ABSTRACT

This paper proposes a clustered approach for blind beamforming from ad-hoc microphone arrays. In such arrangements, microphone placement is arbitrary and the speaker may be close to one, all or a subset of microphones at a given time. Practical issues with such a configuration mean that some microphones might be better discarded due to poor input signal to noise ratio (SNR) or undesirable spatial aliasing effects from large inter-element spacings when beamforming. Large inter-microphone spacings may also lead to inaccuracies in delay estimation during blind beamforming. In such situations, using a cluster of microphones (i.e., a sub-array), closely located both to each other and to the desired speech source, may provide more robust enhancement than the full array. This paper proposes a method for blind clustering of microphones based on the magnitude square coherence function, and evaluates the method on a database recorded using various ad-hoc microphone arrangements.

Index Terms— array signal processing, speech enhancement

1. INTRODUCTION

A microphone array consists of multiple microphones that are combined to spatially filter a sound field by forming a beam toward desired locations. With advances in sensor and sensor network technology, there is potential for applications that employ ad-hoc networks of microphone-equipped devices collaboratively as a virtual microphone array [1]. In this new paradigm of pervasive computing, the user is free from intrusive devices while engaging in ordinary activities. While not an ad-hoc sensor network, conditions approaching this have in effect been imposed in recent NIST ASR evaluations on distant microphone recordings of meetings [2]. The NIST evaluation data comes from multiple sites, each with different and often loosely specified distant microphone configurations.

In scenarios where the microphone positions and likely source locations are not known, beamforming must be achieved blindly. There are two general approaches to blindly estimate the steering vector for beamforming. The first is direct estimation without regard to the microphone and source locations. In the NIST meeting data evaluations, such an approach has been used for the Multiple Distant Microphone (MDM) condition in the AMI system [3] and the ICSI system [4], among others. An alternative approach is to instead first determine the unknown microphone positions through array calibration methods [5, 6], and then use the traditional geometrical formulation for the steering vector estimation [7].

For ad-hoc microphone arrangements in a typical meeting room scenario, an issue which has not received significant attention in the research literature is whether it is best to use all microphones in such a situation, or to select some optimal subset of these. While the achievable array gain generally increases with the number of elements, it is commonly assumed that microphones are physically identical and located spatially close to have similar acoustic conditions. When a microphone is arbitrarily placed, the signal acquired by the transducer depends on its characteristics such as gain and directional response and the acoustic conditions of the room involving reverberation and the presence of localised noise sources. This means that some microphones would be better discarded due to their poor input SNR and signal quality. A further consideration is that large inter-microphone spacing may lead to erroneous Time Difference of Arrival (TDOA) computation, effectively causing steering errors in the beamformer. It is also undesirable to have spatial aliasing effects in the beamformer’s directivity pattern. Therefore, it is hypothesised that using a cluster of microphones (i.e., a sub-array), closely located both to each other and to the desired speech source may in fact provide more robust speech enhancement than the full array. In ad-hoc situations, the lack of prior knowledge of microphone and speaker locations means that the clustering of microphones and the selection of clusters must be done blindly.

The paper is organised as follows. Section 2 presents a method to blindly cluster microphones and rank the clusters according to their proximity to the source. Experiments and results on the database recorded with ad-hoc array geometries are presented in section 3. This is followed by discussion and concluding remarks in section 4 and 5.

2. SYSTEM OVERVIEW

The proposed algorithm first performs a segmentation of the input signals as speech and noise using a conservative voice activity detector. In this article a simple energy-based voice activity detector is employed. Microphones are then grouped into local clusters based on the Magnitude Squared Coherence (MSC) function during noise periods. The TDOA between microphones is then used during speech periods as a basis for ranking the clusters according to their proximity to the source.

2.1. Microphone Clustering in a Diffuse Noise Field

The MSC between two microphone signals $i$ and $j$ at discrete frequency $f$, $C_{ij}(f)$ is calculated in the following manner:

$$C_{ij}(f) \triangleq \frac{|\Phi_{ij}(f)|^2}{\Phi_{ii}(f)\Phi_{jj}(f)}$$

(1)

where $\Phi_{ii}(f)$ and $\Phi_{ij}(f)$ are auto- and cross-power spectral densities, respectively, which may be estimated using a recursive periodogram [8].
Environments such as offices or meeting rooms are usually considered to represent diffuse noise fields. The MSC function between two microphones in a diffuse noise field can be modelled as [9]:

\[ C_{ij}(f) = \sin^2 \left( \frac{2\pi f d_{ij}}{c} \right) \]  

(2)

where \( d_{ij} \) is the distance between microphones \( i \) and \( j \) and \( c \) is the speed of sound. According to this model, the noise coherence between two microphones depends principally on the distance \( d_{ij} \) between them. The first minimum of this MSC function occurs at \( f_m = c/(2d_m) \), and beyond this frequency the coherence approaches zero.

