Semi-Supervised Source Localization on Multiple Manifolds With Distributed Microphones

Bracha Laufer-Goldshtein, *Student Member, IEEE*, Ronen Talmon, *Member, IEEE*, and Sharon Gannot, *Senior Member, IEEE*

***Abstract*—The problem of single-source localization with ad hoc microphone networks in noisy and reverberant enclosures is addressed in this paper. A training set is formed by prerecorded measurements collected in advance and consists of a limited number of labelled measurements, attached with corresponding positions, and a larger number of unlabelled measurements from unknown locations. Further information about the enclosure characteristics or the microphone positions is not required. We propose a Bayesian inference approach for estimating a function that maps measurement-based features to the corresponding positions. The signals measured by the microphones represent different viewpoints, which are combined in a unified statistical framework. For this purpose, the mapping function is modelled by a Gaussian process with a covariance function that encapsulates both the connections between pairs of microphones and the relations among the samples in the training set. The parameters of the process are estimated by optimizing a maximum likelihood criterion. In addition, a recursive adaptation mechanism is derived, where the new streaming measurements are used to update the model. Performance is demonstrated for both simulated data and real-life recordings in a variety of reverberation and noise levels.**

***Index Terms*—Acoustic manifold, Gaussian process, maximum likelihood (ML), relative transfer function (RTF), sound source localization.**

I. Introduction

ACOUSTIC source localization is an essential component in various audio applications, such as: automated camera steering and teleconferencing systems [1], speaker separation [2] and robot audition [3]–[5]. Thus, the localization problem has attracted a significant research attention, and a large variety of localization methods were proposed during the last decades. The main challenge facing the research community is how to perform robust localization in adverse conditions, namely, in the presence of background noise and reverberations, which are the main causes for performance degradation of localization algorithms.

Broadly, traditional localization methods can be divided into three main categories: methods based on maximization of the steered response power (SRP) of a beamformer output, high-resolution spectral estimation techniques, and dual-stage approaches relying on a time difference of arrival (TDOA) estimation. In the first category, the position is estimated directly from the measured signals after being filtered and summed together. Commonly, the maximum likelihood (ML) criterion is applied, which in the case of a single source, culminates in inspecting the output power of a beamformer steered to different locations and in searching the points where it receives its maximum value [6]. The second category consists of high resolution methods, such as multiple signal classification (MUSIC) [7] and estimation of signal parameters via rotational invariance (ESPRIT) [8] algorithms, that are based on the spectral analysis of the correlation matrix of the measured signals. Subspace methods can also be applied using spherical harmonics [9]–[11]. In the third category, a dual stage approach is applied. In the first stage, the TDOAs of different pairs of microphones are estimated and collected. The different TDOA readings correspond to single-sided hyperbolic hyperplanes (in 3D) representing possible positions. The intersection of these hyperplanes yields the estimated position. In these types of approaches the quality of the localization greatly depends on the quality of the TDOA estimation in the first stage. The classical method for TDOA estimation, which assumes a reverberant-free model, is the generalized cross-correlation (GCC) algorithm introduced in the landmark paper by Knapp and Carter [12]. Many improvements of the generalized cross-correlation (GCC) method for the reverberant case were proposed, e.g. in [13]–[17]. Among these methods for TDOA estimation in reverberant conditions, there are subspace methods based on adaptive eigenvalue decomposition [18] and generalized eigenvalue decomposition [19]. Of special importance is the SRP-phase transform (SRP-PHAT) algorithm proposed in [20]. This method is related to both the first and the third categories, since it combines in a single step the features of a steered-beamformer with those of the phase transform weighting of the GCC algorithm.

Most of the traditional localization approaches are based on physical models and rely on certain assumptions regarding the propagation model and the statistics of the signals and the noise. However, real-world scenarios, characterized by complex reflection patterns, can be described by intricate models, which

Manuscript received September 25, 2016; revised January 23, 2017 and March 21, 2017; accepted March 26, 2017. Date of publication April 24, 2017; date of current version June 5, 2017. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Augusto Sarti. *(Corresponding author: Sharon Gannot.)*

B. Laufer-Goldshtein and S. Gannot are with the Faculty of Engineering, Bar-Ilan University, Ramat Gan 5290002, Israel (e-mail: bracha.laufer@biu.ac.il; sharon.gannot@biu.ac.il).

R. Talmon is with the Viterbi Faculty of Electrical Engineering, The Technion—Israel Institute of Technology, Haifa 3200003, Israel (e-mail: ronen@ee.technion.ac.il).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2017.2696310

are difficult to estimate. Recently, there is a growing interest in learning-based localization approaches, which attempt to learn the characteristics of the acoustic environment directly from the data, in contrast to using a predefined physical model. Typically, these approaches assume that a training set of prerecorded measurements is given in advance. Supervised methods utilize microphone measurements of sources from known locations, while unsupervised approaches solely utilize the measurements, without knowing their exact source positions.

Learning-based approaches were proposed for both microphone array localization and binaural localization. In the binaural hearing context, Deleforge and Horaud have proposed a probabilistic piecewise affine regression model that infers the localization-to-interaural data mapping and its inverse [21]. They have extended this approach to the case of multiple sources using the variational Expectation Maximization (EM) framework [22], [23]. In [24], another approach was presented based on a Gaussian Mixture Model (GMM), which was used to learn the azimuth-dependent distribution of the binaural feature space. In [25], a binaural localization method was proposed by assessing the mutual information between each of the spatial cues and the corresponding source location. In [26], GCC-based feature vectors were extracted and used for training a multilayer perceptron neural network that outputs the source direction of arrival (DOA). A method for DOA estimation of multiple sources was presented in [27], using an EM clustering approach. A localization method for a source located behind an obstacle that blocks the direct propagation path was presented in [28]. The algorithm uses co-sparse data analysis based on the physical model of the wave propagation. The model was extended in [29] to the case where the physical properties of the enclosure are not known in advance.

Talmon *et al.* [30] introduced a supervised method based on *manifold learning*, aiming at recovering the fundamental controlling parameter of the acoustic impulse response (AIR), which coincides with the source position in a static environment. The method was applied to a single microphone system with a white Gaussian noise (WGN) input [31]. In [32] we adopted the paradigm of [31] and adapted it to a speech source, using a dual-microphone system with a power spectral density (PSD)-based feature vector. Another approach for semi-supervised source localization with a single microphone pair, based on a regularized optimization in a reproducing kernel Hilbert space (RKHS), was recently presented in [33].

In this paper, we consider a setup consisting of multiple nodes, where each node comprises a pair of microphones. No additional assumptions, particularly on their specific (unknown) locations, are made. We anticipate that such an extension of the setup, comprising much more spatial information, is both practical and may lead to improved accuracy of localization tasks. In our recent work [34], we reformulated the optimization problem presented in [33] using a Bayesian inference approach for the single node case. Following [35], [36], the mapping function between the acoustic samples and their corresponding source positions, was modelled as a Gaussian process with a covariance function that was built based on a certain kernel function. This Bayesian framework serves as a corner stone for extending the single node setup to a network of multiple nodes. Here as well, we utilize a set of prerecorded measurements for identifying unique patterns and geometrical structures, which characterize the acoustic samples in a given enclosure. The gist of the algorithm is the definition of a Gaussian process with a new covariance function that merges the different viewpoints presented by the different nodes. In addition, this statistical framework allows for the rigorous estimation of the model parameters as an integral part of the optimization procedure, through an appropriate maximum likelihood (ML) criterion. Moreover, a recursive version is derived, where the new samples acquired during the test stage are utilized for updating the covariance of the process.

The paper is organized as follows. In Section II, we formulate the problem in a general noisy and reverberant environment. We discuss the existence of an acoustic manifold for each node and present the statistical model. A manifold-based Gaussian process is presented in Section III, and the relations between the nodes are defined. These definitions are unified by the multiple-manifold Gaussian process (MMGP) presented in Section III, which combines together the information from all the nodes. Based on this model a Bayesian estimator is derived in Section V. We present a recursive adaptation mechanism, and describe how to estimate the model parameters using an ML criterion. In Section VI, we demonstrate the algorithm performance by an extensive simulation study, and real-life recordings. Section VII concludes the paper.

II. Problem Formulation

A single source is located in a reverberant enclosure at position $\mathbf{q} = [q_x, q_y, q_z]^T$. Consider an ad hoc network with microphones distributed in the enclosure. We assume that the microphones are arranged in M nodes, where each node consists of a pair of microphones positioned side-by-side (up to half a meter distance). The source produces an unknown speech signal $s(t)$, which is measured by all the microphones. The signal received by the ith microphone of the mth pair, is given by:

$$y_i^m(t) = a_i^m(t, \mathbf{q}) * s(t) + u_i^m(t) \quad m = 1, \ldots, M; \quad i = 1, 2 \tag{1}$$

where $a_i^m(t, \mathbf{q})$ is the acoustic impulse response (AIR) relating the source at position $\mathbf{q}$ and the ith microphone in the mth node, and $u_i^m(t)$ is an additive noise signal, which contaminates the corresponding measured signal. Linear convolution is denoted by $*$.

Clearly, the information required for localization is embedded in the AIR and is independent of the source signal. Thus, from each pair of measurements we extract a feature vector $\mathbf{h}^m$ that depends solely on the two AIRs of the corresponding node and is independent of the non-stationary source signal. More specifically, we use a feature vector based on relative transfer function (RTF) estimates [37] in a certain frequency band, which is commonly used in acoustic array processing [37], [38]. Please refer to Appendix A for further details about the (RTF) and its estimation. The RTFs are typically represented in a high-dimensional space with a large number of coefficients to allow for the full description of the acoustic paths, which represent a complex reflection pattern. The observation that the RTFs are controlled by a small set of parameters, such as room dimensions, reverberation time, location of the source and the sensors etc., gives rise to the assumption that they are confined to a low dimensional manifold. In [39] and [33], we have shown that the RTFs of a certain node have a distinct structure. Hence, they are not uniformly distributed in the entire space, but rather pertain to a manifold $\mathcal{M}_m$ of much lower dimensions.

We define the function $f_a^m : \mathcal{M}_m \rightarrow \mathbb{R} \quad a \in \{x, y, z\}$ which maps an RTF sample $\mathbf{h}^m$ associated with the mth node to the corresponding x, y or z coordinate of the source position $f_a^m(\mathbf{h}^m)$. In the following derivation the three coordinates are estimated individually. Further justification for a separate treatment for each coordinate is discussed in Section VI-C. Since the same estimation is used for each coordinate, the axis notation is omitted henceforth. Let $p_i^m \equiv f^m(\mathbf{h}_i^m)$ denote the position evaluated by the function f^m for the RTF sample $\mathbf{h}_i^m$, where i is a sample index referring to a certain position. In this notation, the superscript denotes association to a certain node, and the subscript denotes association to a certain position index. Note that although the position of the source does not depend on the specific node, the notation p_i^m is used to express that the mapping is obtained from the measurements of the mth node.

The mth RTF represents the reflection pattern originating from the source and received by the mth node. Assuming that the different nodes are scattered over the room area, they experience a distinct reflection pattern, which differs from that experienced by other nodes. Each RTF $\mathbf{h}^m$ represents a different view point on the same acoustic event of a source speaking at some location in the enclosure. A particular node may have an accurate view of certain regions in the room and yet lacking on others. For example, closer distances are better viewed, while remote positions are not well distinguished. The view point of each node is reflected by the manifold $\mathcal{M}_m$ whose structure represents the relations between different RTFs, as they are inspected by that node. Combining the information from the different nodes may therefore increase the spatial separation and improve the ability to accurately locate the source. The central issue is then how to fuse the information provided by each of the M nodes to achieve this goal.

Let $\mathbf{h} = \left[[\mathbf{h}^1]^T, \ldots, [\mathbf{h}^M]^T\right]^T$ denote the aggregated RTF (aRTF), which is a concatenation of the RTF vectors from every node. We define the scalar function $f : \cup_{m=1}^M \mathcal{M}_m \rightarrow \mathbb{R}$ which attaches an aRTF sample $\mathbf{h}_i$ with the corresponding x, y or z coordinate of the source position $p_i \equiv f(\mathbf{h}_i)$. In the first step, we discuss each node and its mapping function f^m, and then we combine the different views in the definition of the function f.

