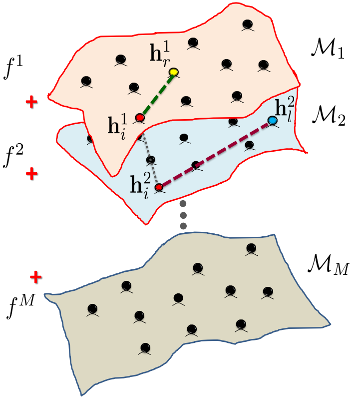



**The covariance between $p_r = f(\mathbf{h}_r)$ and $p_l = f(\mathbf{h}_l)$**

$$\mathrm{cov}(f(\mathbf{h}_r), f(\mathbf{h}_l)) \equiv \tilde{k}(\mathbf{h}_r, \mathbf{h}_l) = \frac{1}{M^2} \sum_{q,w=1}^{M} \sum_{i=1}^{n_D} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q) k_w(\mathbf{h}_l^w, \mathbf{h}_i^w)$$

# Multiple Manifold Gaussian Process (MMGP)

Define the average process $f = \frac{1}{M}(f^1 + f^2 + \ldots + f^M) \sim \mathcal{GP}(0, \tilde{k})$

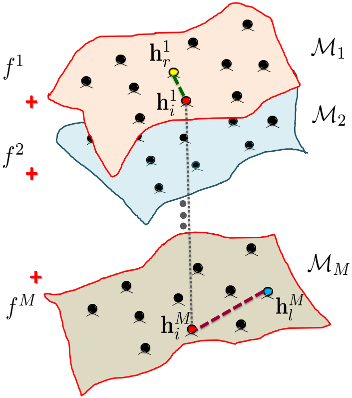



**The covariance between $p_r = f(\mathbf{h}_r)$ and $p_l = f(\mathbf{h}_l)$**

$$\mathrm{cov}(f(\mathbf{h}_r), f(\mathbf{h}_l)) \equiv \tilde{k}(\mathbf{h}_r, \mathbf{h}_l) = \frac{1}{M^2} \sum_{q,w=1}^{M} \sum_{i=1}^{n_D} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q) k_w(\mathbf{h}_l^w, \mathbf{h}_i^w)$$

# Multiple Manifold Gaussian Process (MMGP)

Define the average process $f = \frac{1}{M}(f^1 + f^2 + \ldots + f^M) \sim \mathcal{GP}(0, \tilde{k})$

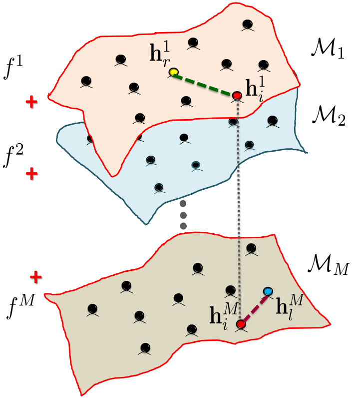



**The covariance between $p_r = f(\mathbf{h}_r)$ and $p_l = f(\mathbf{h}_l)$**

$$\mathrm{cov}(f(\mathbf{h}_r), f(\mathbf{h}_l)) \equiv \tilde{k}(\mathbf{h}_r, \mathbf{h}_l) = \frac{1}{M^2} \sum_{q,w=1}^{M} \sum_{i=1}^{n_D} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q) k_w(\mathbf{h}_l^w, \mathbf{h}_i^w)$$

# Multiple Manifold Gaussian Process (MMGP)

Define the average process $f = \frac{1}{M}(f^1 + f^2 + \ldots + f^M) \sim \mathcal{GP}(0, \tilde{k})$

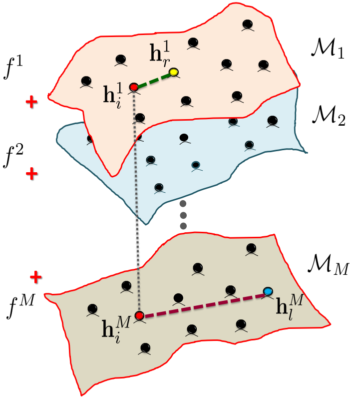



**The covariance between $p_r = f(\mathbf{h}_r)$ and $p_l = f(\mathbf{h}_l)$**

$$\mathrm{cov}(f(\mathbf{h}_r), f(\mathbf{h}_l)) \equiv \tilde{k}(\mathbf{h}_r, \mathbf{h}_l) = \frac{1}{M^2} \sum_{q,w=1}^{M} \sum_{i=1}^{n_D} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q) k_w(\mathbf{h}_l^w, \mathbf{h}_i^w)$$

