Applied Acoustics 89 (2015) 71–77

Contents lists available at ScienceDirect

Applied Acoustics

journal homepage: www.elsevier.com/locate/apacoust

Distributed Kalman filter-based speaker tracking in microphone array networks

Ye Tian, Zhe Chen, Fuliang Yin *

School of Information and Communication Engineering, Dalian University of Technology, Dalian 116023, China

ARTICLE INFO

Article history:
Received 8 February 2014
Received in revised form 15 August 2014
Accepted 3 September 2014
Available online 28 September 2014

Keywords:
Distributed Kalman filter
Microphone array network
Time delay of arrival

ABSTRACT

Using a microphone array network, a speaker tracking method based on distributed Kalman filter (DKF) in a noisy and reverberant environment is proposed. Firstly, the time delay of arrival (TDOA) in each microphone pair is estimated by the generalized cross-correlation (GCC) method. Next, the Langevin model is used as state equation to model the speaker's movement, meanwhile the measurement equations with true TDOA are deduced by linearizing the TDOA model. Finally, the moving speaker's positions are estimated by distributed Kalman filtering in a microphone array network. The proposed method is scalable. It can obtain a trajectory of the speaker's movement smoothly with excellent tracking accuracy. Simulation results verify the effectiveness of the proposed method.

1. Introduction

Speaker localization and tracking with microphone arrays is useful in many applications, including audio/video conference system [1], smart video monitor system [2], robot, human–machine interface, far distance speech capture and recognition, etc.

The topics of speaker localization [3–5] and speaker tracking [6–11] have been studied for many years. However, traditional methods usually require dedicated devices, and need to know the positions and geometry structure of microphone arrays.

In practice, it is possible that the geometry structure of microphone arrays is irregular and the positions of them are also distributed randomly. The geometry structure and the positions of microphone arrays can be obtained by self-calibration methods [12,13]. To determine speaker's positions in spatially irregular microphone arrays, the distributed speaker localization methods [14,15] were proposed recently. In [16], the global coherence field (GCF) method was proposed, which was defined over the space of possible sound source locations to represent the plausibility that a sound source was active at a given point. In [17,18], the GCF was extended to Oriented GCF (OGCF) which was allowed to estimate both the position and the head orientation of a single active speaker. In [19], multiple speaker localization with the GCF based on acoustic map de-emphasis was proposed. In [14,20], the steered response power–phase transform (SRP–PHAT) method and its modification were proposed, which steered the microphone array to all potential source positions to search for the candidate source position. In [21,22], the localization performance of the SRP–PHAT method was significantly improved by the selection of suitable microphone pairs in a microphone array network. In [15], Canclini et al. proposed a distributed speaker localization algorithm by minimizing a cost function, which was a fourth-order polynomial obtained by combining hyperbolic constrains from multiple sensors. However, these distributed speaker localization methods only depend on signals in the current frame. They are not yet robust against high room reverberation, and even fail under impulse noise conditions, such as door shutting. Further, in these localization methods spurious sources may be generated in noisy and reverberant environments, sometimes stronger than true speech sources. To deal with these problems, the speaker tracking methods are used to estimate speaker's positions, which depend on not only the current measurement but also a series of past measurements. In this way, a smoothed trajectory of the speaker's movement can be obtained robustly.

Distributed state estimate algorithms such as distributed Kalman filter (DKF) [23,24] have received great attention recently. In the DKF, each node in sensor networks is required to estimate the state of a linear dynamic system by sharing data only with its neighboring nodes each time. Being advantageous over the centralized state estimation algorithms, the DKF do not require a fuse center and is hence robust against its failure.

In this paper, the DKF theory is introduced into a distributed microphone array network and a DKF-based speaker tracking method in a noisy and reverberant environment is proposed.