In a fixed acoustic environment, the function f^m that relates $\mathbf{h}_i^m$ to its position p_i^m (which is a scalar since it represents the x, y or z coordinate of the position), is deterministic, in the sense that a certain reflection pattern expressed by the mth RTF is exclusively associated with a certain position. However, even when all the environmental parameters are fixed and known, there is no simple model that links a given RTF sample to its position. Hence, we use the statistical model presented in [34]. An RTF $\mathbf{h}_i^m$ is assumed to be sampled from the manifold $\mathcal{M}_m$. The RTF sample $\mathbf{h}_i^m$ is related by the function f^m to the corresponding position p_i^m. We assume that p^m follows a Gaussian process, as will be described in Section III. A nomenclature listing the different symbols and their meanings is given in Table I.

TABLE I
NOMENCLATURE

$\mathbf{h}_i^m$	an RTF sample of the m th node associated with position $\mathbf{p}_i$
$\mathbf{h}_i$	an aggregated RTF (aRTF) sample associated with position $\mathbf{p}_i$, consisting of RTFs of all M nodes
p_i^m	a position associated with $\mathbf{h}_i^m$, drawn from the Gaussian process p^m of the m th node
p_i	a position associated with $\mathbf{h}_i$, drawn from the Gaussian process p
$\mathcal{M}_m$	the manifold associated with RTFs of the mth node

The estimation is semi-supervised and is based on a training set of aRTF samples associated with various source positions, measured in advance. However, the microphone positions may be unknown since they are not required for the estimation. The training set consists of two subsets: a small subset of aRTF samples with 'labels', i.e. with known source positions, and a large subset of aRTF samples without labels, i.e., with unknown source locations. The first subset consists of n_L labelled samples, denoted by $H_L = \{\mathbf{h}_i\}_{i=1}^{n_L}$, and their associated measured positions $\{\bar{p}_i\}_{i=1}^{n_L}$. The labelled positions are marked by bars since they may slightly differ from the actual positions due to imperfections in the measurements. Note also that though all three coordinates of the position are measured for each labelled sample, P_L is defined as a collection of scalars (associated with a certain coordinate) rather than vectors, since the same derivation applies separately to each coordinate. The second subset consists of n_U unlabelled samples, denoted by $H_U = \{\mathbf{h}_i\}_{i=n_L+1}^{n_D}$, where $n_D = n_L + n_U$. The entire training set consists of n_D aRTF samples and is denoted by $H_D = H_L \cup H_U$. In the test stage, we receive a new set $H_T = \{\mathbf{h}_i\}_{i=n_D+1}^{n}$ of n_T new aRTF samples from unknown locations, where $n = n_D + n_T$. The entire set, including both the training and the test samples, is denoted by $H = H_D \cup H_T$.

III. Manifold-Based Gaussian Process

We first present the statistical model for each node individually, and then discuss the relations between different nodes. Finally, we define the function f that combines the data from all the nodes in a way that respects both the intra-relations within each node and the inter-relations between the different nodes.

We assume that p^m follows a Gaussian process, i.e. any finite set of positions associated with RTFs of the mth node, are jointly distributed Gaussian variables. The Gaussian process is a convenient choice since it is entirely defined by its second order statistics, and is widely used for regression problems [40]. We use a zero-mean Gaussian process for simplicity. By setting the origin to the middle of the enclosure of interest, the zero-mean assumption reflects that all possible source positions are distributed around the origin. The covariance function is a pairwise affinity measure between two RTF samples. We suggest to use a manifold-based covariance function, in which the relation between two RTFs is not only a function of the current samples, but also uses the information from the entire available set of RTF samples:

$$\begin{aligned}\text{cov}(p_r^m, p_l^m) &\equiv \sum_{i=1}^{n_D} k_m(\mathbf{h}_r^m, \mathbf{h}_i^m) k_m(\mathbf{h}_l^m, \mathbf{h}_i^m) \\ &= 2k_m(\mathbf{h}_r^m, \mathbf{h}_l^m) + \sum_{\substack{i=1 \\ i\neq l,r}}^{n_D} k_m(\mathbf{h}_r^m, \mathbf{h}_i^m) k_m(\mathbf{h}_l^m, \mathbf{h}_i^m) \end{aligned} \quad (2)$$

where l and r represent ascription to certain positions, and k_m is a standard pairwise function $k_m : \mathcal{M}_m \times \mathcal{M}_m \longrightarrow \mathbb{R}$, often termed "kernel function". The equality in (2) holds for kernels that satisfy: $k_m(\mathbf{h}_i^m, \mathbf{h}_j^m) = 1$ for $i = j$. A common choice is to use a Gaussian kernel, with a scaling factor ε_m:

$$k_m(\mathbf{h}_i^m, \mathbf{h}_j^m) = \exp\left\{-\frac{\|\mathbf{h}_i^m - \mathbf{h}_j^m\|^2}{\varepsilon_m}\right\}. \quad (3)$$

The definition of the covariance in (2), induces a new type of manifold-based kernel $\tilde{k}_m$:

$$\tilde{k}_m(\mathbf{h}_r^m, \mathbf{h}_l^m) \equiv \text{cov}(p_r^m, p_l^m) \quad (4)$$

In [34] we adopted the manifold-based kernel proposed by Sindhwani *et al.* [36]. Here, we propose another type of kernel, which is more convenient for estimating the model hyperparameters and for deriving a recursive adaptation mechanism. A similar kernel was used to define a graph-based diffusion filter in [41], and was applied in a patch-based de-noising algorithm in [42]. The Euclidean distance between the high-dimensional RTFs, used in the standard kernel $k_m(\mathbf{h}_r^m, \mathbf{h}_l^m)$, does not properly reflect their distance with respect to the manifold $\mathcal{M}_m$ [39]. The new kernel $\tilde{k}_m(\mathbf{h}_r^m, \mathbf{h}_l^m)$ is based on $\{k_m(\mathbf{h}_l^m, \mathbf{h}_i^m)\}_{i=1}^{n_D}$ and $\{k_m(\mathbf{h}_r^m, \mathbf{h}_i^m)\}_{i=1}^{n_D}$, which represent the relations between each sample to all the training samples. The covariance in (2) between $\mathbf{h}_l^m$ and $\mathbf{h}_r^m$ is evaluated by the correlation between all the inspected relations, namely between $\{k(\mathbf{h}_l^m, \mathbf{h}_i^m)\}_{i=1}^{n_D}$ and $\{k(\mathbf{h}_r^m, \mathbf{h}_i^m)\}_{i=1}^{n_D}$. In this formulation, the covariance is determined according to the extent of correspondence between the mutual relations of $\mathbf{h}_l^m$ and $\mathbf{h}_r^m$ to other samples on the manifold. When both samples have similar relations to other samples, it indicates that they are closely related, and the value of $\tilde{k}(\mathbf{h}_r^m, \mathbf{h}_l^m)$ increases respectively.

We also define the relation between the functions of different nodes q and w, evaluated for two RTF samples associated with different source positions. Namely, we define the relation between p_r^q and p_l^w for $1 \leq l, r \leq n_D$. We assume that p_r^q and p_l^w are jointly distributed Gaussian variables and that their covariance is defined by:

$$\text{cov}(p_r^q, p_l^w) \equiv \tilde{k}_{qw}(\mathbf{h}_r^q, \mathbf{h}_l^w) = \sum_{i=1}^{n_D} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q) k_w(\mathbf{h}_l^w, \mathbf{h}_i^w). \quad (5)$$

It is important to note that when examining the relation between functions evaluated for different nodes, we cannot directly compute the distance between the corresponding RTF samples since they represent different views. In (5), we examine the intra-relations $\{k_q(\mathbf{h}_r^q, \mathbf{h}_i^q)\}_{i=1}^{n_D}$ in the qth manifold and the intra-relations $\{k_w(\mathbf{h}_r^w, \mathbf{h}_i^w)\}_{i=1}^{n_D}$ in the wth manifold. The inter-relations between $\mathbf{h}_r^q$ and $\mathbf{h}_l^w$ are evaluated by the correlation between the relations formed on each manifold individually. The covariance defined in (5) emphasizes similar relations observed by both nodes, and discard relations observed by only one of the nodes. An illustration of the inter-relation between the two manifolds is illustrated in Fig. 1. Note that the single node relation (2) can be considered as a special case of the multi-node relation (5).

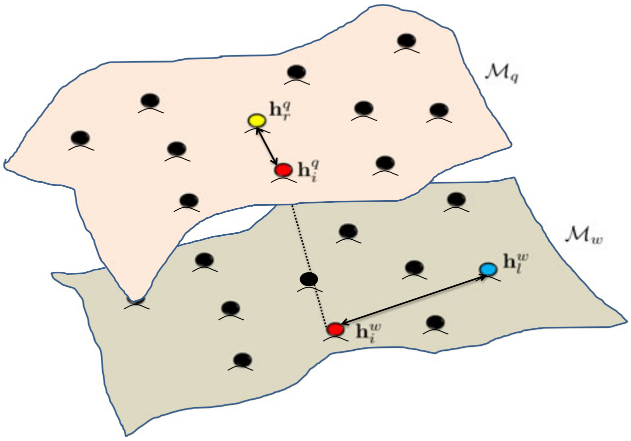


Fig. 1. An illustration of the covariance computation for RTF samples of different nodes q and w.

IV. Multinode Data Fusion

So far, we have presented the statistical model and defined a Gaussian process p_i^m for each node. In addition, we have defined the covariance of each individual process of a particular node (2) and the cross-covariance between two processes of two different nodes (5). Our goal is to unify these definitions under one statistical umbrella which combines the information provided by the different pairs and establishes a foundation for deriving a Bayesian estimator for the source position.

A. Multiple-Manifold Gaussian Process

To fuse the different perspectives presented by the different nodes, we define the multiple-manifold Gaussian process (MMGP) p as the mean of the Gaussian processes of all the nodes, i.e. each position p_i drawn from the process is given by:

$$p_i = \frac{1}{M}\left(p_i^1 + p_i^2 + \cdots + p_i^M\right). \quad (6)$$

Due to the assumption that the processes are jointly Gaussian, the process p is also Gaussian with zero-mean and covariance function given by:

$$\begin{aligned}\text{cov}(p_r, p_l) &= \frac{1}{M^2}\text{cov}\left(\sum_{q=1}^{M} p_r^q, \sum_{w=1}^{M} p_l^w\right) \\ &= \frac{1}{M^2}\sum_{q,w=1}^{M} \text{cov}(p_r^q, p_l^w). \end{aligned} \quad (7)$$

Using the definitions of (2) and (5) we obtain the covariance for p_r and p_l:

$$\begin{aligned}\operatorname{cov}(p_r, p_l) &\equiv \tilde{k}(\mathbf{h}_r, \mathbf{h}_l)\\ &= \frac{1}{M^2}\sum_{i=1}^{n_D}\sum_{q,w=1}^{M} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q)k_w(\mathbf{h}_l^w, \mathbf{h}_i^w). \qquad (8)\end{aligned}$$

Here, the covariance, evaluated for two samples from the process p, is determined using all M^2 relations between the different nodes and by averaging over all the samples in H_D. Regarding computational complexity, note that due to symmetry, some terms in (8) are equal when q and w are swapped, and that $k(\mathbf{h}_i, \mathbf{h}_j) = 1$ for $i = j$. The covariance in (8), consists of all inter-relations between the different nodes, enhancing observations which are common to pairs of nodes, and ignoring relations that appear in only one node. Through the lens of kernel-based learning, $\tilde{k}(\mathbf{h}_r, \mathbf{h}_l)$ can be considered as a *composition of kernels*, which, in addition to connections acquired in each node separately, incorporates the extra spatial information in the mutual relationship between RTFs of different nodes. This formulation represents a robust measurement of correlation by utilizing multiple view-points of the same acoustic scene, aiming to improve the localization capabilities.