# Multiple Manifold Gaussian Process (MMGP)

Define the average process $f = \frac{1}{M}(f^1 + f^2 + \ldots + f^M) \sim \mathcal{GP}(0, \tilde{k})$

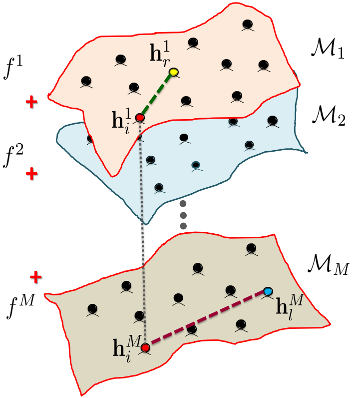



**The covariance between $p_r = f(\mathbf{h}_r)$ and $p_l = f(\mathbf{h}_l)$**

$$\text{cov}(f(\mathbf{h}_r), f(\mathbf{h}_l)) \equiv \tilde{k}(\mathbf{h}_r, \mathbf{h}_l) = \frac{1}{M^2} \sum_{q,w=1}^{M} \sum_{i=1}^{n_D} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q) k_w(\mathbf{h}_l^w, \mathbf{h}_i^w)$$

# Multiple Manifold Gaussian Process (MMGP)

Define the average process $f = \frac{1}{M}(f^1 + f^2 + \ldots + f^M) \sim \mathcal{GP}(0, \tilde{k})$




**The covariance between $p_r = f(\mathbf{h}_r)$ and $p_l = f(\mathbf{h}_l)$**

$$\mathrm{cov}(f(\mathbf{h}_r), f(\mathbf{h}_l)) \equiv \tilde{k}(\mathbf{h}_r, \mathbf{h}_l) = \frac{1}{M^2} \sum_{q,w=1}^{M} \sum_{i=1}^{n_D} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q) k_w(\mathbf{h}_l^w, \mathbf{h}_i^w)$$

# Localization

## MAP/MMSE estimator:

- Goal: estimate the function value at some test sample $\mathbf{h}_t \in \mathcal{M}$
- The training positions $\bar{\mathbf{p}}_L = \mathrm{vec}\{P_L\}$ and $f(\mathbf{h}_t)$ are jointly Gaussian ($\tilde{\mathbf{\Sigma}}_{HH} \Leftrightarrow \tilde{k}(\mathbf{h}_i, \mathbf{h}_j)$):

$$\begin{bmatrix} \bar{\mathbf{p}}_L \\ f(\mathbf{h}_t) \end{bmatrix} \Big| H_L, H_U \sim \mathcal{N}\left(\mathbf{0}_{n_L+1}, \begin{bmatrix} \tilde{\mathbf{\Sigma}}_{LL} + \sigma^2 \mathbf{I}_{n_L} & \tilde{\mathbf{\Sigma}}_{Lt} \\ \tilde{\mathbf{\Sigma}}_{Lt}^T & \tilde{\mathbf{\Sigma}}_{tt} \end{bmatrix}\right)$$

- The posterior $p(f(\mathbf{h}_t)|P_L, H_L, H_U)$ is a multivariate Gaussian with:

$$\mu_{\mathrm{cond}} = \tilde{\mathbf{\Sigma}}_{Lt}^T \left(\tilde{\mathbf{\Sigma}}_{LL} + \sigma^2 \mathbf{I}_{n_L}\right)^{-1} \bar{\mathbf{p}}_L$$

$$\sigma^2_{\mathrm{cond}} = \tilde{\Sigma}_{tt} - \tilde{\mathbf{\Sigma}}_{Lt}^T \left(\tilde{\mathbf{\Sigma}}_{LL} + \sigma^2 \mathbf{I}_{n_L}\right)^{-1} \tilde{\mathbf{\Sigma}}_{Lt}$$