This dependence of the diffuse noise coherence on the distance can be used to indicate how close two microphones are, since closely-spaced microphones will have wider main lobes in the coherence function compared to distantly-spaced pairs. To give a measure of overall coherence between microphones, and hence a measure of their proximity, the MSC may be integrated across frequencies.

\[ T_{MSC}^{ij} = \sum_{f=0}^{f_{max}} C_{ij}(f) \]  

(3)

where the summation range is limited by \( f_{max} \) to improve robustness, as the measured coherence function often varies significantly from the theoretical model for frequencies much beyond the main lobe.

In order to cluster microphones, the measure in Equation 3 may be compared to some threshold value to determine if two microphones are sufficiently close to each other. A threshold value can be computed to correspond to a desired distance, \( d_k \), by using the theoretical coherence model from Equation 2 and summing up to a threshold frequency \( f_k = c/(2d_k) \) (corresponding to its first minimum):

\[ T_k = \sum_{i=0}^{f_k} \sin^2 \left( \frac{2\pi f d_k}{c} \right) \]  

(4)

The measured value for \( T_{MSC}^{ij} \) may then be compared to this intra-cluster threshold \( T_k \). If \( T_{MSC}^{ij} \geq T_k \), then microphones \( i \) and \( j \) are grouped in the same cluster, otherwise they belong to separate clusters. This conservative binary classification is evaluated over all microphone pairs to form an initial set of clusters. Clusters containing close elements may then be merged in a subsequent pass. The proposed clustering algorithm is as follows:

1. Assign \( b_{ij} = 1 \) if \( T_{MSC}^{ij} \geq T_k \) for \( i, j = 1, \ldots, N \).
2. Compute \( B_i = \sum_{j=1}^{N} b_{ij} \) for \( i = 1, \ldots, N \).
3. Select the microphone belonging to the most pairs as the centre microphone of the first cluster, ie \( m_1 = \arg \max_i B_i \).
4. Form cluster 1, \( Q_1 \), with \( Q_1 = \{ j | b_{m_1j} = 1 \} \).
5. Remove microphones belonging to cluster 1 from consideration, then repeat the above steps to form clusters \( K = 2 : K \) until all microphones have been assigned a cluster.
6. Once the set of initial clusters \( Q_{1:K} \) is formed, a merging pass is conducted. Two clusters are merged if \( T_{MSC}^{ij} \geq T_k \) where microphone \( i \) belongs to one of the cluster and \( j \) belongs to another, and \( T_k \) is an inter-cluster threshold calculated using a more restrictive distance criteria \( d_k \) in Equation 4.
7. In the case the above steps result in the formation of any single-element clusters, these may be merged with the closest cluster if a relaxed inter-cluster threshold is satisfied.

### 2.2. TDOA-based Cluster Ranking

Assuming a known period of speech from a single person, the delay in receiving a sound wave between clusters indicates their relative distance to that speaker. The TDOA between microphone \( i \) and \( j \) is computed by means of Generalised Cross-Correlation (GCC) function with phase-transform (PHAT) weighting defined as [10]:

\[ \hat{G}_{PHAT}^{ij}(\tau) = \frac{x_i(f)x_j^*(f)}{|x_i(f)x_j^*(f)|} \]  

(5)

\[ \tau(i, j) = \arg \max_{\tau} \left( \hat{G}_{PHAT}^{ij}(\tau) \right) \]  

(6)

where \( \hat{G}_{PHAT}^{ij}(\tau) \) is the inverse Fourier Transform of Equation 5.

The detailed steps to rank clusters based on their proximity to the speaker are outlined below. The algorithm considers both the proximity of the closest microphone within each cluster (using the TDOAs between a reference microphone from each cluster), as well as the spatial extent of the cluster (using a measure of the spread of TDOAs within each cluster).