The resulting Gaussian process is zero-mean with covariance function $\tilde{k}$:

$$p \sim \mathcal{GP}(0, \tilde{k}). \qquad (9)$$

Accordingly, the random vector $\mathbf{p}_H = [p_1, \ldots, p_n]^T$, which consists of n samples from the process p, has a multivariate Gaussian distribution, i.e.,

$$\mathbf{p}_H \sim \mathcal{N}(\mathbf{0}_n, \tilde{\mathbf{\Sigma}}_H) \qquad (10)$$

where $\mathbf{0}_n$ is an $n \times 1$ vector of all zeros and $\tilde{\mathbf{\Sigma}}_H$ is the covariance matrix with elements $\tilde{k}(\mathbf{h}_i, \mathbf{h}_j),\ \mathbf{h}_i, \mathbf{h}_j \in H$. Note that the covariance matrix $\tilde{\mathbf{\Sigma}}_H$ can be expressed in terms of the covariance matrices of all the individual nodes $\mathbf{K}_H^m$, defined by the standard kernel $(\mathbf{K}_H^m)_{ij} = k_m(\mathbf{h}_i^m, \mathbf{h}_j^m)$ of (3):

$$\tilde{\mathbf{\Sigma}}_H = \frac{1}{M^2}\sum_{q,w=1}^{M} \mathbf{K}_H^q \mathbf{K}_H^w. \qquad (11)$$

In this representation, the covariance matrix for any finite set of samples from the process is computed by a sum of all pairwise multiplications between the covariance matrices of each of the nodes.

B. Alternating Diffusion Interpretation

Before we proceed to the derivation of the estimation procedure, which is based on these definitions, we present an alternative interpretation using a geometrical perspective from the field of diffusion maps [43]. Specifically, we provide an interpretation for the definitions of the covariance functions in (5) and (8). As discussed above, every node represents a different view point, which is realized by the structure of the associated manifold $\mathcal{M}_m$. We can create a discrete representation of the mth manifold by a graph G^m in which the vertices represent the RTF samples of the mth node and the weights connecting between them are stored in the matrix $\mathbf{K}_H^m$. This way, we obtain M graphs with matching vertices that are associated with the same positions, but with different weighted edges determined by the distances between the samples within each separate node. In [44], the authors defined an alternating diffusion operator, which constitutes a combined graph G^{qw}, where the weight matrix is given by $\mathbf{K}_H^{qw} \equiv \mathbf{K}_H^q \mathbf{K}_H^w$. They have shown that the Markov process defined on the resulting graph extracts the underlying source of variability common to the two graphs q and w (related to the microphone nodes q and w).

In our case, an RTF is closely related to its associated position, however it may be influenced by other factors as well, such as estimation errors and noise. We assume that the interferences introduced by a particular node differ from the ones introduced by the other nodes. When measuring the correlation between two nodes, we would like to emphasize the common source of variability, namely the source position, and to suppress artifacts and interferences, which are node-specific effects. By multiplying the kernels of each two nodes as indicated in (11), we average out incoherent node-specific variables and remain only with the common variable, which is the position of the source. This perspective provides a justification to the averaging over different nodes as well as over different samples, constituting a robust measure of correlation between samples in terms of the physical proximity between the corresponding source positions.

V. Bayesian Inference With Multiple-Manifold Gaussian Process

In the previous section we presented the MMGP p that relates aRTF samples to the corresponding source positions. We have shown that the covariance of the process depends on both the internal relations within the same manifold (same node) and the pairwise connections between different manifolds (different nodes). Note that the covariance function of the process (8) is based only on the aRTF samples in H_D, and does not take into account the labellings. The information implied by the labelled samples H_L and their associated labels P_L is used to update our prior belief about the behaviour of the process p, and to derive its posterior distribution. The pairs $\{\mathbf{h}_i, \bar{p}_i\}_{i=1}^{n_L}$ serve as anchor points utilized for interpolating a realization of the process p, while the Gaussian process assumption in (9) is designed to ensure the smoothness of the solution.

A. Localization With Multiple-Manifold Gaussian Process

Following the statistical model stated in Section II, we assume that the measured positions $P_L = \{\bar{p}_i\}_{i=1}^{n_L}$ of the labelled set arise from a noisy observation model, given by:

$$\bar{p}_i = p_i + \eta_i;\ i = 1, \ldots, n_L \qquad (12)$$

where $\eta_i \sim \mathcal{N}(0, \sigma^2)\ i = 1, \ldots, n_L$ are i.i.d. Gaussian noises, independent of p_i. The noise in (12) reflects uncertainties due to imperfect measurements of the source positions while acquiring the labelled set. Note that since the Gaussian variables p_i and η_i are independent, they are jointly Gaussian. Consequently, p_i and $\bar{p}_i$ are also jointly Gaussian. We define the likelihood of the process p based on the probability of the labelled examples:

$$\Pr(P_L|p, H_L) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n_L}(\bar{p}_i - p_i)^2\right\}. \quad (13)$$

To perform localization, we are interested in estimating the position of a new test sample $\mathbf{h}_t \in H_T$ of an unknown source from an unknown location. The estimation is based on the posterior probability $\Pr(p_t \equiv f(\mathbf{h}_t)|P_L, H_L)$. According to (10) and (13), the function value at the test point p_t and the concatenation of all labelled training positions $\mathbf{p}_L = \mathrm{vec}\{P_L\} \equiv [\bar{p}_1, \ldots, \bar{p}_{n_L}]^T$ are jointly Gaussian, with:

$$\begin{bmatrix}\mathbf{p}_L \\ p_t\end{bmatrix}\Bigg| H_L \sim \mathcal{N}\left(\mathbf{0}_{n_L+1}, \begin{bmatrix}\tilde{\mathbf{\Sigma}}_L + \sigma^2\mathbf{I}_{n_L} & \tilde{\mathbf{\Sigma}}_{Lt} \\ \tilde{\mathbf{\Sigma}}_{Lt}^T & \tilde{\Sigma}_t\end{bmatrix}\right) \quad (14)$$

where $\tilde{\mathbf{\Sigma}}_L$ is an $n_L \times n_L$ covariance matrix defined over the function values at the labelled samples H_L, $\tilde{\mathbf{\Sigma}}_{Lt}$ is an $n_L \times 1$ covariance vector between the function values at H_L and p_t, $\tilde{\Sigma}_t$ is the variance of p_t, and $\mathbf{I}_{n_L}$ is the $n_L \times n_L$ identity matrix. This implies that the conditional distribution $\Pr(p_t|P_L, H_L)$ is a multivariate Gaussian with μ_{cond} mean and σ^2_{cond} variance given by:

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

$$\sigma^2_{\text{cond}} = \tilde{\Sigma}_t - \tilde{\mathbf{\Sigma}}_{Lt}^T\left(\tilde{\mathbf{\Sigma}}_L + \sigma^2\mathbf{I}_{n_L}\right)^{-1}\tilde{\mathbf{\Sigma}}_{Lt}. \quad (15)$$

Hence, the maximum a posteriori probability (MAP) estimator of p_t, which coincides with the minimum mean squared error (MMSE) estimator in the Gaussian case, is given by:

$$\hat{p}_t = \mu_{\text{cond}} = \tilde{\mathbf{\Sigma}}_{Lt}^T\tilde{\mathbf{p}}_L \quad (16)$$

where $\tilde{\mathbf{p}}_L \equiv \mathbf{\Gamma}_L\mathbf{p}_L$ is a vector of weights which are independent of the current test sample, and $\mathbf{\Gamma}_L = \left(\tilde{\mathbf{\Sigma}}_L + \sigma^2\mathbf{I}_{n_L}\right)^{-1}$. Note that the estimator in (16) is obtained as a linear combination of the kernel $\tilde{k}$ evaluated for the test sample $\mathbf{h}_t$ and each of the labelled samples H_L, weighted by the entries of $\tilde{\mathbf{p}}_L$. Note that the posterior is defined only with respect to the labelled samples, hence the covariance terms are calculated based solely on the labelled samples H_L, without taking into account the samples in the set H_U as was defined in general in the previous section. Although the unlabelled samples do not appear explicitly in (16), they take role in the computation of the correlation terms as implied by (8). In fact, the unlabelled samples are essential both for obtaining a more accurate computation of the weights $\tilde{\mathbf{p}}_L$, and for better quantifying the relations between the current test sample and each of the labelled samples. The variance of the estimator is given by σ^2_{cond} in (15). It can be seen that the posterior variance σ^2_{cond} is smaller than the prior variance $\tilde{\Sigma}_t$, indicating that the labelled examples reduce the uncertainty in the behaviour of the Gaussian process. The variance of the estimator is smaller for test samples which are close to a large number of labelled samples, increasing the second term in (15), and therefore decreasing the overall variance. The estimation is more reliable in regions where the labelled samples are dense, and becomes more uncertain in sparse regions.

B. Recursive Algorithm

In this section, we develop a recursive version for the estimator in (16). The Gaussian process is adapted by the information provided by new (streaming) RTF samples, in the test stage. Any new RTF sample $\mathbf{h}_t$ can be considered as an additional unlabelled sample, hence, can be used to update the covariance terms in (2) and (5). Considering also the new sample, the covariance is given by an average of $n_D + 1$ kernel values for all the training set and the current test sample. Accordingly, the covariance in (8) for two labelled samples $1 \leq l, r \leq n_L$, is updated by:

$$\tilde{k}^*(\mathbf{h}_r, \mathbf{h}_l) = \frac{1}{M^2}\sum_{q,w=1}^{M}\left(\underbrace{\sum_{i=1}^{n_D} k_q(\mathbf{h}_r^q, \mathbf{h}_i^q)k_w(\mathbf{h}_l^w, \mathbf{h}_i^w)}_{\text{training}} + \underbrace{k_q(\mathbf{h}_r^q, \mathbf{h}_t^q)k_w(\mathbf{h}_l^w, \mathbf{h}_t^w)}_{\text{new test sample}}\right)$$

$$= \tilde{k}(\mathbf{h}_r, \mathbf{h}_l) + \frac{1}{M^2}\left(\sum_{q=1}^{M} k_q(\mathbf{h}_r^q, \mathbf{h}_t^q)\right) \times \left(\sum_{w=1}^{M} k_w(\mathbf{h}_l^w, \mathbf{h}_t^w)\right) \quad (17)$$

where * stands for an updated term. Similarly, for kernels satisfying $k_m(\mathbf{h}_i^m, \mathbf{h}_j^m) = 1$ for $i = j$, the covariance in (8), when measured between the new test sample $\mathbf{h}_t$ and a labelled sample $\mathbf{h}_l, 1 \leq l \leq n_L$, is given by:

$$\tilde{k}^*(\mathbf{h}_t, \mathbf{h}_l) = \tilde{k}(\mathbf{h}_t, \mathbf{h}_l) + \frac{1}{M}\sum_{q=1}^{M} k_q(\mathbf{h}_l^q, \mathbf{h}_t^q) \quad (18)$$

According to (17) and (18), the updated forms of the covariance matrix $\tilde{\mathbf{\Sigma}}_L$ and of the covariance vector $\tilde{\mathbf{\Sigma}}_{Lt}$, are given by:

$$\tilde{\mathbf{\Sigma}}_L^* = \tilde{\mathbf{\Sigma}}_L + \frac{1}{M^2}\mathbf{k}_{Lt}\mathbf{k}_{Lt}^T$$

$$\tilde{\mathbf{\Sigma}}_{Lt}^* = \tilde{\mathbf{\Sigma}}_{Lt} + \frac{1}{M}\mathbf{k}_{Lt} \quad (19)$$

where $\mathbf{k}_{Lt} = \left[\sum_{q=1}^{M} k_q(\mathbf{h}_1^q, \mathbf{h}_t^q), \ldots, \sum_{q=1}^{M} k_q(\mathbf{h}_{n_L}^q, \mathbf{h}_t^q)\right]^T$. Using the Woodbury matrix identity [45] and (19), we obtain the adaptation rule for $\mathbf{\Gamma}_L = \left(\tilde{\mathbf{\Sigma}}_L + \sigma^2\mathbf{I}_{n_L}\right)^{-1}$:

$$\mathbf{\Gamma}_L^* = \left(\mathbf{\Gamma}_L^{-1} + \frac{1}{M^2}\mathbf{k}_{Lt}\mathbf{k}_{Lt}^T\right)^{-1}$$

$$= \mathbf{\Gamma}_L - \frac{\mathbf{\Gamma}_L\mathbf{k}_{Lt}\mathbf{k}_{Lt}^T\mathbf{\Gamma}_L}{M^2 + \mathbf{k}_{Lt}^T\mathbf{\Gamma}_L\mathbf{k}_{Lt}} \quad (20)$$