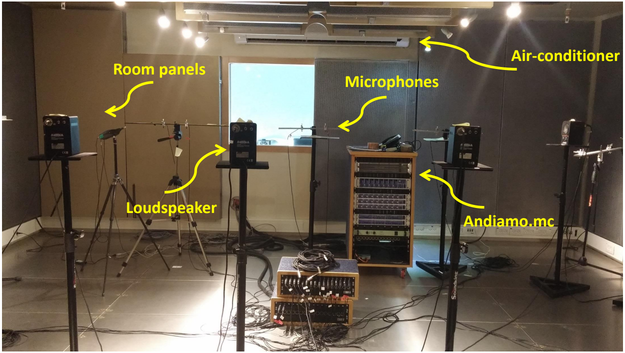
The MAP/MMSE estimator of $f(\mathbf{h}_t)$ is given by:

$$\hat{f}(\mathbf{h}_t) = \mu_{\mathrm{cond}} = \tilde{\mathbf{\Sigma}}_{Lt}^T \left(\tilde{\mathbf{\Sigma}}_{LL} + \sigma^2 \mathbf{I}_{n_L}\right)^{-1} \bar{\mathbf{p}}_L$$

# Recordings Setup

## Setup:

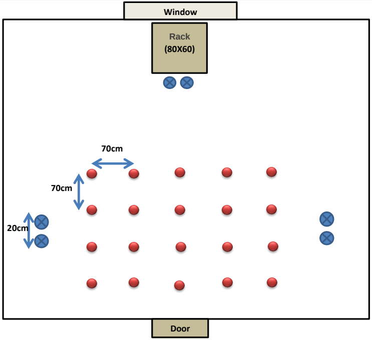
- Real recordings carried out at Bar-Ilan acoustic lab
- A $6 \times 6 \times 2.4$m room controllable reverberation time (set to 620ms)
- Region of interest: Source position is confined to a $2.8 \times 2.1$m area
- 3 microphone pairs with inter-distance of 0.2m (position unknown)



# Recordings Setup

## Setup:

- Real recordings carried out at Bar-Ilan acoustic lab
- A $6 \times 6 \times 2.4$m room controllable reverberation time (set to 620ms)
- Region of interest: Source position is confined to a $2.8 \times 2.1$m area
- 3 microphone pairs with inter-distance of 0.2m (position unknown)



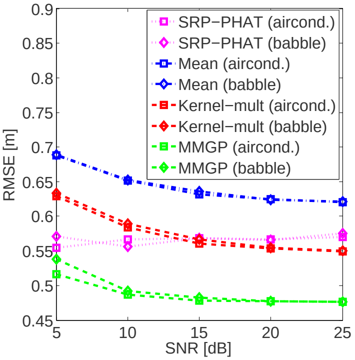
# Experimental Results [Laufer-Goldshtein et al., 2016d]

## Setup:

- Training: 20 labelled samples (0.7m resolution), 50 unlabelled samples
- Test: 25 random samples in the defined region
- Two noise types: air-conditioner noise and babble noise

## Compare with:

- Concatenated independent measurements (Kernel-mult)
- Average of single-node estimates (Mean)
- Beamformer scanning (SRP-PHAT [DiBiase et al., 2001])



# Outline

# Dynamic Scenario

## Received Signals

$$y^{mi}(n) = \sum_k a_n^{mi}(k)s(n-k) + u^{mi}(n); \quad m = 1, \ldots, M, i = 1, 2$$

- $a_n^{mi}$ - a time-varying AIR at node $m$, microphone $i$ in time $n$
- $\mathbf{h}^m(t)$ - the RTF vector at node $m$ in the STFT frame $t$
- $\mathbf{h}(t) = \left[[\mathbf{h}(t)^1]^T, \ldots, [\mathbf{h}(t)^M]^T\right]^T$ - a concatenation of the RTF vectors from all nodes
- $p_c(t) = f(\mathbf{h}(t)), c \in \{x, y, z\}$ - mapping of the concatenated RTF vector to position (for brevity $p_c(t) \equiv p(t)$)

## Reminder: The covariance between $p_r = f(\mathbf{h}_r)$ and $p_l = f(\mathbf{h}_l)$

$$\mathrm{cov}(f(\mathbf{h}_r), f(\mathbf{h}_l)) \equiv \tilde{k}(\mathbf{h}_r, \mathbf{h}_l) = \frac{1}{M^2} \sum_{q,w=1}^{M} \sum_{i=1}^{n_D} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q) k_w(\mathbf{h}_l^w, \mathbf{h}_i^w)$$

