6M-06

Iterative Outlier Removal Method Using In-Cluster Variance Changes in Multi-Microphone Array Sound Source Localization

Daniel Gabriel[1], Ryosuke Kojima[1], Kotaro Hoshiba[1], Kazuhiro Nakadai[1,2]
1 Tokyo Institute of Technology 2 Honda Research Institute Japan Co., Ltd.

1 Introduction

For auditory scene analysis, which requires obtaining the 6W information (What, Who, Where, When, Why, hoW) [1], knowing the sound source location is essential. In many studies contributed to sound source localization, only azimuth and/or elevation in the microphone-array coordinates have been considered. In applications such as bird song analysis, obtaining the estimated position is the most desirable. Several studies reported such position estimation based on triangulation using multiple microphone arrays [2] [3]. However, the main problem in this approach is the appearance of outliers, which are crossing points of unmatched sound source localization results and noise. This paper presents an iterative outlier removal method which tackles this problem. The presented method successfully localized the desired sound source, labelled it as an inlier and outperformed the other tested method.

Key words*: auditory scene analysis, robot audition, microphone array, outlier detection, clustering*

2 Proposed method

2.1 Initial sound processing

In order to obtain the estimated locations of the sound sources, vector triangulation is performed. We consider M microphone arrays $(MA_1, MA_2, \cdots, MA_M)$ which are distributed in a natural environment. Let $\boldsymbol{d}_m = [d_m(1), d_m(2), \cdots, d_m(D_m)]^T$ be the directions estimated by MA_m every $10\,ms$, where $m = 1, 2, \cdots, M$. The proposed method uses the open source software for robot audition, HARK (Honda Research Institute Japan Audition for Robots with Kyoto University) [4], to obtain these sound source directions. Specifically, a beamforming based method called SEVD-MUSIC (Multiple Signal Classification based on Standard Eigenvalue Decomposition) [5] was used. Next, a triplet, defined as $\bar{\boldsymbol{d}} = [\bar{d}_{m_1}(\alpha), \bar{d}_{m_2}(\beta), \bar{d}_{m_3}(\gamma)]^T$ is created by selecting three estimated vectorised directions obtained from different microphone arrays. For each pair of directions in a triplet, triangulation is performed to obtain three crossing points $P_1 = p(m_1, \alpha, m_2, \beta), P_2 = p(m_2, \beta, m_3, \gamma)$ and $P_3 = p(m_1, \alpha, m_3, \gamma)$. Then, the Euclidean distances between these crossing points are calculated, denoted by L_{12}, L_{23}, L_{13}. Finally, the sound source position x on a 2D plane is estimated by:

$$x = \begin{cases} \frac{1}{3}\sum_i^3 P_i, & (L_{12}, L_{23}, L_{13} \le \theta_1) \\ \varnothing, & \text{otherwise} \end{cases}$$

where θ_1 is a threshold which determines if a triangulation result is valid to be considered a cross point. After all possible combinations of triplets have been processed, all valid sound source positions throughout the recorded data are held in a matrix $\boldsymbol{x}$ of size $N \times 2$.

HARK was also used in sound separation process. For each microphone array m, GHDSS (Geometric High-order Decorrelation-based Source Separation) [6] was used to obtain the separated sounds $\boldsymbol{S}_m$ consisting of s_m elements.

2.2 Iterative outlier removal

This section introduces an outlier removal algorithm using separated sounds. For clustering separated sounds, the 128 point STFT (Short Time Fourier Transform) of $\boldsymbol{S}_m$ is performed to form $\boldsymbol{F}_m = [\boldsymbol{F}_{m,1}, \boldsymbol{F}_{m,2}, \cdots, \boldsymbol{F}_{m,s_m}]^T$. Then, $\boldsymbol{F}$ is defined by $\boldsymbol{F} = [\boldsymbol{F}_{1,1}, \cdots, \boldsymbol{F}_{1,s_1}, \boldsymbol{F}_{2,1}, \cdots, \boldsymbol{F}_{2,s_2}, \cdots, \boldsymbol{F}_{M,1}, \cdots, \boldsymbol{F}_{M,s_M}]^T$. Iterative outlier removal is then performed with R iterations. The score $e(n, r)$ for every sound source position $n = 1, 2, \cdots, N$ in every iteration $r = 1, 2, \cdots, R-1$ is initialized with values 0. First, $\boldsymbol{F}$ is used in k-means clustering:

$$\boldsymbol{c} = kmeans(\boldsymbol{F}, K)$$

where K is the number of clusters and $\boldsymbol{c}$ is a set of cluster indices to which each element of $\boldsymbol{F}$ has been assigned. It has been observed that when clustering with the spectra of the separated sounds and using relatively few clusters in comparison to the size of the dataset, noise data becomes assigned to clusters with different data in every iteration.

Next, sound source position indexing is performed. Since we know which separated sound corresponds to which cross point x, $\boldsymbol{F}$ is connected with $\boldsymbol{x}$ by holding mapping data, which also allows reshaping $\boldsymbol{c}$ into $\boldsymbol{c}^*$ in order to assign a value c_{m,x_n} to each sound source position x_n for every microphone array m :

$$\boldsymbol{c}^* = \begin{bmatrix} c_{1,x_1} & \cdots & c_{1,x_N} \\ \vdots & \ddots & \vdots \\ c_{M,x_1} & \cdots & c_{M,x_N} \end{bmatrix}_{M \times N}$$

A cross point x_n is considered valid, if all of the cluster indices $(c_{1,x_n}, \cdots, c_{M,x_n})$ are the same. This allows us to reduce the $\boldsymbol{c}^*$ matrix into a vector $\hat{\boldsymbol{c}} = [c_{x_1}, c_{x_2}, \cdots, c_{x_N}]^T$.