* Corresponding author.
E-mail addresses: y.tian@mail.dlut.edu.cn (Y. Tian), zhechen@dlut.edu.cn (Z. Chen), flyin@dlut.edu.cn (F. Yin).

http://dx.doi.org/10.1016/j.apacoust.2014.09.004
0003-682X/

Firstly, the time delay of arrival (TDOA) of the speech signals received by microphone arrays in the network is estimated by the generalized cross-correlation (GCC) method. For each node in the microphone array network, whether or not the TDOA is true is checked, and the true measurements are gathered from its neighboring nodes for speaker tracking. Then, the Langevin model [8,10] is introduced as state equation to represent the time-varying locations of a moving speaker, meanwhile the measurement equations with true TDOA are deduced by linearizing the TDOA model. Finally, the distributed Kalman filter is used to estimate the time varying speaker's positions. Since each node in the network only communicates with its neighbors, the proposed speaker tracking method is scalable (i.e. new nodes can join in the network or any node in the network can leave freely), and is robust against lost data or link failure.

The rest of this paper is organized as follows. The DKF theory is introduced in Section 2. The DKF-based speaker tracking method is proposed in Section 3. The effectiveness of the proposed method is verified with simulations in Section 4. Finally, conclusions are drawn in Section 5.

2. Distributed Kalman filter (DKF)

2.1. Data model and problem formulation

Consider a sensor network with N nodes labeled by an index $i = 1, 2, \ldots, N$ spatially distributed as shown in Fig. 1. The communications between them are modeled by a graph $\mathcal{G} = (\mathcal{E}, \mathcal{V})$, where $\mathcal{V} = \{1, 2, \ldots, N\}$ is the vertex set, $\mathcal{E} \subset \{(i,j)|i,j \in \mathcal{V}\}$ is the edge set. An edge (i,j) is in $\mathcal{E}$ if and only if the node i can communicate with the node j. We only consider undirected communication structures, i.e. graphs in which $(i,j) \in \mathcal{E} \iff (j,i) \in \mathcal{E}$ and assume that $\mathcal{G}$ is connected, i.e. there is a path between any two nodes. Let $\mathcal{N}_{i,k} = \{j \in \mathcal{V} | (i,j) \in \mathcal{E}\} \cup \{i\}$ denote the set of neighbors of node i at time k, where a node is also regarded as a neighbor of itself.

Node i takes a measurement $y_{i,k} \in \mathbb{R}^q$ of the common environment state $x_k \in \mathbb{R}^p$ independently at time k, where $\mathbb{R}^n$ denotes a $n \times 1$ real column vector space. The state-space model associated with the measurement of node i is of the form

$$\begin{cases} \boldsymbol{x}_{k+1} = F_k \boldsymbol{x}_k + \Gamma_k w_k \\ \boldsymbol{y}_{i,k} = H_{i,k} \boldsymbol{x}_k + v_{i,k} \end{cases} \tag{1}$$

where w_k is the process noise; $v_{i,k}$ is the measurement noise; and F_k, $H_{i,k}$ and Γ_k are the real transformation matrices at time k. The initial state x_0 is assumed to be zero-mean with covariance matrix $\Pi_0 > 0$, uncorrelated with w_k and $v_{i,k}$.

Assume w_k and $v_{i,k}$ are zero-mean, uncorrelated white noises with the relationship below

$$E\begin{bmatrix} w_k \\ v_{i,k} \end{bmatrix} \begin{bmatrix} w_l \\ v_{j,l} \end{bmatrix}^T = \begin{bmatrix} Q_k \delta_{k,l} & 0 \\ 0 & R_{i,k} \delta_{k,l} \delta_{i,j} \end{bmatrix}, \tag{2}$$

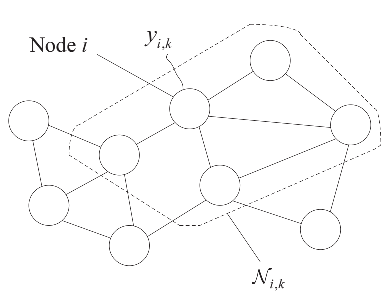

Fig. 1. At time k, node i collects a measurement $\boldsymbol{y}_{i,k}$.

where superscript $(\cdot)^T$ denotes the matrix transpose; $\delta_{i,j}$ is the Kronecker delta function; and Q_k, $R_{i,k}$ are assumed to be positive definite.