# Bayesian Inference for Source Tracking

## Standard (Nonlinear) State-Space Model

$$p(t) = b_t(p(t-1)) + \xi_t$$
$$q_t = c_t(p(t)) + \zeta_t$$

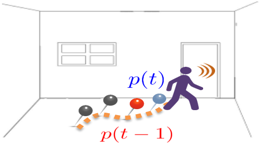
# Bayesian Inference for Source Tracking

## Standard (Nonlinear) State-Space Model

$$p(t) = b_t(p(t-1)) + \xi_t$$
$$q_t = c_t(p(t)) + \zeta_t$$

## Propagation Model

- Relate current and previous positions arbitrarily using random walk or Langevin
- Independent of measurements
- Noise statistics is unknown



# Bayesian Inference for Source Tracking

## Standard (Nonlinear) State-Space Model

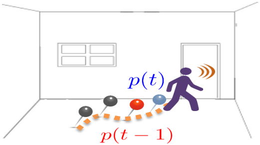
$$p(t) = b_t(p(t-1)) + \xi_t$$
$$q_t = c_t(p(t)) + \zeta_t$$

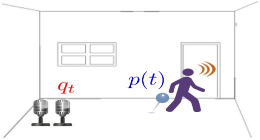
## Propagation Model

- Relate current and previous positions arbitrarily using random walk or Langevin
- Independent of measurements
- Noise statistics is unknown

## Observation Model

- Relate current position to measurements
- Examples: TDOA or steered response power readings
- Noise statistics is unknown

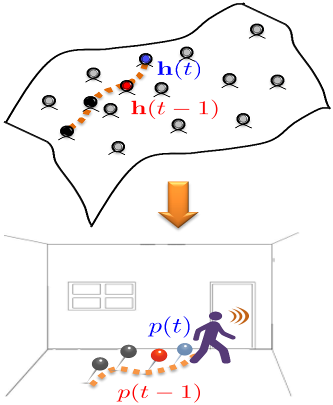


# Tracking on the Manifold [Laufer-Goldshtein et al., 2017]

## Propagation Model - Local

Transform nonlinear regression of high-dimensional RTFs to linear transition of source positions



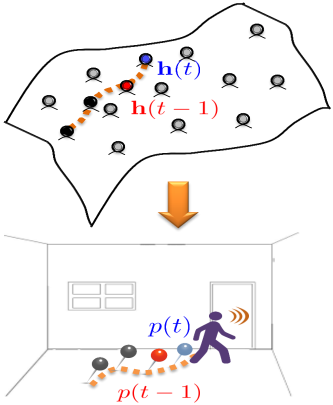
# Tracking on the Manifold [Laufer-Goldshtein et al., 2017]

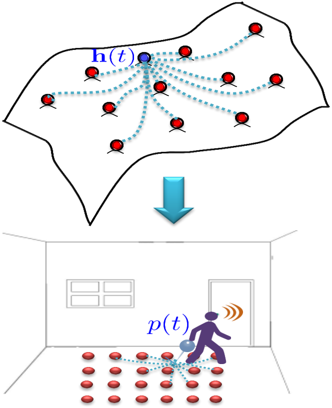
## Propagation Model - Local

Transforms nonlinear regression of high-dimensional RTFs to linear transition of source positions

## Observation model - Global

Formed by a regression of training positions according to relations on the manifold



# State Space Representation (1)

## Probabilistic Motion Model:

- Current and previous positions, $p(t) = f(\mathbf{h}(t))$ and $p(t-1) = f(\mathbf{h}(t-1))$, are jointly GP:

$$\begin{bmatrix} p(t) \\ p(t-1) \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \tilde{\Sigma}_{t,t} & \tilde{\Sigma}_{t,t-1} \\ \tilde{\Sigma}_{t,t-1} & \tilde{\Sigma}_{t-1,t-1} \end{bmatrix}\right)$$