1. **Find the closest microphone to the speaker within each cluster** and set it as reference \( m_k \). To do this, for cluster \( k \), choose an initial arbitrary reference microphone \( m_k^* \) and calculate \( \tau(i, m_k^*) \) for each microphone \( i \) in the cluster. Update the reference microphone for the cluster to be the closest microphone by selecting the one having the minimum TDOA, ie \( m_k = \arg \min_i \tau(i, m_k^*) \).
2. **As a measure of cluster spread**, calculate the mid-range TDOA offset for each cluster \( \delta_k \) relative to its reference microphone. To do this, for cluster \( k \), calculate the TDOA of each other microphone with respect to the reference microphone selected in the previous step, ie \( \tau(i, m_k) \) for each microphone \( i \) in the cluster. Set the mid-range TDOA offset for the cluster to be half of the maximum TDOA, ie \( \delta_k = \frac{1}{2} \max_i \tau(i, m_k) \).
3. **Find the reference cluster** \( c_{ref} \) as the one with its reference microphone closest to the speaker. To do this, first choose an arbitrary reference cluster \( k \), and calculate the set of TDOAs between its reference microphone and the reference in all clusters \( k, \tau(m_k, m_k^*) \). Update the reference cluster to be the one which has the minimum TDOA, ie \( c_{ref} = \arg \min_k \tau(m_k, m_k^*) \).
4. **Form the final proximity score** \( D_k \) for each cluster by compensating the inter-cluster TDOAs with the cluster mid-range offsets. Given the set of mid-range offsets for all clusters, \( \delta_k \), and the set of TDOAs with respect to the reference cluster \( c_{ref} \), \( D_k = \tau(m_k, m_{c_{ref}}) + \delta_k - \delta_{c_{ref}} \).

The clusters may then be ranked according to their proximity to the speaker according to the score \( D_k \). Note that it is possible that this score may be negative, indicating that the reference cluster from step 3 did not turn out to be the closest cluster when considering the mid-range offsets. Considering these mid-range TDOA offsets is a means to compensate for the differing spatial extents of clusters. For instance, while some clusters may have a reference microphone that is close to the speaker, they may also be large clusters with other microphones that are quite far from the speaker.
3. EXPERIMENTS

3.1. Experimental Setup

Experiments were conducted in a meeting room of size 5.3m x 4.4m x 2.7m, as shown in Figure 1. The main sources of noise were a PC, laptop, a projector, and air conditioning. To experiment with different ad-hoc array geometries, microphones were mounted in varying positions on two cork boards placed on top of the meeting table. A total of 8 microphones (AKG C417 omnidirectional condenser microphones) were used for each ad-hoc geometry. The microphones were recorded using a MOTU 8pre audio interface and SONAR 8 software, allowing simultaneous, fully synchronised playback and recording of multiple audio channels.

3.2. Microphone Clustering Evaluation

Compared to classification, clustering can be difficult to objectively evaluate, as often there is no correct grouping that can be considered as ground-truth. To evaluate use of the noise coherence feature, the results of the proposed automatic cluster algorithm based on noise recordings were therefore compared to sub-arrays formed by applying the same algorithm to ground-truth distances between known microphone positions. The comparison is illustrated for three ad-hoc geometries in Figure 2. For the clustering algorithm, the intra-cluster distance threshold $d_i$ and inter-cluster distance threshold $d_o$ are set to be 30cm and 20cm respectively. Single element clusters are merged to the nearest cluster if they satisfy a relaxed threshold of 50cm.

To measure overall performance, the purity measure used in speaker clustering literature is adapted to the current context [11, 12]. Dual purity measures are used to evaluate how well closely the automatic clusters match the ‘ground-truth’ sub-arrays, and vice versa. First, define:

$N_c$: total number of true sub-arrays.

$N_s$: total number of found clusters.

$n_{ij}$: total microphones in cluster $i$ that are from sub-array $j$.

$n_i$: total microphones in cluster $i$.

$n_j$: total microphones in sub-array cluster $j$.

The purity of a cluster $p_i$ is defined as:

$$ p_i = \frac{N_s}{N_c} \sum_{j=1}^{N_s} n_{ij}^2 / n_i^2 $$  \hspace{1cm} (7)

and the average cluster purity $acp$ is:

$$ acp = \frac{1}{N_c} \sum_{i=1}^{N_c} p_i \cdot n_i. $$  \hspace{1cm} (8)

Similarly, the sub-array purity $p_{ij}$ and $asp$ are defined as $p_{ij} = \sum_{i=1}^{N_c} n_{ij}^2 / n_j^2$ and $asp = \frac{1}{N_s} \sum_{j=1}^{N_s} p_{ij} \cdot n_j$.

The $asp$ gives a measure of how well a sub-array matches only one cluster, and the $acp$ gives a complementary measure of how well a cluster matches only one sub-array. These scores can be combined to obtain an overall score, $K = \sqrt{acp \times asp}$. Table 1 presents the average score results for $acp$, $asp$, and $K$ for 20 different ad-hoc geometries.