2.2. Distributed Kalman filtering algorithm

In [23], a distributed Kalman filter was proposed with objective for every node in the network to compute an estimate of the unknown state x_k by sharing data only with its neighbors $\mathcal{N}_{i,k}$ at time k. The configuration is shown in Fig. 1. The distributed Kalman filter, also called diffusion Kalman filter, is obtained by adding a diffusion step, which is a convex combination of neighboring estimates after a conventional Kalman filtering measurement update.

The diffusion Kalman filtering algorithm defines an $N \times N$ matrix C with real, non-negative entries $c_{l,i}$, which satisfies

$$1^T C = 1^T, \quad c_{l,i} \geqslant 0 \ \forall l, i \quad \text{and} \quad c_{l,i} = 0 \text{ if } l \notin \mathcal{N}_{i,k} \tag{3}$$

where 1 is a $N \times 1$ column vector with unity entries; and $c_{l,i}$ is the (l,i) element of matrix C. The matrix C is called the diffusion matrix, since it governs the diffusion process, and plays an important role in the steady-state performance of the network.

The diffusion Kalman filtering algorithm is summarized below.

Algorithm 1. Diffusion Kalman filter

Start with: $\hat{\boldsymbol{x}}_{i,0|-1} = 0, \quad P_{i,0|-1} = \Pi_0 > 0, \quad k = 0.$
At time k, the following steps are calculated.
Step 1 Incremental Update:
$\boldsymbol{\psi}_{i,k} \leftarrow \hat{\boldsymbol{x}}_{i,k|k-1}$
$P_{i,k} \leftarrow P_{i,k|k-1}$
for every neighboring node $l \in \mathcal{N}_{i,k}$, repeat:

$R_e \leftarrow R_{l,k} + H_{l,k} P_{i,k} H_{l,k}^T$
$\boldsymbol{\psi}_{i,k} \leftarrow \boldsymbol{\psi}_{i,k} + P_{i,k} H_{l,k}^T R_e^{-1} [\boldsymbol{y}_{l,k} - H_{l,k} \boldsymbol{\psi}_{i,k}]$
$P_{i,k} \leftarrow P_{i,k} - P_{i,k} H_{l,k}^T R_e^{-1} H_{l,k} P_{i,k}$

end

Step 2 Diffusion Update:
$\hat{\boldsymbol{x}}_{i,k|k} \leftarrow \sum_{l \in \mathcal{N}_{i,k}} c_{l,i} \boldsymbol{\psi}_{l,k}$
$P_{i,k|k} \leftarrow P_{i,k}$
$\hat{\boldsymbol{x}}_{i,k+1|k} \leftarrow F_k \hat{\boldsymbol{x}}_{i,k|k}$
$P_{i,k+1|k} \leftarrow F_k P_{i,k|k} F_k^T + \Gamma_k Q_k \Gamma_k^T$

where $P_{i,k'|k}$ denotes the covariance matrix of estimation error $\tilde{\boldsymbol{x}}_{i,k'|k} \triangleq \boldsymbol{x}_i - \hat{\boldsymbol{x}}_{i,k'|k}$; and '$\leftarrow$' denotes a sequential, or non-concurrent assignment. According to the algorithm above, node i receives the message $msg_l = (H_{l,k}, R_{l,k}, y_{l,k})$ from its neighbors $\mathcal{N}_{i,k}$ for the incremental update and sends message $msg_i = (\boldsymbol{\psi}_{i,k})$ to its neighbors $\mathcal{N}_{i,k}$ for the diffusion update at time k.

The works in [24,25] applied a weight adaptive strategy to diffusion matrix, which could be adapted to the changes in the data statistics in the diffusion update of the diffusion Kalman filtering algorithm.

Let n_i denote the degree of node i (i.e. the number of nodes connected to the node i including itself), and $\{i_1, i_2, \cdots, i_{n_i}\}$ denote the indexes of the neighbors of node i. A matrix S_i is defined as

$$S_i = [e_{i_1} \ e_{i_2} \cdots e_{i_{n_i}}]_{N \times n_i}, \tag{4}$$

where e_l denotes the lth column of an $N \times N$ identity matrix. According to [24,25], the diffusion matrix can be obtained adaptively as follows.