- Their conditional probability is given by:

$$p(t)|p(t-1) \sim \mathcal{N}\left(\frac{\tilde{\Sigma}_{t,t-1}}{\tilde{\Sigma}_{t-1,t-1}} p(t-1), \tilde{\Sigma}_{t,t} - \frac{\tilde{\Sigma}^2_{t,t-1}}{\tilde{\Sigma}_{t-1,t-1}}\right)$$

where $\tilde{\Sigma}_{t,\tau} \equiv \tilde{k}\left(\mathbf{h}(t), \mathbf{h}(\tau)\right)$

# State Space Representation (2)

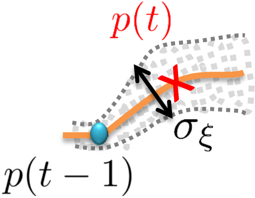
## Propagation Model:

Can be transformed into a linear propagation equation with an additive Gaussian noise $\xi_t$:

$$p(t) = b_t \cdot p(t-1) + \xi_t$$

with

- $b_t = \frac{\tilde{\Sigma}_{t,t-1}}{\tilde{\Sigma}_{t-1,t-1}}$ - The Wiener filter
- $\xi_t \sim \mathcal{N}\left(0, \sigma_\xi^2\right)$ with $\sigma_\xi^2 = \tilde{\Sigma}_{t,t} - \frac{\tilde{\Sigma}^2_{t,t-1}}{\tilde{\Sigma}_{t-1,t-1}}$, the corresponding variance



# State Space Representation (3)

## Probabilistic Motion Model:

- $\bar{\mathbf{p}}_L = [\bar{p}_1, \ldots, \bar{p}_{n_L}]^T$ - measured positions of the labelled set
- $\bar{p}_i = p_i + \eta_i$ - noisy versions of the actual position $p_i$
- $\eta_i$ - independent Gaussian noise with variance $\sigma^2$
- $p(t) = f(\mathbf{h}(t))$ and $\bar{\mathbf{p}}_L$ are jointly GP:

$$\begin{bmatrix} p(t) \\ \bar{\mathbf{p}}_L \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \tilde{\Sigma}_{t,t} & \tilde{\boldsymbol{\Sigma}}_{Lt} \\ \tilde{\boldsymbol{\Sigma}}_{Lt} & \tilde{\boldsymbol{\Sigma}}_{LL} + \sigma^2 \mathbf{I}_{n_L} \end{bmatrix}\right)$$

- Their conditional probability is given by:

$$p(t)|\bar{\mathbf{p}}_L \sim \mathcal{N}\left(\tilde{\boldsymbol{\Sigma}}_{Lt}^H \left(\tilde{\boldsymbol{\Sigma}}_L + \sigma^2 \mathbf{I}_{n_L}\right)^{-1} \bar{\mathbf{p}}_L, \tilde{\Sigma}_{t,t} - \tilde{\boldsymbol{\Sigma}}_{Lt}^H \left(\tilde{\boldsymbol{\Sigma}}_L + \sigma^2 \mathbf{I}_{n_L}\right)^{-1} \tilde{\boldsymbol{\Sigma}}_{Lt}\right)$$

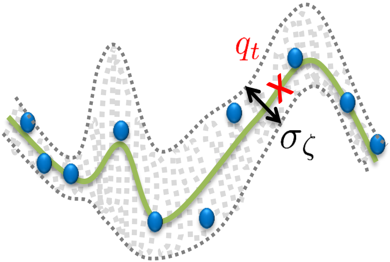
# State-Space Representation (4)

## Observation model:

- Can be transformed into a noisy *artificial observation $q_t$* that represents a linear regression on the training set:

$$q_t = \mathbf{Q}_t \bar{\mathbf{p}}_L$$

where $\mathbf{Q}_t = \tilde{\boldsymbol{\Sigma}}_{Lt}^H \left( \tilde{\boldsymbol{\Sigma}}_{LL} + \sigma^2 \mathbf{I}_{n_L} \right)^{-1}$




The corresponding observation model:

$$q_t = p(t) + \zeta_t$$

where $\zeta_t \sim \mathcal{N}\left(0, \sigma_\zeta^2\right)$ with $\sigma_\zeta^2 = \tilde{\Sigma}_{t,t} - \tilde{\boldsymbol{\Sigma}}_{Lt}^H \left( \tilde{\boldsymbol{\Sigma}}_{LL} + \mathbf{I}_{n_L} \right)^{-1} \tilde{\boldsymbol{\Sigma}}_{Lt}$.