Hence, the updated weights are $\tilde{\mathbf{p}}_L^* = \mathbf{\Gamma}_L^*\mathbf{p}_L$, and the estimated position is given by:

$$\hat{p}_t = \tilde{\mathbf{\Sigma}}_{Lt}^{*T}\tilde{\mathbf{p}}_L^*. \quad (21)$$

C. Learning the Hyperparameters

The zero-mean Gaussian process model is fully specified by its covariance function. Thus, the predictions obtained by this model depend on the chosen covariance function. In practice, we use a parametric family of functions, i.e. a Gaussian kernel as in (3) with a scaling-parameter ε_m. The values of the parameters $\{\varepsilon_m\}_{m=1}^{M}$ can be inferred from the data by optimizing the likelihood function of the labelled samples. From the distribution defined in (14), the log-likelihood function of the labelled samples get the form of a multivariate Gaussian distribution, given by:

$$\begin{aligned} L = \ln \Pr(\mathbf{p}_L | H_L; \Theta) &= -\frac{1}{2}\mathbf{p}_L^T \left(\tilde{\boldsymbol{\Sigma}}_L + \sigma^2 \mathbf{I}_{n_L}\right)^{-1} \mathbf{p}_L \\ &- \frac{1}{2}\ln\left|\tilde{\boldsymbol{\Sigma}}_L + \sigma^2\mathbf{I}_{n_L}\right| - \frac{n_L}{2}\ln(2\pi), \end{aligned} \tag{22}$$

where Θ denotes the set of model parameters. In (22), the first term measures how well the parameters fit the given labelled samples, and the second term reflects the model complexity, which is evaluated through the determinant of the covariance matrix. The optimization requires the computation of the gradients of the log-likelihood function with respect to each of the parameters. The partial derivative with respect to ε_m can be generally expressed by (see [40, Ch. 5]):

$$\begin{aligned} \frac{\partial L}{\partial \varepsilon_m} &= -\frac{1}{2}\text{trace}\left\{\boldsymbol{\Gamma}_L \frac{\partial \tilde{\boldsymbol{\Sigma}}_L}{\partial \varepsilon_m}\right\} + \frac{1}{2}\mathbf{p}_L^T \boldsymbol{\Gamma}_L \frac{\partial \tilde{\boldsymbol{\Sigma}}_L}{\partial \varepsilon_m} \boldsymbol{\Gamma}_L \mathbf{p}_L \\ &= \frac{1}{2}\text{trace}\left\{\left[(\boldsymbol{\Gamma}_L\mathbf{p}_L)(\boldsymbol{\Gamma}_L\mathbf{p}_L)^T - \boldsymbol{\Gamma}_L\right]\frac{\partial \tilde{\boldsymbol{\Sigma}}_L}{\partial \varepsilon_m}\right\} \end{aligned} \tag{23}$$

where the partial derivative of $\tilde{\boldsymbol{\Sigma}}_L$ in (23) with respect to each ε_m, is given by:

$$\begin{aligned} M^2 \frac{\partial \tilde{\boldsymbol{\Sigma}}_L}{\partial \varepsilon_m} &= \frac{\partial\left(\sum_{q,w=1}^{M} \mathbf{K}_L^q \mathbf{K}_L^w\right)}{\partial \varepsilon_m} \\ &= \frac{\partial \mathbf{K}_L^m}{\partial \varepsilon_m}\left(\sum_{q=1}^{M}\mathbf{K}_L^q\right) + \left(\sum_{q=1}^{M}\mathbf{K}_L^q\right)\frac{\partial \mathbf{K}_L^m}{\partial \varepsilon_m} \end{aligned} \tag{24}$$

where $\frac{\partial \mathbf{K}_L^m}{\partial \varepsilon_w}$ is an $n_L \times n_L$ matrix with (i,j)th entry given by $\frac{\|\mathbf{h}_i - \mathbf{h}_j\|^2}{\varepsilon_m^2}\exp\left\{-\frac{\|\mathbf{h}_i - \mathbf{h}_j\|^2}{\varepsilon_m}\right\}$.

Similarly, we can also estimate the optimal value for the variance σ^2 of the observation noise. The partial derivative with respect to σ^2 has similar form to (23):

$$\frac{\partial L}{\partial \sigma^2} = \frac{1}{2}\text{trace}\left\{(\boldsymbol{\Gamma}_L\mathbf{p}_L)(\boldsymbol{\Gamma}_L\mathbf{p}_L)^T - \boldsymbol{\Gamma}_L\right\}. \tag{25}$$

Based on (23)–(25), Eq. (22) can be optimized using an efficient gradient-based optimization algorithm. It should be noted that the parameter values are optimized through the likelihood of the labelled set, hence, optimality for the test samples cannot be guaranteed. This optimization can serve as an initialization for the parameter values, which may then be fine-tuned by other prevailing methods, such as cross-validation. A flow diagram of the entire algorithm is illustrated in Fig. 2.

D. Computational Complexity

In this section we analyse the computational complexity of the proposed method. The major factors that influence the complexity of the implementation are: the number of training samples $n_D = n_L + n_U$, the number of nodes M, the window length N, the number of frequency bins D, and the number of time frames for each measurement T. For simplicity, we equally weight multiplications, divisions, additions, subtractions and exponentiations. We list the number of operations required for each step in the algorithm. Note that the operations in training phase are performed in advance only once, while the operations in the test phase are performed for each test sample.

Training Phase:

1) *RTF estimation:* The estimation of each RTF requires $\mathcal{O}\left(N^2\log_2(N)T\right)$ operations. We estimate the RTF for each training measurement with respect to each node, hence the estimation of all the training RTF samples requires $\mathcal{O}\left(N^2\log_2(N)TMn_D\right)$ operations.
2) *Covariance estimation:* The estimation of the position in the test phase by either (16) or (21), depends on $\boldsymbol{\Gamma}_L = (\boldsymbol{\Sigma}_L + \sigma^2\mathbf{I}_{n_L})^{-1}$, which can be computed in advance. First, we need to evaluate the kernel $k_m(\mathbf{h}_i^m, \mathbf{h}_j^m)$ for all $1 \le i \le n_L$, $1 \le j \le n_D$, $1 \le m \le M$, which requires $\mathcal{O}(DMn_Ln_D)$ operations. Second, we need to evaluate the kernel $\tilde{k}(\mathbf{h}_i, \mathbf{h}_j)$ in (8) for all $1 \le i, j \le n_L$, which requires $\mathcal{O}\left(M^2n_Dn_L^2\right)$ operations. The inversion of the matrix $\boldsymbol{\Sigma}_L + \sigma^2\mathbf{I}_{n_L}$ requires $\mathcal{O}\left(n_L^3\right)$ operations.

Hence, the total number of operations in the training phase is given by:

$$\begin{aligned} \text{CMP}_{\text{tr}} = \mathcal{O}\Big(&N^2\log_2(N)TMn_D + DMn_Ln_D \\ &+ M^2n_L^2n_D + n_L^3\Big) \end{aligned} \tag{26}$$

Test Phase:

1) *RTF estimation:* The estimation of the test RTFs with respect to each node requires $\mathcal{O}\left(N^2\log_2(N)TM\right)$ operations.
2) *Covariance estimation:* In order to compute the covariance between the test sample and the labelled samples $\boldsymbol{\Sigma}_{Lt}$, we first need to evaluate the kernel $k_m(\mathbf{h}_i^m, \mathbf{h}_t^m)$ for all $1 \le i \le n_D$, $1 \le m \le M$, which requires $\mathcal{O}(DMn_D)$ operations. Next, we need to evaluate the kernel $\tilde{k}(\mathbf{h}_t, \mathbf{h}_i)$ in (8) for all $1 \le i \le n_L$, which requires $\mathcal{O}\left(M^2n_Ln_D\right)$ operations.
3) *Adaptation:* The adaptation of $\boldsymbol{\Gamma}_L$ in (20) requires $\mathcal{O}\left(n_L^2\right)$ operations.
4) *Position estimation:* The estimation of the position by either (16) or (21) requires $\mathcal{O}\left(n_L^2\right)$ operations.

Hence, the total number of operations in the test phase is given by:

$$\text{CMP}_{\text{ts}} = \mathcal{O}\Big(N^2\log_2(N)TM + DMn_D + M^2n_Ln_D + n_L^2\Big) \tag{27}$$

It should be noted that in both the training and the test stages, the complexity is dominated by the RTF estimation, i.e. $\text{CMP}_{\text{tr}} \approx \mathcal{O}\left(N^2\log_2(N)TMn_D\right)$ and $\text{CMP}_{\text{ts}} \approx$

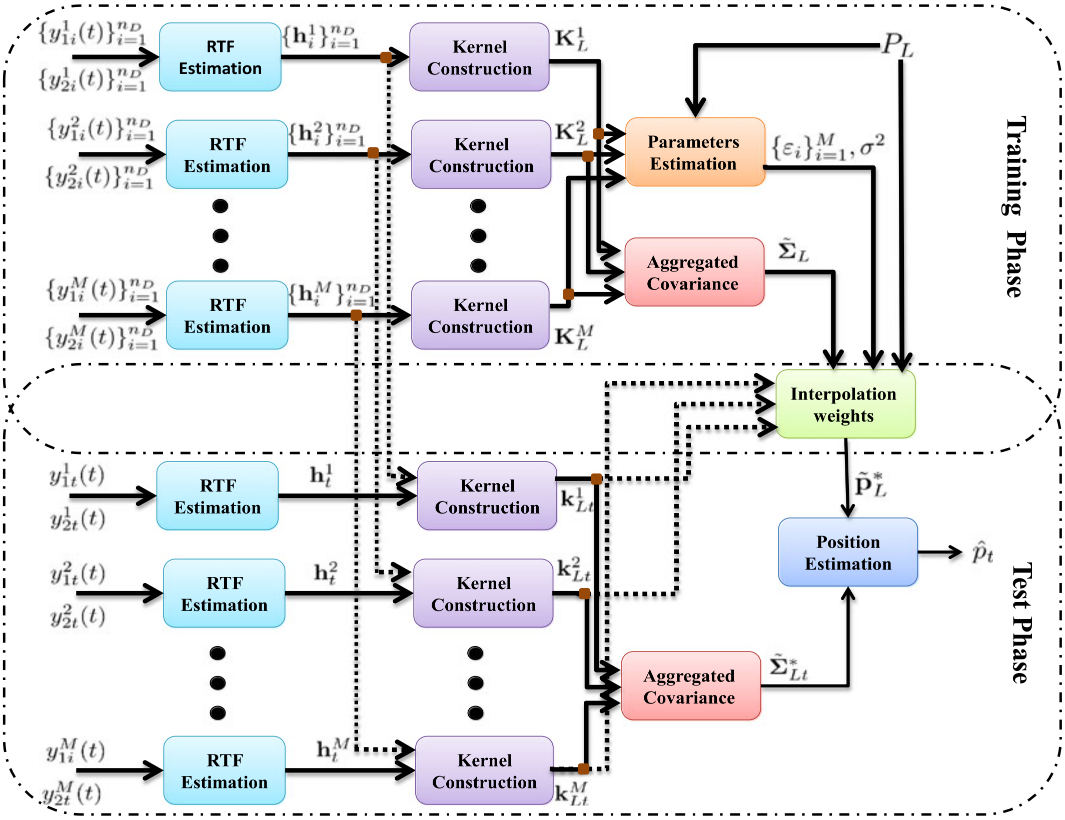


Fig. 2. A flow diagram of the proposed algorithm. The algorithm consists of a training phase (the upper part) and a test phase (the lower part). In the training phase, we estimate the RTFs for the training set, compute the covariance matrix of the labelled samples, and optimize the model parameters. In the test phase, we estimate the aRTF for the current sample, compute the covariance between the current sample and the labelled set, update the covariance terms, and estimate the position.