3.3. Cluster Selection and Blind Beamforming

A speech enhancement experiment was performed on a database recorded with microphones configured to form two equally sized sub-arrays. Three different speaker positions were configured to examine the effect of different speaking orientations and relative distances to clusters. Speaker 1 faced all microphones but was closer to one cluster than the other. Speaker 2 was oriented to face one of the clusters and be approximately perpendicular to the other. Speaker
Table 1. Clustering results in terms of average acp, asp and K over 20 ad-hoc geometries.

<table>
<thead>
<tr>
<th></th>
<th>acp</th>
<th>asp</th>
<th>K</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.855</td>
<td>0.945</td>
<td>0.887</td>
<td></td>
</tr>
</tbody>
</table>

3’s orientation was between these extremes. These positions are illustrated in Figure 1. A total of 30 utterances from the WSJ/CAMO evaluation set were recorded for testing [13]. To simulate random microphone placement within these constraints, the array geometry were rearranged for every 10 recorded sentences. The average segmental signal-plus-noise to noise ratio [14] was computed on signals obtained using blind delay-sum beamforming. The speech/noise segmentation for segmental SNR calculation was obtained using the supplied phonetic time-aligned transcriptions from the database. The beamforming output $d_{\text{out}}$ for the closest cluster, the second cluster, and using all microphones, are presented in Table 2, along with the average single channel input segmental SNR $\bar{x}_{\text{in}}$ in each cluster.

4. DISCUSSION

The high purity measures in Table 1 indicate that clustering using the magnitude square coherence feature well-approximates sub-arrays formed from known microphone positions. The lower value of $\bar{acp}$ compared to $\bar{asp}$ shows that the automatic measure tends to create larger clusters for microphones separations near the threshold value, indicating that the measured coherence tends to exceed that predicted by the diffuse model in this particular environment. Examples of this occurring are shown in Figure 2(ii)-(iii), however when the separation between clusters is clear, as in Figure 2(i), the algorithm succeeds. This observation may motivate using more conservative distances than those desired when formulating the thresholds.

In the enhancement experiments, blind delay-sum beamforming using only the closest cluster gives an SNR improvement over the single channel input, and provides greater enhancement than the full array. Note that the magnitude of SNR differences in the result table represent reasonable practical results using delay-sum from a 4-element array. For the second closest cluster, the beamforming does not yield improvement over individual input microphones. Subsequent investigation attributed this to inaccuracy in the time delay estimations when using microphones that were further from the source. A similar effect is observed in the beamformer output from the full array. The TDOA computation in this case is done by selecting a reference microphone and calculating delays relative to this reference. Unfortunately the calculation of delays in this way causes inaccuracies between more distant pairs. The cross-correlation function between two microphones which are spatially close will be dominated by a peak corresponding to the TDOA difference, as they receive signals which have otherwise undergone very similar acoustic transfer functions from the source. For microphones with large distance however, the two impulse responses are likely to be different, increasing the probability of reflections obscuring the cross-correlation peak.

Table 2. Segmental SNR (dB) of cluster beamforming in Experiment A. Results are averaged over 30 utterances.

<table>
<thead>
<tr>
<th></th>
<th>Closest cluster</th>
<th>$2^{nd}$ closest clust.</th>
<th>All mic.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$x_{\text{in}}$</td>
<td>$d_{\text{out}}$</td>
<td>$x_{\text{in}}$</td>
</tr>
<tr>
<td>1</td>
<td>5.6, 7.8</td>
<td>11.90</td>
<td>12.89</td>
</tr>
<tr>
<td>2</td>
<td>5.6, 7.8</td>
<td>12.35</td>
<td>13.16</td>
</tr>
<tr>
<td>3</td>
<td>1.2, 3.4</td>
<td>10.40</td>
<td>10.96</td>
</tr>
</tbody>
</table>

5. CONCLUSION AND FUTURE WORKS

This paper proposed a novel method for blind clustering of ad-hoc microphone arrays and examined its use on a new database recorded for this purpose. Experiments demonstrated that a measure derived from the noise magnitude square coherence was successful in grouping microphones into meaningful proximate clusters. Subsequent enhancement experiments also demonstrated that beamforming using a cluster of microphones located close to a speaker could be advantageous compared to using the full set of microphones in certain scenarios. Ongoing research is investigating whether this in turn leads to improvements in speech recognition performance when both microphone and speaker locations are unknown. Finally, it is noted that while this research has been constrained to propose solutions for unknown geometries, in true ad-hoc scenarios robust methods must also consider potential differences in microphone quality, calibration and synchronisation.

6. REFERENCES