# Tracking Algorithm

## Space-State Representation:

The proposed state-space model is given by:

$$p(t) = b_t \cdot p(t-1) + \xi_t$$

$$q_t = p(t) + \zeta_t$$

**Kalman Filter**

### Time Propagation

- Predicted Position:

$$\hat{p}(t \mid t-1) = b_t \cdot \hat{p}(t-1 \mid t-1)$$

- Predicted Covariance:

$$\gamma(t \mid t-1) = g_t^2 \gamma(t-1 \mid t-1) + \sigma_\xi^2$$

### Measurement Update

- Kalman Gain:

$$\kappa(t) = \frac{\gamma(t \mid t-1)}{\gamma(t \mid t-1) + \sigma_\zeta^2}$$

- Updated position estimate:

$$\hat{p}(t \mid t) = \hat{p}(t \mid t-1) + \kappa(t)\left(q_t - \hat{p}(t \mid t-1)\right)$$

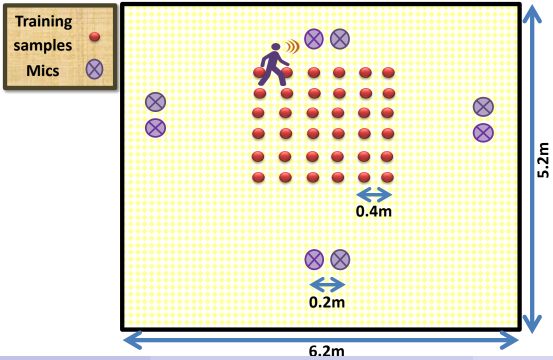
- Updated Covariance:

$$\gamma(t \mid t) = \left(1 - \kappa(t)\right)\gamma(t \mid t-1)$$

# Experimental Results

## Setup:

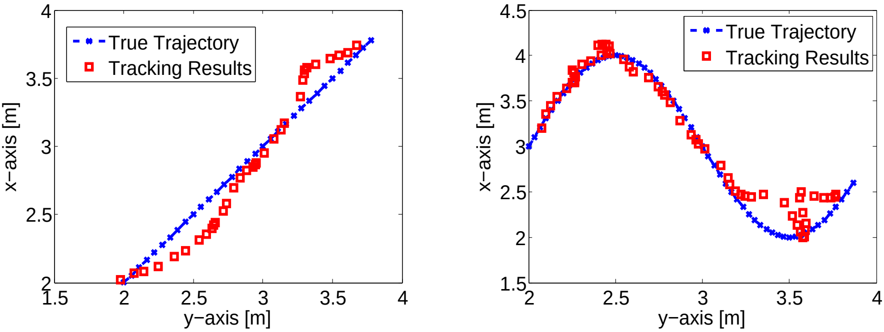
- A $5.2 \times 6.2 \times 3$m room with $T_{60} = 300$ms
- $M = 4$ nodes with 0.2m distance between microphones
- **Region of interest:** a $2 \times 2$m square region
- **Training:** 36 samples (0.4m resolution)



# Results

## Test:

- **Trajectories:** straight line (for 3s) and sinusoidal movement (for 5s).
- **Velocity:** approximately 1m/s




**RMSE:** 13cm for straight line and 17cm for sinusoidal movement.

# Challenges and Perspectives

## Manifold learning approach for source localization

- Data-driven manifold inference with a few labeled anchor positions and unknown microphone positions.
- Location is shown to be the controlling variable of the RTF manifold
- It's practical!
- Active research field [Deleforge et al., 2015][Yu et al., 2016][Xiao et al., 2015]
- Improved speaker tracking ⇒ Hybrid approach [Laufer-Goldshtein et al., 2018]

## Challenges

- Robustness to changes in array constellation and acoustic scenario
- Application to multiple concurrent speakers
- Beamforming - more complicated as it targets enhanced speech rather than its location
  - A first attempt using projections to the inferred manifold [Talmon and Gannot, 2013]