$\mathcal{O}\left(N^2 \log_2(N)TM\right)$. This part can be replaced if different acoustic features are used instead of RTFs. For demonstration, let: $n_L = 36$, $n_U = 100$, $M = 5$, $N = 2048$, $D = 291$ $T = 150$ (corresponding to 5s long signals). Using a Matlab implementation on a standard PC (CPU Intel Core2 Quad 3.7 GHz, RAM 8 GB) the training phase takes on average 67.21 s. The test phase takes on average 0.51 s per a single test sample of 5 s. For comparison, the SRP-PHAT implementation [46] on the same PC takes 0.44 s per test sample of 5 s.

VI. Experimental Results

In this section, we demonstrate the performance of the proposed method for localization of a single source in noisy and reverberant conditions. We focus on localization in both the x and the y coordinates of the source position, for a fixed height of the source. Further discussion on localization in all three coordinates appears in Section VI-C. The performance is evaluated using both simulated data and real-life recordings. The simulation is used to give a wide comparison of the effect of different noise and reverberation levels. However, the examination of real recordings is of great importance, since the simulation may not faithfully represent the physical phenomena encountered in real-life scenarios.

A. Simulation Results

We simulated a $5.2 \times 6.2 \times 3.5$ m room with different reverberation levels, using an efficient implementation [47] of the image method [48]. Six pairs of microphones were located around the room. The source positions were confined to a 2×2 m squared region, at 0.5 m distance from one of the room walls. The training set consisted of $n_L = 36$ labelled samples creating a grid with a resolution of 40 cm. In addition, there were $n_U = 100$ unlabelled measurements from unknown locations in the same region. The room setup and the positions of the training set are illustrated in Fig. 3. For each position, we simulated a source uttering a WGN signal for the labelled points and a speech signal for the unlabelled points. The algorithm was tested on $n_T = 200$ measurements of unknown sources from unknown locations with unique speech signals. All the measurements were 5 s long, and were contaminated by additive WGN. For each point, the cross PSD (CPSD) and the PSD were estimated with Welch's method with 0.128 s windows and 75% overlap, and were utilized for estimating the RTF in (30) for 2048 frequency bins. The RTF vector consisted of $D = 291$ frequency bins corresponding approximately to 100–2400 Hz, in which most of the speech components are concentrated (for details please refer to Appendix A).

For the proposed method we used (21) to update the model according to the current test sample, i.e. for each test point the

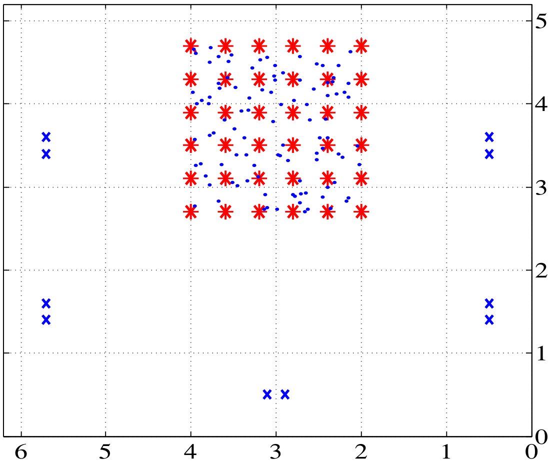


Fig. 3. The simulated room setup. The blue x-marks denote the microphones, the red asterisks denote the labelled samples and the blue dots denote the unlabelled samples.

correlation was obtained by an average of $n_D + 1$ points (the entire training set and the current test point). For comparison, we also examined the performance of two other algorithms which, although based on manifold considerations, heuristically fuse the data from the nodes. Both algorithms rely on the manifold-based Gaussian process regression described in [34]. The first approach ("Mean" in the graph) simply averages the estimates obtained by each single node separately. The second algorithm ("Kernel-Mult" in the graph) uses a Gaussian process with a covariance function that is given by the product of the individual kernels of the single nodes (3). For a Gaussian kernel, using the product between the kernels of the different nodes is identical to using the aRTF as an input to the kernel, i.e.

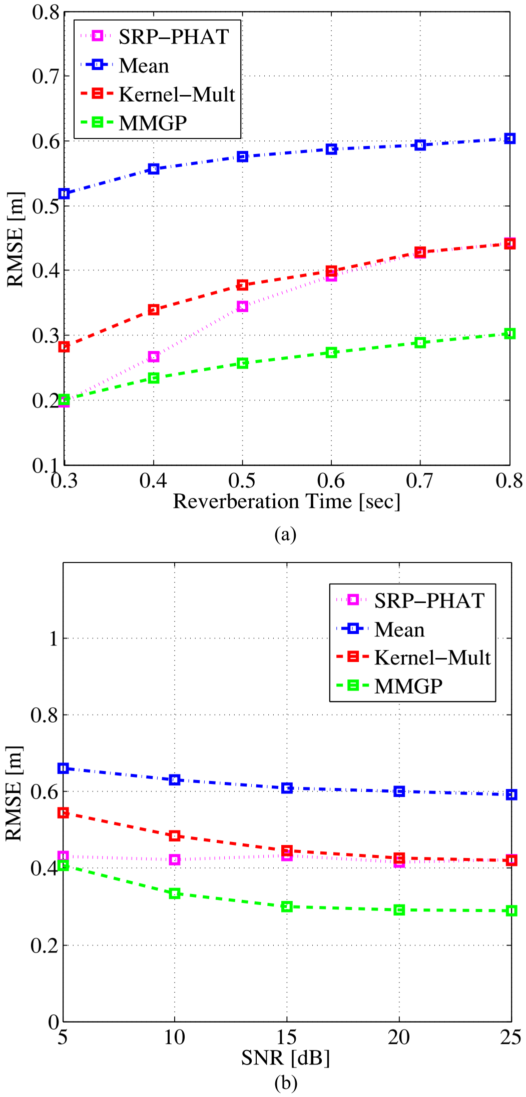
$$k(\mathbf{h}_i, \mathbf{h}_j) = k(\mathbf{h}_i^1, \mathbf{h}_j^1) \cdot k(\mathbf{h}_i^2, \mathbf{h}_j^2) \cdots k(\mathbf{h}_i^M, \mathbf{h}_j^M) \qquad (28)$$

since multiplying the kernels results in the summation of the squared distances, which equals the distance between the corresponding aRTFs. This means that the algorithm regards the aRTF as a one long feature vector, and is indifferent to the fact that the measurements are aggregated by different nodes. In contrast, the proposed method individually refers to each node and its associated RTF. As a baseline, we also compared the results with a modified version of the SRP-PHAT algorithm [46]. Note that, opposed to the learning-based methods, the SRP-PHAT algorithm requires the knowledge of the exact microphone positions.

The root mean square errors (RMSEs) attained by all four algorithms are compared in two scenarios. In the first scenario, various reverberation levels are examined while the signal to noise ratio (SNR) is set to 25 dB in both the training and the test phases. In the second scenario, the SNR is varying while the reverberation time is set to 700 ms. In the second scenario, the training set is generated with a fixed SNR of 20 dB. All the results are summarised in Fig. 4.

It can be observed that the reverberation level has a direct influence on the performance, and all four algorithms exhibit degraded performance as reverberation increases. Regarding noise, it can be seen that the SNR level does not have a clear impact on the performance. From the comparison between the algorithms it is indicated that the proposed method outperforms the other learning-based algorithms and obtains a significantly smaller error. The SRP-PHAT has comparable results for low reverberation levels, yet it is inferior for high reverberation levels. In addition, the proposed method obtains a smaller error compared to the SRP-PHAT for all noise levels, in high reverberation conditions.

We also examined the algorithm performance in several nontrivial test cases, to better understand its performance and to quantify its robustness. We do not present the results for the other training-based approaches, which were shown to be inferior to the proposed method. However, we present also the results of the SRP-PHAT algorithm, when the comparison to the proposed method is meaningful. First, we examined the reliance upon the direct path information compared with the



(a)

(b)

Fig. 4. The RMSE (a) for various reverberation times (SNR is set to 25 dB) and (b) for various noise levels (reverberation time is set to 700 ms). The proposed method ("MMGP") is compared with two other training-based approaches based on [34] ("Mean" and "Kernel-Mult") and to the SRP-PHAT algorithm [46].

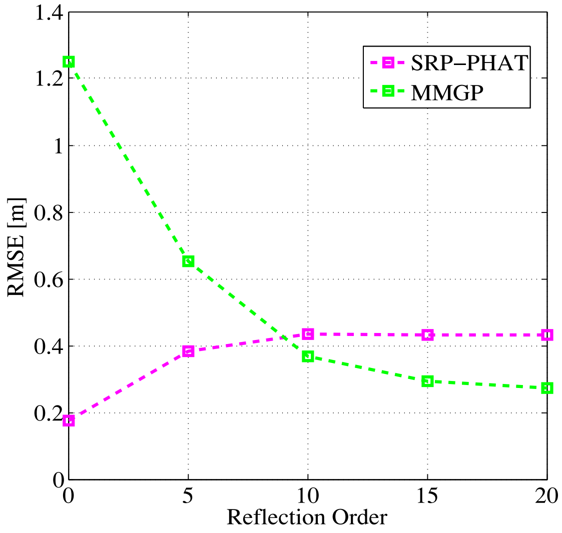


Fig. 5. The RMSE received for various reflection orders of the AIR, for reverberation time set to 700 ms. For the proposed method the training was performed with AIRs with maximum reflection order.

reverberant information. The training set was generated using full AIRs, which consist of reflections of all orders, at a fixed reverberation level of 700 ms. We examined the error obtained in the test phase for various reflection orders of the AIRs, at the same reverberation level. Fig. 5 depicts the RMSE as a function of the reflection order for the proposed method and for the SRP-PHAT algorithm. It can be observed that the errors obtained by the two algorithms represent two opposite trends. The SRP-PHAT algorithm relies on the direct-path information, hence, its performance degrades as the reflection order increases. Conversely, the proposed method relies on the full reflection pattern captured by the receptive RTF, hence, it performs better in high reflection order.

Moreover, we examined a scenario in which the test positions are outside the specified training region. In Fig. 6, we present two cases of a slight deviation from the designated region of up to 0.1 m, and a large deviation from the designated region of up to 1 m. Fig. 6(a) and (c) depict the test positions for each case, and Fig. 6(b) and (d) depict the true y coordinate and the estimated y coordinate for each case. We observe that the estimated position is limited to the designated region, and that in the case of deviation, the estimated position is close to either of the borders of the region. We conclude that the algorithm does not perform extrapolation, however it does make coherent decisions within the defined region.

In addition, we examined the influence of changes in the microphone positions and orientations. The first change was a movement of the nodes after training. In the test phase we randomly shifted the microphones in each node in both the x and the y coordinates. For each node, each coordinate was shifted by an independent random Gaussian variable with variance σ_x^2 or σ_y^2. The RMSE obtained for each total shifting variance $\sigma_{xy}^2 \equiv \sigma_x^2 + \sigma_y^2$ is depicted in Fig. 7(a). The second change regarded the orientation of the microphones. In all other simulations, the microphones were assumed to be omnidirectional, whereas in this simulation we used microphones with cardioid directivity pattern. In both the training and the test phases the microphones were uniformly oriented between zero degrees and a certain maximum orientation angle. The RMSE obtained for each maximum orientation angle is depicted in Fig. 7(b). It can be seen that both the proposed method and the SRP-PHAT algorithm are influenced by changes in the microphone positions and orientations. The influence of these changes on both methods is comparable, and the proposed method maintains the advantage over the SRP-PHAT algorithm.