Finally, the cluster evaluation is performed. Let $\hat{v}_k$ and v_k be the variance of the k-th cluster from the previous and current iteration, respectively. The $e(n, r)$ is incremented if $v_k \le \hat{v}_k$ and $c_{x_n} = k$. After all iterations, a comparison index set is introduced $\boldsymbol{I}_n = \{n' \mid x_n = x_{n'}, n' = 1, 2, \cdots, N\}$, which holds the indices of sound source positions x which have the same coordinates. Next, the following mean operation is performed:

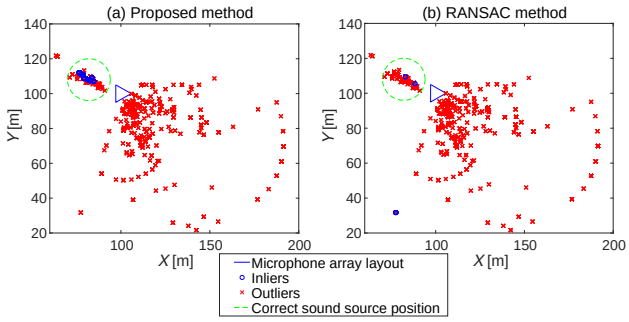

Fig. 1 Index assign comparison of the two methods

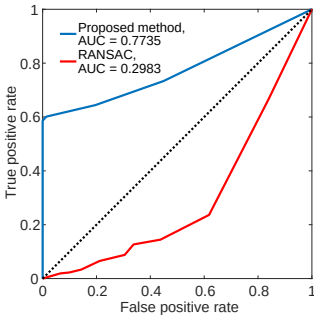

Fig. 2 ROC evaluation of the two methods

$$\mu_n = \frac{1}{R\,|\boldsymbol{I}_n|} \sum_{i \in \boldsymbol{I}_n} \sum_{r} e(i,r)$$

The decision if a cross point is an inlier or an outlier is conducted by :

$$i_x = \begin{cases} 1, & \mu_n \geq \theta_2 \\ 0, & \text{otherwise} \end{cases}$$

where θ_2 is a threshold and i_x indicates an inlier when has the value 1, and an outlier when has the value 0.

3 Evaluation

3.1 Experimental setting

The dataset tested in this experiment contains of about a 10 minute recording of actual bird songs made in Inabu field, Nagoya University. Three 7-chanel microphone arrays were placed at pecks of a equilateral triangle with an edge length of 10 m. The data was taken in a noisy environment, with a river on the south-east of the microphone arrays. The performance of the proposed method was compared with a well-known outlier detection method, RANSAC (Random Sample Consensus) [7].

3.2 Results

The visual comparison of inlier/outlier index assigning results are shown in Fig. 1. The green circle indicates the correct position from which the birds were singing. In this experiment, the more blue circles (inliers) are in the the green circle area, the better the algorithm's performance. On the other hand, blue circles anywhere except the green area as well as red crosses inside the green area are not desired. These figures were made with the best parameter values for each algorithm. The results for different parameter values were evaluated with the ROC (Receiver operating characteristic) method, which are presented in Fig. 2.

As shown in both figures, the proposed method is overwhelmingly better than the RANSAC method. This is because, in this paper, we used separated sounds as inputs to the RANSAC algorithm, which proved troublesome for RANSAC to estimate the frequencies properly due to the noisy environment. Also RANSAC requires an appropriate model to perform data estimation, and for the proposed algorithm a model does not need to be chosen. This provides a major simplification in implementation of the algorithm. Another problem with RANSAC in this application is that very slight changes of the parameters, being of 10^{-4}th order, change the results by a very large factor. On the other hand, the proposed algorithm proved to have clear results, making the decision of the threshold very intuitive.

4 Summary

In this paper an outlier detection algorithm for sound source localization has been proposed. We have described the main idea of the algorithm, shown the results of an experiment and compared them with another vastly used outlier method, RANSAC. We have proven that the proposed method performs better than the other method and that it is easier to apply it in sound source localization and outlier extraction problems using microphone arrays.

Acknowledgement

This work was supported by KAKENHI No.24220006, 16H02884, 16K00294, and ImPACT Tough Robotics Challenge.

References

[1] R. Kojima et al., Semi-Automatic Bird Song Analysis by Spatial-Cue-Based Integration of Sound Source Detection, Localization, Separation, and Identification, *IROS 2016*, pp.1287-1292.

[2] Y. Sasaki et al., Map-generation and identification of multiple sound sources from robot in motion, *IROS 2010*, pp.437-443.

[3] K. Sekiguchi et al., Online Simultaneous Localization and Mapping of Multiple Sound Sources and Asynchronous Microphone Arrays, *IROS 2016*, pp.1973-1979.

[4] K. Nakadai et al., Design and Implementation of Robot Audition System "HARK", *Advanced Robotics*, 24(5–6), pp.739-761, 2010.

[5] R.O. Schmidt et al., Multiple emitter location and signal parameter estimation, *IEEE Trans. Antennas and Propagation*, AP-34(3), pp.276-280, 1986.

[6] H. Nakajima et al., Blind source separation with parameter-free adaptive step-size method for robot audition, *IEEE Trans. ASLP*, 18(6), pp.1476-1485, 2010.

[7] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Communications of the ACM*, 24(6), pp.381-395, 1981.