# Manifold Regularization

## Measuring smoothness over $\mathcal{M}$:

- The gradient $\bigtriangledown_{\mathcal{M}} f(\mathbf{h})$ represents variations around $\mathbf{h}$
- A natural choice for intrinsic regularization:

$$\|f\|^2_{\mathcal{M}} = \int_{\mathcal{M}} \| \bigtriangledown_{\mathcal{M}} f(\mathbf{h})\|^2 dp(\mathbf{h})$$

which is a global measure of smoothness for $f$

- Stokes' theorem links gradient and Laplacian:

$$\int_{\mathcal{M}} \| \bigtriangledown_{\mathcal{M}} f(\mathbf{h})\|^2 dp(\mathbf{h}) = \int_{\mathcal{M}} f(\mathbf{h}) \bigtriangleup_{\mathcal{M}} f(\mathbf{h}) dp(\mathbf{h}) = \langle f(\mathbf{h}), \bigtriangleup_{\mathcal{M}} f(\mathbf{h}) \rangle$$

where $\bigtriangleup_{\mathcal{M}}$ is the Laplace-Beltrami operator

How to reconstruct the Laplace-Beltrami operator on $\mathcal{M}$, given the training samples from the manifold?

# Manifold Regularization

## Graph Laplacian:

- The manifold is empirically represented by a graph $G$, with weights:

$$W_{ij} = \begin{cases} \exp\left\{-\frac{\|\mathbf{h}_i - \mathbf{h}_j\|^2}{\varepsilon_w}\right\} & \text{if } \mathbf{h}_j \in \mathcal{N}_i \text{ or } \mathbf{h}_i \in \mathcal{N}_j \\ 0 & \text{otherwise} \end{cases}$$

  where $\mathcal{N}_j$ is a set consisting of the $d$ nearest-neighbours of $\mathbf{h}_j$.
- The graph Laplacian of $G$: $\mathbf{M} = \mathbf{S} - \mathbf{W}$, with $S_{ii} = \sum_{j=1}^{n_D} \mathbf{W}_{ij}$.
- Smoothness functional of $G$:

$$\langle \mathbf{f}_D, \mathbf{M}\mathbf{f}_D \rangle = \mathbf{f}_D^T \mathbf{M} \mathbf{f}_D$$

  where $\mathbf{f}_D = [f(\mathbf{h}_1), ..., f(\mathbf{h}_{n_D})]$
- It can be shown: Back

$$\mathbf{f}_D^T \mathbf{M} \mathbf{f}_D = \frac{1}{2} \sum_{i,j=1}^{n_D} W_{ij} \left(f(\mathbf{h}_i) - f(\mathbf{h}_j)\right)^2$$

# References

[Allen and Berkley, 1979] Allen, J. and Berkley, D. (1979).
Image method for efficiently simulating small-room acoustics.
*J. Acoustical Society of America*, 65(4):943–950.

[Belkin et al., 2006] Belkin, M., Niyogi, P., and Sindhwani, V. (2006).
Manifold regularization: A geometric framework for learning from labeled and unlabeled examples.
*Journal of Machine Learning Research*.

[Coifman and Lafon, 2006] Coifman, R. and Lafon, S. (2006).
Diffusion maps.
*Appl. Comput. Harmon. Anal.*, 21:5–30.

[Dal-Degan and Prati, 1988] Dal-Degan, N. and Prati, C. (1988).
Acoustic noise analysis and speech enhancement techniques for mobile radio application.
*Signal Processing*, 15(4):43–56.

[Deleforge et al., 2015] Deleforge, A., Forbes, F., and Horaud, R. (2015).
Acoustic space learning for sound-source separation and localization on binaural manifolds.
*International journal of neural systems*, 25(1).

[DiBiase et al., 2001] DiBiase, J. H., Silverman, H. F., and Brandstein, M. S. (2001).
Robust localization in reverberant rooms.
In *Microphone Arrays*, pages 157–180. Springer.

[Gannot et al., 2001] Gannot, S., Burshtein, D., and Weinstein, E. (2001).
Signal enhancement using beamforming and nonstationarity with applications to speech.
*IEEE Transactions on Signal Processing*, 49(8):1614 –1626.

[Habets and Gannot, 2007] Habets, E. and Gannot, S. (2007).
Generating sensor signals in isotropic noise fields.
*The Journal of the Acoustical Society of America*, 122:3464–3470.

# References (cont.)

[Jot et al., 1997] Jot, J.-M., Cerveau, L., and Warusfel, O. (1997).
Analysis and synthesis of room reverberation based on a statistical time-frequency model.
In *Audio Engineering Society Convention 103*. Audio Engineering Society.