B. Real Recordings

The algorithm performance was also tested using real recordings carried out in the speech and acoustic lab of Bar-Ilan University. This is a $6 \times 6 \times 2.4$ m room controllable reverberation time, utilizing 60 interchangeable panels covering the room facets. The measurement equipment consists of an RME Hammerfall HDSPe MADI sound-card and an Andiamo.mc (for Microphone pre-amplification and digitization (A/D)). As sources we used Fostex 6301BX loudspeakers, which have a rather flat response in the frequency range 80 Hz–13 kHz. The signals were measured by 6 AKG type CK-32 omnidirectional microphones, which were placed in pairs with internal distance of 0.2 m. The reverberation level was set to $T_{60} = 620$ ms, which was determined by changing the panels configuration. An illustration of the room layout is depicted in Fig. 8(a), and a photograph of the room and the experimental setup is presented in Fig. 8(b).

The source position was confined to a 2.8×2.1 m area located near the room entrance. In this region, we generated $n_L = 20$ equally-spaced labelled samples with resolution of 0.7 m. Additional $n_U = 50$ unlabelled measurements, were generated in this region in random positions. The algorithm performance was examined on 25 test samples also generated in random positions, in the defined region. For generating the labelled samples a chirp signal was used, while for generating both the unlabelled samples and the test samples we used 75 different speech signals of both males and females drawn from the TIMIT database. All the measurements were 10 s long, and were carried out with a sampling frequency of 48 kHz and a resolution of 24-bits. The measured signals were then downsampled to 16 kHz to reflect the frequency content of the TIMIT signals. The RTF estimation was performed similarly to the way it was defined in the simulation part.

We examine two different types of noise sources: air-conditioner noise and babble noise, which is simultaneously played from 3 loudspeakers located in the room. The RMSEs obtained for different SNR levels, when the reverberation is fixed to $T_{60} = 620$ ms, are depicted in Fig. 9(a). We observe that the proposed algorithm outperforms the other methods, and obtains a smaller error for both noise types. It can also be observed that the results obtained based on the lab recordings exhibit the same trends as the results based on the simulated data.

We also applied the recursive adaptation process presented in Section V-B. The positions of the 25 test samples were estimated sequentially, where in each time step, the current sample

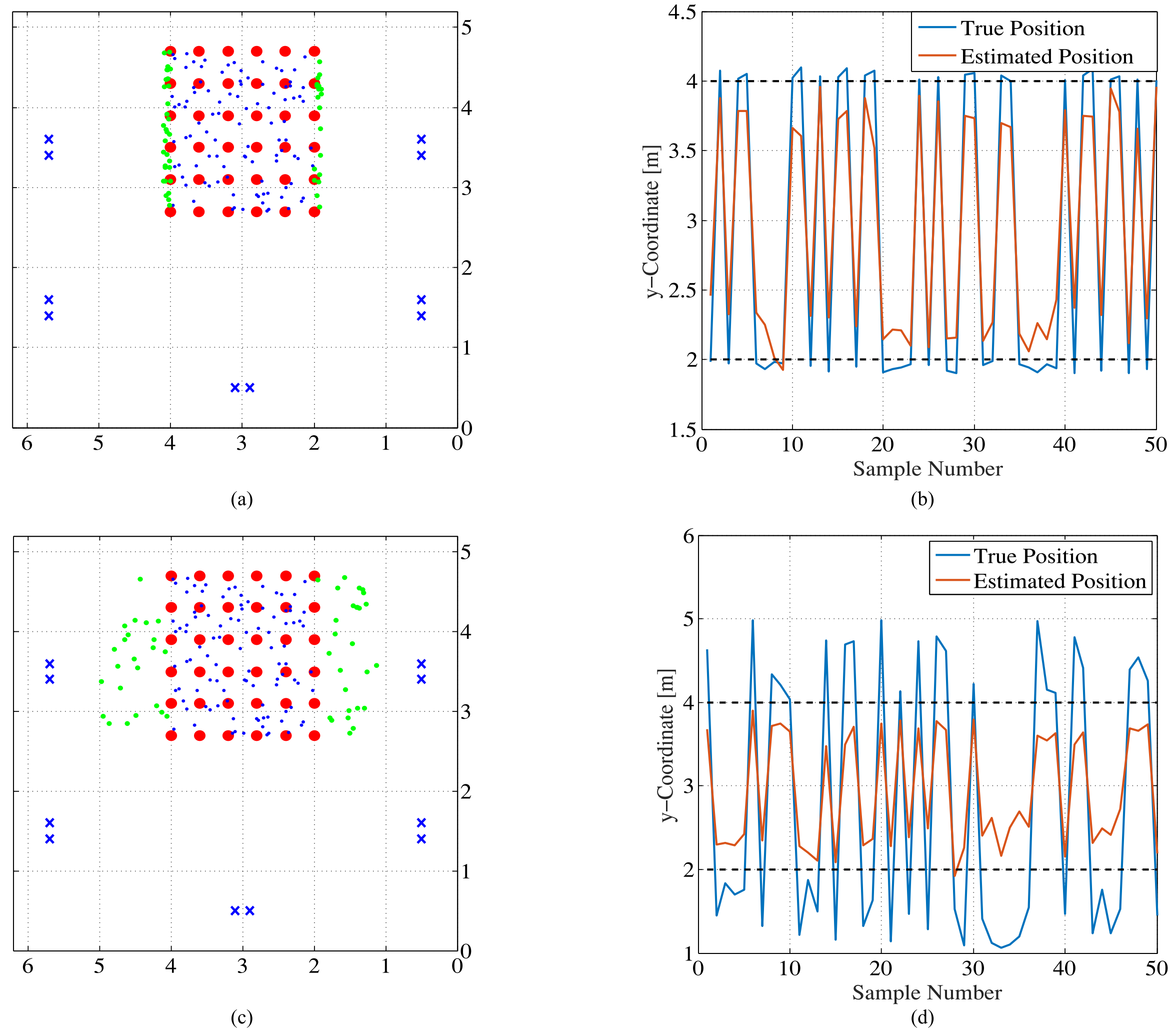


Fig. 6. The test positions for deviations of up to (a) 0.1 m and (c) 1 m (the blue x-marks denote the microphones, the red asterisks denote the labelled samples, the blue dots denote the unlabelled samples, and the green dots denote the test samples). The true and the estimated y-coordinate for deviations of up to (b) 0.1 m and (d) 1 m. Reverberation time is set to 400 ms.

was treated as an additional unlabelled sample, and was used to update the covariance of the MMGP according to (20) and (21). The samples in the test set were initially ordered according to their physical adjacency, so that neighbouring samples were added in a sequential manner. We used the same set of samples, and repeated the sequential adaptation, when applied to different orders of the samples in the set, by mixing the order of neighbouring samples. In addition, we averaged the error for sets of 5 consecutive time steps. Both averages are essential for the sake of generality to ensure that the results are neither tailored to a specific ordering of the samples in the set, nor reflect the quality of a particular sample. Fig. 10 depicts the average RMSE. We observe a monotonic decrease in the error as more samples are added to the computation of the covariance function in a recursive manner. These results also emphasize the importance of the semi-supervised approach, i.e. the significant role that unlabelled samples have in the estimation process.

Another examination was carried out to inspect the effectiveness of the parameter optimization through the ML criterion of the labelled samples, as presented in Section V-C. In Fig. 11, we present the error of the estimated test positions obtained for different values of ε_1 in the range between 100–1000, while the other parameters remain fixed. It can be observed that the optimal value is around 500. For comparison, we followed the proposed optimization using gradient decent starting from an initial value of 100. We obtain that the optimal value for ε_1 is 514, which resembles the empirical value that optimized the performance on the test samples as implied by Fig. 11. This indicates that the parameter values, obtained through an optimization over the labelled samples, yields, in practice, plausible results for estimating the unknown positions of the test samples.

Finally, we investigated the effect of changes in the environmental conditions between the training and the test stages. Training-based approaches are often criticized for being impractical, since identical conditions in both the training and the test phases cannot be guaranteed (e.g. door and windows may be opened or closed, people may move in the room etc.). We examined two types of changes: the door of the room changed from closed (during training) to open (during test) and slight changes in the panel configuration (decreasing the room

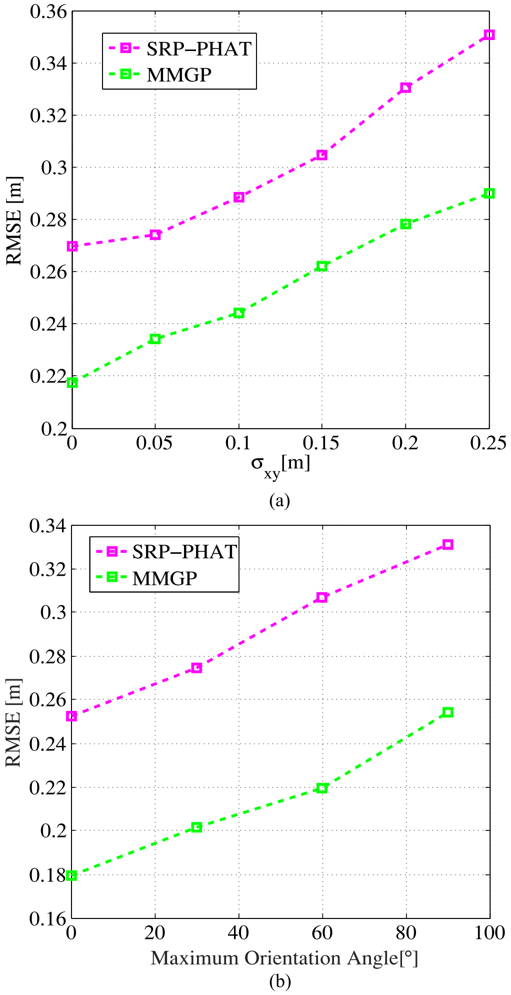


Fig. 7. The RMSE received for (a) node movements during test phase and for (b) microphones with cardioid pattern with different randomization of the orientation level. Reverberation time is set to 400 ms.

reverberation time by about 5%). We repeated the measurements of 20 test samples in both scenarios (the training samples are left unchanged), and compared the results obtained under these conditions to the nominal results, where there is no change in the environmental conditions between the training set and the test set. This comparison is summarized in Table II, which presents the RMSEs in all the defined scenarios. It can be seen that either opening the door or changing the panel configuration does not have a significant impact on the localization results of the proposed method, which indicates that the algorithm is robust to slight changes that are likely to occur in practical scenarios. Note that the results of the SRP-PHAT algorithm are slightly improved under these changes due to the reduction in the reverberation level.

C. Discussion

In this section we discuss several practical aspects of the proposed method. We first discuss the implementation of the method for localization in all three coordinates x, y and z. Experimental results demonstrate the mapping of the RTFs to the x and y coordinates of the source position, for a fixed height. Note that the RTFs used in the proposed algorithm, consist of reflections impinging on the array also from non-horizontal directions. Therefore, a full localization in all directions is feasible. In this case, one has to perform training in the vertical axis as well, and to form a 3D training region. An additional variability in

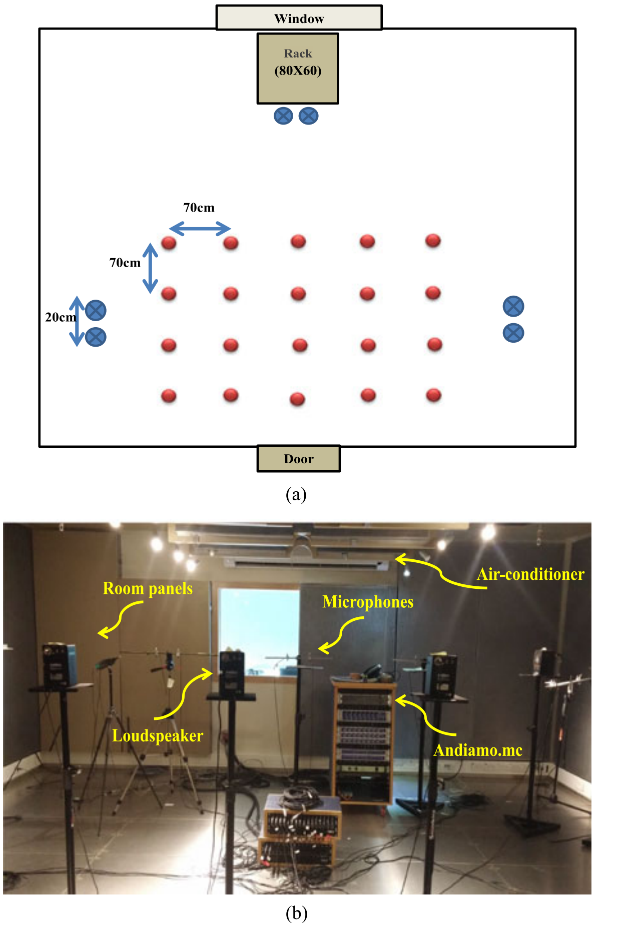


Fig. 8. (a) The room layout: the microphone positions are marked by blue "x" marks, and the positions of the labelled samples are marked by red circles. (b) A photograph of the room.

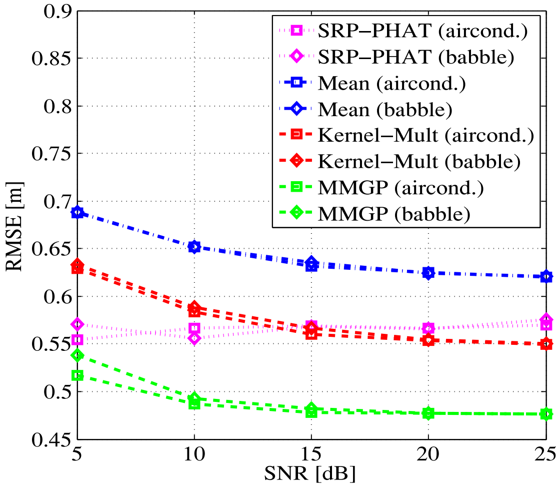


Fig. 9. The RMSE for different noise levels with two types of noise signals: air-conditioner noise and babble noise.

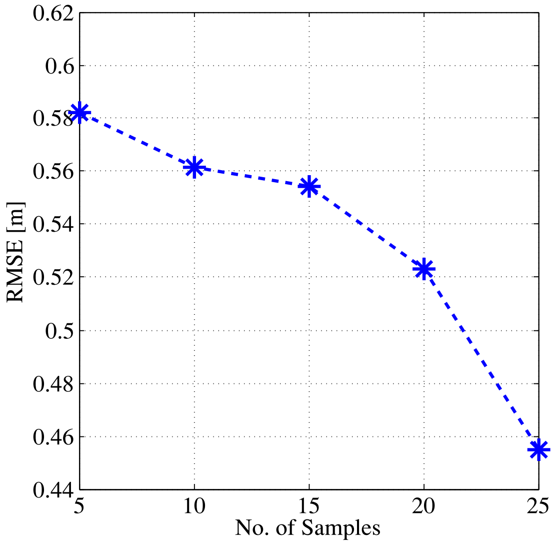


Fig. 10. Demonstration of the recursive adaptation process: in each step the current sample is used to update the covariance function of the process. The results are averaged over groups of five samples.

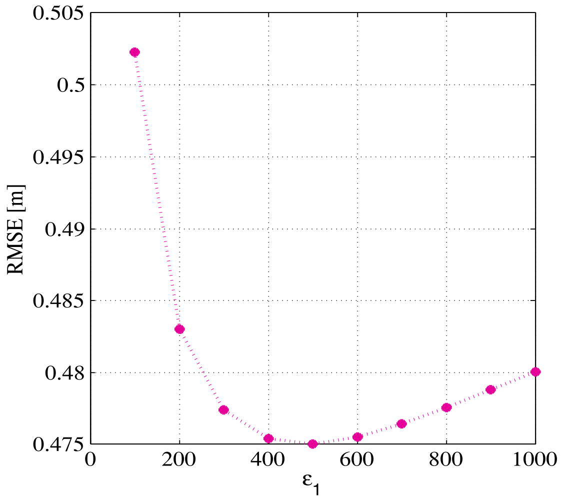


Fig. 11. The RMSE obtained for different values of the kernel scaling parameter ε_1.

TABLE II
COMPARISON BETWEEN THE RMSE OBTAINED IN THE CASE WHERE THE TRAINING AND THE TEST SETS ARE GENERATED EXACTLY WITH THE SAME CONDITIONS (FIRST COLUMN) AND WHEN THE TEST IS GENERATED UNDER SOME ENVIRONMENTAL CHANGES: OPEN DOOR (SECOND COLUMN) OR CHANGES IN THE PANEL CONFIGURATION (THIRD COLUMN)

	Nominal	Door	Panel
MMGP	0.465	0.493	0.506
SRP-PHAT	0.540	0.516	0.531

the z coordinate increases the computational complexity and may influence the quality of the localization results. Another possibility is to form a unique training set for each speaker (and therefore for a fixed height), and circumvent localization in the z coordinate. To form a speaker-specific training set, one needs to acquire a small number of labelled samples in advance. Unlabelled measurements can be collected during run-time, and can be utilized to update the initial model, using the adaptation process presented in Section V-B.

A second issue requiring further discussion is the fact that the estimation of each coordinate is performed separately. Here, we assume that variations of the RTFs reflect an independent movement of the source in either direction. We use this independence assumption to simplify the derived mapping. It is important to note that the same covariance terms are used in (16) or in (21) for the estimation in either axis, implying a direct connection between the estimators. The fact that both estimators rely on the same covariance terms, goes hand in hand with the assumption that similar RTFs are associated with close positions in both coordinates. In general, the converse does not hold, namely, when two RTFs are associated with remote positions, there can still be proximity in one coordinate and remoteness in the other coordinate.

Third, it is important to note that the proposed method is derived for and applied to a localization of a single source. In the case of multiple sources, the method is applicable when the RTFs of each of the sources can be estimated separately. Several works on RTF estimation for multiple sources have been published recently [49], [50], with applications also to multi-source localization [51], [52].

VII. CONCLUSION

In this paper, a novel mathematical approach was developed to fuse the information acquired in a multi-node scenario. This approach, when applied to source localization in ad hoc networks of distributed microphones, deviates from the common practice in the field since it is devised in a semi-supervised manner based on a data-driven model rather than on mathematically predefined relationships. A Gaussian process is used for modelling the unknown relation between the acoustic measurements and the corresponding source positions. The prerecorded training measurements provide useful information about the characteristics of the acoustic environment, and are used to define the covariance of the Gaussian process by averaging over both the different nodes and the different relations to other available acoustic samples. As for the practical aspect, the method produces satisfactory results in challenging adverse conditions including high reverberation and noise levels, with no need for microphone calibration (the algorithm is blind to their positions). The experimental results based on real lab recordings further emphasize the applicability of the algorithm and its ability to successfully locate the source in involved scenarios with possibly natural variations between the training and the test phases. Moreover, the gradual improvement in the performance, as demonstrated in the sequential application of the algorithm, verify the relevance of the information manifested in unlabelled training recordings to the localization task.

APPENDIX A

We consider the relative impulse response $h^m(t, \mathbf{q})$, which satisfies: $a_2^m(t, \mathbf{q}) = h^m(t, \mathbf{q}) * a_1^m(t, \mathbf{q})$. The AIR is typically very long and complicated since it consists of the direct path between the source and the relevant microphone, and the various reflections from the different surfaces and objects in the enclosure. Thus, the relative impulse response also has a complex high-dimensional nature. However, in a static environment, where the acoustic conditions and the microphone positions are fixed, the only parameter that distinguishes between the different AIRs is the source position. For convenience, we work in the frequency domain, and use the relative transfer function (RTF) $H^m(k,\mathbf{q})$, which is the Fourier transform of the relative impulse response $h^m(t,\mathbf{q})$, where k is the frequency index. Accordingly, the mth RTF is given by the ratio between the two acoustic transfer functions (ATFs) of the two microphones in the mth pair, i.e. $H^m(k,\mathbf{q}) = A_2^m(k,\mathbf{q})/A_1^m(k,\mathbf{q})$, where $A_i^m(k,\mathbf{q})$ is the acoustic transfer function (ATF) of the respective AIR $a_i^m(t,\mathbf{q})$. Assuming uncorrelated noise, the mth RTF can be computed using the PSD and CPSD of the measured signals and the noise at the mth pair:

$$\begin{aligned} H^m(k,\mathbf{q}) &= \frac{S^m_{y_2 y_1}(k,\mathbf{q})}{S^m_{y_1 y_1}(k,\mathbf{q}) - S^m_{u_1 u_1}(k)} \\ &= \frac{S_{ss}(k)A_2^m(k,\mathbf{q})A_1^{m*}(k,\mathbf{q})}{S_{ss}(k)|A_1^m(k,\mathbf{q})|^2} = \frac{A_2^m(k,\mathbf{q})}{A_1^m(k,\mathbf{q})} \end{aligned} \quad (29)$$

where $S^m_{y_2 y_1}(k,\mathbf{q})$ is the CPSD between $y_1^m(t)$ and $y_2^m(t)$, $S^m_{y_1 y_1}(k,\mathbf{q})$ is the PSD of $y_1^m(t)$, $S^m_{u_1 u_1}(k)$ is the PSD of the noise $u_1^m(t)$ in the first microphone, and $S_{ss}(k)$ is the PSD of the source $s(t)$. We use a biased estimator of the RTF, neglecting the noise PSD in the denominator of (29):

$$\hat{H}^m(k,\mathbf{q}) \equiv \frac{\hat{S}^m_{y_2 y_1}(k,\mathbf{q})}{\hat{S}^m_{y_1 y_1}(k,\mathbf{q})}. \quad (30)$$

where $\hat{S}^m_{y_2 y_1}(k,\mathbf{q})$ and $\hat{S}^m_{y_1 y_1}(k,\mathbf{q})$ are estimated based on the measured signals. Let $\mathbf{h}^m(\mathbf{q}) = [\hat{H}^m(k_1,\mathbf{q}),\ldots,\hat{H}^m(k_D,\mathbf{q})]^T$, be a concatenation of RTF estimates of the mth node in D frequency bins. Due to the symmetry of the Fourier transform for real valued functions, only the first half of the transform is considered. In addition, we consider only those frequency bins where the speech components are most likely to be present, to avoid poor estimates of (30) in frequencies where the speech components are absent. For the sake of clarity, the position index is omitted throughout the paper.

References

[1] Y. Huang, J. Benesty, and G. W. Elko, "Passive acoustic source localization for video camera steering," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2000, vol. 2, pp. 909–912.

[2] M. I. Mandel, R. J. Weiss, and D. P. W. Ellis, "Model-based expectation-maximization source separation and localization," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 2, pp. 382–394, Feb. 2010.

[3] K. Nakadai, H. G. Okuno, H. Kitano, and G. Hiroshi, "Real-time sound source localization and separation for robot audition," in *Proc. IEEE Int. Conf. Spoken Lang. Process.*, 2002, pp. 193–196.

[4] J.-M. Valin, F. Michaud, J. Rouat, and D. Létourneau, "Robust sound source localization using a microphone array on a mobile robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2003, vol. 2, pp. 1228–1233.

[5] J. Hornstein, M. Lopes, J. Santos-Victor, and F. Lacerda, "Sound localization for humanoid robots-building audio-motor maps based on the HRTF," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2006, pp. 1170–1176.

[6] K. Yao, J. C. Chen, and R. E. Hudson, "Maximum-likelihood acoustic source localization: experimental results," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2002, vol. 3, pp. 2949–2952.