[Koldovsky et al., 2014] Koldovsky, Z., Malek, J., and Gannot, S. (2014).
Spatial source subtraction based on incomplete measurements of relative transfer function.
*IEEE Transactions on Audio, Speech, and Language Processing*, 23(8):1335–1347.

[Laufer-Goldshtein et al., 2015] Laufer-Goldshtein, B., Talmon, R., and Gannot, S. (2015).
Study on manifolds of acoustic responses.
In *Interntional Conference on Latent Variable Analysis and Signal Seperation (LVA/ICA)*, Liberec, Czech Republic.

[Laufer-Goldshtein et al., 2016a] Laufer-Goldshtein, B., Talmon, R., and Gannot, S. (2016a).
Manifold-based Bayesian inference for semi-supervised source localization.
In *IEEE International Conference on Audio and Acoustic Signal Processing (ICASSP)*, Shanghai, China.

[Laufer-Goldshtein et al., 2016b] Laufer-Goldshtein, B., Talmon, R., and Gannot, S. (2016b).
A real-life experimental study on semi-supervised source localization based on manifold regularization.
In *International conference on the science of electrical engineering (ICSEE)*, Eilat, Israel.

[Laufer-Goldshtein et al., 2016c] Laufer-Goldshtein, B., Talmon, R., and Gannot, S. (2016c).
Semi-supervised sound source localization based on manifold regularization.
*IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(8):1393–1407.

[Laufer-Goldshtein et al., 2016d] Laufer-Goldshtein, B., Talmon, R., and Gannot, S. (2016d).
Semi-supervised source localization on multiple-manifolds with distributed microphones.
*arXiv preprint arXiv:1610.04770*.

[Laufer-Goldshtein et al., 2017] Laufer-Goldshtein, B., Talmon, R., and Gannot, S. (2017).
Speaker tracking on multiple-manifolds with distributed microphones.
In *The 13th International Conference on Latent Variable Analysis and Signal Separation (LVA-ICA)*, Grenoble, France.

# References (cont.)

[Laufer-Goldshtein et al., 2018] Laufer-Goldshtein, B., Talmon, R., and Gannot, S. (2018).
A hybrid approach for speaker tracking based on TDOA and data-driven models.
*IEEE Tran. Audio, Speech, Lang. Process.*, 26(4):725–735.

[Markovich-Golan and Gannot, 2015] Markovich-Golan, S. and Gannot, S. (2015).
Performance analysis of the covariance subtraction method for relative transfer function estimation and comparison to the covariance whitening method.
In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 544–548, Brisbane, Australia.

[Peterson, 1986] Peterson, P. (1986).
Simulating the response of multiple microphones to a single acoustic source in a reverberant room.
*J. Acoust. Soc. Am*, 76(5):1527–1529.

[Polack, 1993] Polack, J.-D. (1993).
Playing billiards in the concert hall: The mathematical foundations of geometrical room acoustics.
*Applied Acoustics*, 38(2):235–244.

[Schroeder, 1996] Schroeder, M. R. (1996).
The "schroeder frequency" revisited.
*The Journal of the Acoustical Society of America*, 99(5):3240–3241.

[Sindhwani et al., 2007] Sindhwani, V., Chu, W., and Keerthi, S. S. (2007).
Semi-supervised gaussian process classifiers.
In *IJCAI*, pages 1059–1064.

[Talmon and Gannot, 2013] Talmon, R. and Gannot, S. (2013).
Relative transfer function identification on manifolds for supervised GSC beamformers.
In *21st European Signal Processing Conference (EUSIPCO)*, Marrakech, Morocco.

# References (cont.)

[Xiao et al., 2015] Xiao, X., Zhao, S., Zhong, X., Jones, D. L., Chng, E. S., and Li, H. (2015).
A learning-based approach to direction of arrival estimation in noisy and reverberant environments.
In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 2814–2818. IEEE.

[Yu et al., 2016] Yu, Y., Wang, W., and Han, P. (2016).
Localization based stereo speech source separation using probabilistic time-frequency masking and deep neural networks.
*EURASIP Journal on Audio, Speech, and Music Processing*, 2016(1):7.