[7] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. AP-34, no. 3, pp. 276–280, Mar. 1986.

[8] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 7, pp. 984–995, Jul. 1989.

[9] T. D. Abhayapala and H. Bhatta, "Coherent broadband source localization by modal space processing," in *Proc. 10th Int. Conf. Telecommun.*, 2003, vol. 2, pp. 1617–1623.

[10] H. Teutsch and W. Kellermann, "EB-ESPRIT: 2D localization of multiple wideband acoustic sources using eigen-beams," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2005, vol. 3, pp. iii/89–iii/92.

[11] O. Nadiri and B. Rafaely, "Localization of multiple speakers under high reverberation using a spherical microphone array and the direct-path dominance test," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 10, pp. 1494–1505, Oct. 2014.

[12] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-24, no. 4, pp. 320–327, Aug. 1976.

[13] M. S. Brandstein and H. F. Silverman, "A robust method for speech signal time-delay estimation in reverberant rooms," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1997, vol. 1, pp. 375–378.

[14] A. Stéphenne and B. Champagne, "A new cepstral prefiltering technique for estimating time delay under reverberant conditions," *Signal Process.*, vol. 59, no. 3, pp. 253–266, 1997.

[15] Y. Rui and D. Florencio, "Time delay estimation in the presence of correlated noise and reverberation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2004, vol. 2, pp. 133–136.

[16] T. Dvorkind and S. Gannot, "Time difference of arrival estimation of speech source in a noisy and reverberant environment," *Signal Process.*, vol. 85, no. 1, pp. 177–204, Jan. 2005.

[17] J. Scheuing and B. Yang, "Disambiguation of TDOA estimation for multiple sources in reverberant environments," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 8, pp. 1479–1489, Nov. 2008.

[18] J. Benesty, "Adaptive eigenvalue decomposition algorithm for passive acoustic source localization," *J. Acoust. Soc. Amer.*, vol. 107, no. 1, pp. 384–391, 2000.

[19] S. Doclo and M. Moonen, "Robust adaptive time delay estimation for speaker localization in noisy and reverberant acoustic environments," *EURASIP J. Appl. Signal Process.*, vol. 2003, pp. 1110–1124, 2003.

[20] J. H. DiBiase, H. F. Silverman, and M. S. Brandstein, "Robust localization in reverberant rooms," in *Microphone Arrays*. New York, NY, USA: Springer, 2001, pp. 157–180.

[21] A. Deleforge and R. Horaud, "2D sound-source localization on the binaural manifold," in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process.*, Santander, Spain, Sep. 2012, pp. 1–6.

[22] A. Deleforge, F. Forbes, and R. Horaud, "Variational EM for binaural sound-source separation and localization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 76–80.

[23] A. Deleforge, F. Forbes, and R. Horaud, "Acoustic space learning for sound-source separation and localization on binaural manifolds," *Int. J. Neural Syst.*, vol. 25, no. 1, 2015, Art. no. 1440003.

[24] T. May, S. van de Par, and A. Kohlrausch, "A probabilistic model for robust localization based on a binaural auditory front-end," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 1, pp. 1–13, Jan. 2011.

[25] X. Wu, D. S. Talagala, W. Zhang, and T. D. Abhayapala, "Spatial feature learning for robust binaural sound source localization using a composite feature vector," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Shanghai, China, Mar. 2016, pp. 6320–6324.

[26] X. Xiao, S. Zhao, X. Zhong, D. L. Jones, E. S. Chng, and H. Li, "A learning-based approach to direction of arrival estimation in noisy and reverberant environments," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 76–80.

[27] X. Xiao, S. Zhao, T. N. T. Nguyen, D. L. Jones, E. S. Chng, and H. Li, "An expectation-maximization eigenvector clustering approach to direction of arrival estimation of multiple speech sources," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 6330–6334.

[28] S. Kitić, N. Bertin, and R. Gribonval, "Hearing behind walls: Localizing sources in the room next door with cosparsity," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 3087–3091.

[29] N. Bertin, S. Kitić, and R. Gribonval, "Joint estimation of sound source location and boundary impedance with physics-driven cosparse regularization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Shanghai, China, Mar. 2016, pp. 6340–6344.

[30] R. Talmon, D. Kushnir, R. Coifman, I. Cohen, and S. Gannot, "Parametrization of linear systems using diffusion kernels," *IEEE Trans. Signal Process.*, vol. 60, no. 3, pp. 1159–1173, Mar. 2012.

[31] R. Talmon, I. Cohen, and S. Gannot, "Supervised source localization using diffusion kernels," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2011, pp. 245–248.

[32] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, "Relative transfer function modeling for supervised source localization," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, New Paltz, NY, USA, Oct. 2013, pp. 1–4.

[33] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, "Semi-supervised sound source localization based on manifold regularization," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 8, pp. 1393–1407, Aug. 2016.

[34] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, "Manifold-based Bayesian inference for semi-supervised source localization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Shanghai, China, Mar. 2016, pp. 6335–6339.

[35] V. Sindhwani, P. Niyogi, and M. Belkin, "Beyond the point cloud: from transductive to semi-supervised learning," in *Proc. 22nd Int. Conf. Mach. Learn.*, 2005, pp. 824–831.

[36] V. Sindhwani, W. Chu, and S. S. Keerthi, "Semi-supervised Gaussian process classifiers," in *Proc. 20th Int. Joint Conf. Artif. Intell.*, 2007, pp. 1059–1064.

[37] S. Gannot, D. Burshtein, and E. Weinstein, "Signal enhancement using beamforming and nonstationarity with applications to speech," *IEEE Trans. Signal Process.*, vol. 49, no. 8, pp. 1614–1626, Aug. 2001.

[38] S. Markovich, S. Gannot, and I. Cohen, "Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 6, pp. 1071–1086, Aug. 2009.

[39] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, "Study on manifolds of acoustic responses," in *Proc. Int. Conf. Latent Variable Anal. Signal Separation*, Liberec, Czech Republic, Aug. 2015, pp. 203–210.

[40] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[41] D. Kushnir, A. Haddad, and R. R. Coifman, "Anisotropic diffusion on sub-manifolds with application to earth structure classification," *Appl. Comput. Harmon. Anal.*, vol. 32, no. 2, pp. 280–294, 2012.

[42] A. Haddad, D. Kushnir, and R. R. Coifman, "Texture separation via a reference set," *Appl. Comput. Harmon. Anal.*, vol. 36, no. 2, pp. 335–347, 2014.

[43] R. Coifman and S. Lafon, "Diffusion maps," *Appl. Comput. Harmon. Anal.*, vol. 21, pp. 5–30, Jul. 2006.

[44] R. R. Lederman and R. Talmon, "Learning the geometry of common latent variables using alternating-diffusion," *Appl. Comput. Harmon. Anal.*, 2015, to be published.

[45] M. A. Woodbury, "Inverting modified matrices," *Memorandum Rep.*, vol. 42, p. 106, 1950.

[46] H. Do, H. F. Silverman, and Y. Yu, "A real-time SRP-PHAT source location implementation using stochastic region contraction (SRC) on a large-aperture microphone array," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 1, 2007, pp. 121–124.

[47] E. A. P. Habets, "Room impulse response (RIR) generator," Jul. 2006. [Online]. Available: https://www.audiolabs-erlangen.de/fau/professor/habets/software/rir-generator

[48] J. Allen and D. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, no. 4, pp. 943–950, Apr. 1979.

[49] N. Ito, S. Araki, and T. Nakatani, "Permutation-free clustering of relative transfer function features for blind source separation," in *Proc. IEEE 23rd Eur. Signal Process. Conf.*, 2015, pp. 409–413.

[50] S. Meier and W. Kellermann, "Analysis of the performance and limitations of ICA-based relative impulse response identification," in *Proc. IEEE 23rd Eur. Signal Process. Conf.*, 2015, pp. 414–418.

[51] A. Deleforge, S. Gannot, and W. Kellermann, "Towards a generalization of relative transfer functions to more than one source," in *Proc. IEEE 23rd Eur. Signal Process. Conf.*, 2015, pp. 419–423.

[52] X. Li, L. Girin, S. Gannot, and R. Horaud, "Multiple-speaker localization based on direct-path features and likelihood maximization with spatial sparsity regularization," *arXiv preprint arXiv:1611.01172*, 2016.

Bracha Laufer-Goldshtein received the B.Sc. (*summa cum laude*) and M.Sc. (*cum laude*) degrees in electrical engineering in 2013 and 2015, respectively, from Bar-Ilan University, Ramat Gan, Israel, where she is currently working toward the Ph.D. degree with the Speech and Signal Processing Laboratory, Faculty of Engineering. She received the Adams Fellowship for the year 2017–2018. Her research interests include statistical signal processing, speaker localization, array processing, and geometric methods for data analysis.

Ronen Talmon received the B.A. (*cum laude*) degree in mathematics and computer science from the Open University of Israel, Ra'anana, Israel, in 2005, and the Ph.D. degree in electrical engineering from the Technion—Israel Institute of Technology, Haifa, Israel, in 2011. In 2014, he joined the Department of Electrical Engineering, The Technion—Israel Institute of Technology, where he is an Assistant Professor of electrical engineering. From 2000 to 2005, he was a Software Developer and Researcher at a technological unit of the Israeli Defense Forces. From 2005 to 2011, he was a Teaching Assistant at the Department of Electrical Engineering, Technion. From 2011 to 2013, he was a Gibbs Assistant Professor at the Mathematics Department, Yale University, New Haven, CT, USA. His research interests include statistical signal processing, analysis and modeling of signals, speech enhancement, biomedical signal processing, applied harmonic analysis, and diffusion geometry. He has received the Irwin and Joan Jacobs Fellowship, the Andrew and Erna Fince Viterbi Fellowship, and the Horev Fellowship.

Sharon Gannot (S'92–M'01–SM'06) received the B.Sc. (*summa cum laude*) degree from the Technion—Israel Institute of Technology, Haifa, Israel, in 1986, and the M.Sc. (*cum laude*) and Ph.D. degrees from Tel Aviv University, Tel Aviv, Israel, in 1995 and 2000, respectively, all in electrical engineering. In 2001, he was Postdoctoral Researcher with the Department of Electrical Engineering (ESAT-SISTA), Katholieke Universiteit Leuven, Leuven, Belgium. From 2002 to 2003, he held a research and teaching position at the Faculty of Electrical Engineering, The Technion—Israel Institute of Technology. He is currently a Full Professor at the Faculty of Engineering, Bar-Ilan University, Ramat Gan, Israel, where he is heading the Speech and Signal Processing Laboratory and the Signal Processing Track. He received the Bar-Ilan University Outstanding Lecturer Award for 2010 and 2014. His research interests include multimicrophone speech processing and specifically distributed algorithms for ad hoc microphone arrays for noise reduction and speaker separation; dereverberation; single microphone speech enhancement; and speaker localization and tracking. He is a co-recipient of seven best paper awards. He has served as an Associate Editor of the *EURASIP Journal of Advances in Signal Processing* in 2003–2012, and as an Editor of several special issues on Multimicrophone Speech Processing of the same journal. He has also served as a Guest Editor of Elsevier's *Speech Communication* and *Signal Processing* journals. He has served as an Associate Editor of the IEEE TRANSACTIONS ON SPEECH, AUDIO, AND LANGUAGE PROCESSING in 2009–2013. He is currently a Senior Area Chair of the same journal. He also serves as a Reviewer of many IEEE journals and conferences. He has been a member of the Audio and Acoustic Signal Processing Technical Committee of the IEEE since January 2010 and the Committee Chair since January 2017. He has also been a member of the Technical and Steering Committee of the International Workshop on Acoustic Signal Enhancement (IWAENC) since 2005 and was the General co-Chair of IWAENC held at Tel Aviv, in August 2010. He has served as the General co-Chair of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics in October 2013. He was selected (with colleagues) to present a tutorial sessions at ICASSP 2012, EUSIPCO 2012, ICASSP 2013, and EUSIPCO 2013.