BEAMFORMING ALTERNATIVES FOR MULTI-CHANNEL TRANSIENT ACOUSTIC EVENT CLASSIFICATION

Brandon Smith, Les Atlas, and Maya R. Gupta

Department of Electrical Engineering,
University of Washington, Seattle, WA 98195-2500, USA

ABSTRACT

Signals acquired through a microphone array are typically beamformed to combine channels and improve the signal-to-noise ratio (SNR). However, it has been previously shown that alternative methods for handling multi-channel systems can outperform beamforming for speech recognition applications. In this paper, we implemented a comprehensive set of classification tests using multiple classifiers and feature extraction techniques to determine whether the alternative methods generalize beyond speech recognition applications. We show that applying the alternative methods (in a slightly simpler form) outperforms beamforming when used for classifying transient acoustic projectile weapon signals. Furthermore, an additional technique is introduced which outperforms both beamforming and previously proposed alternatives in certain classification scenarios. For the majority of classification tests, the improvements seen through the use of these alternative methods are statistically significant.

Index Terms— Pattern classification, feature extraction, cepstral analysis, acoustic beam steering, transient propagation.

1. INTRODUCTION

In many classification applications involving microphone arrays, it is common to combine the signals from each element of the array to yield event predictions. Beamforming can improve the signal-to-noise ratio (SNR) by steering the array in the source direction. Typically beamforming or other multi-channel combination techniques are applied before classification. Alternatively, the channel combination methods proposed by Wang et al. [1] independently classify each channel and provide an event prediction by combining class likelihoods assigned during classification. Originally designed for speech recognition applications, we simplify this method for a more generalized class of signals.

In this paper, we summarize the key results of a comprehensive set of classification tests on a database of transient acoustic projectile weapon signals acquired with a 4-channel microphone array. These tests were designed to compare the baseline beamforming technique to several alternative channel combination methods.

Thanks to the Army Research Laboratory for funding.

2. CHANNEL COMBINATION METHODS

This section describes methods for combining signals captured with a microphone array. Fig. 1 and Fig. 2 display the differences between microphone array event classification with beamforming and the alternative processing techniques discussed in this paper.

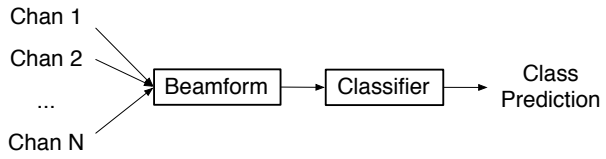

Fig. 1. Event classification with standard beamforming.

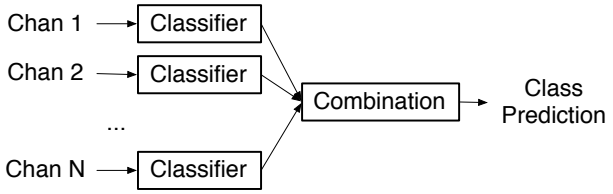

Fig. 2. Event classification with a likelihood-dependent channel combination.

2.1. Beamforming

Beamforming is a mature signal processing technique. See, for example, Brandstein and Ward [2] for a review. We implemented a simple delay-and-sum beamformer, where the array channels are combined by summing time-aligned versions of each signal. The beamformed event is then represented by a single signal and sent through a classifier (Fig. 1). The classifier outputs a set of probabilities (or likelihoods) $\{p_c\}$ where

c is one of a finite number of classes. The classifier then predicts the class with the highest probability,

$$\hat{y} = \underset{c}{\operatorname{argmax}}\ p_c. \tag{1}$$

2.2. Likelihood-Dependent Channel Combination (LDCC)

Likelihood-dependent channel combination (LDCC) is a simplified version of the previously proposed single-decoder processing technique [1]. The Viterbi decoder stage of the single-decoder processing method becomes unnecessary for our class of signals. Instead of combining channels before classification (as with beamforming), each channel is classified individually and combined using their assigned class likelihoods (Fig. 2). These likelihoods will be written $p_{g,c}$ where $g \in \{1, 2, \ldots, N\}$ for N channels. The following three methods for combining class likelihoods are investigated.

The voting-method [1] finds the class with the largest likelihood for each channel $\hat{y}_g$ and computes a majority vote over the chosen classes,

$$\hat{y} = \underset{c}{\operatorname{argmax}} \sum_g I_{\hat{y}_g = c}. \tag{2}$$

A tie is settled by summing the likelihoods associated with the tie and choosing the class associated with the maximum sum.

The maximum-summation-likelihood method [1] is found by summing across the channel likelihoods. A class is then predicted based on the largest marginal,

$$\hat{y} = \underset{c}{\operatorname{argmax}} \sum_g p_{g,c}. \tag{3}$$

The maximum-likelihood method is an extension to the two previously discussed methods. This method predicts a class by finding the maximum likelihood over all channels and all classes.

$$\hat{y} = \underset{g,c}{\operatorname{argmax}}\ p_{g,c}. \tag{4}$$

Each of the above methods provide simple automatic data censoring. A malfunctioning channel may have very little effect in the above methods, but may highly distort a beamformed signal.

3. DATABASE

To test the above-mentioned techniques, we used a set of multi-channel transient acoustic signals provided by the Army Research Laboratory.

The database consists of digital sound recordings of the launch and impact sounds from three types of weapons: mortars, rockets, and rocket-propelled grenades (RPGs). The signals are transient, on the order of 500 milliseconds in duration, and significant reverberation and/or dispersive effects are audibly present. The database contains six possible classes, all hand-labeled during acquisition. A total of 1200 events were recorded, each with four channels (one for each microphone in the system). Table 1 shows the number of events associated with each class. In addition to the 6-class problem, 2 and 3-class problems were investigated by appropriately grouping classes, providing a greater understanding of the classification potential. The 2-class problem was Impact vs. Launch. The 3-class problem was Mortar vs. Rocket vs. RPG.

Table 1. The number of events organized by class.

	Launch	Impact
Mortar	505	340
Rocket	56	15
RPG	51	50

These acoustic signals were captured by an Unattended Transient Acoustic MASINT (measurement and signatures intelligence) System (UTAMS). This 4-channel microphone array forms a regular tetrahedron and is configured as shown in Table 2. The array is typically raised approximately one meter off the ground, so Mic1 is a total of two meters above the ground.

Table 2. Geometric configuration of UTAMS 4-channel microphone array (in meters).

	x-coor	y-coor	z-coor
Mic1	0	0	1
Mic2	0	1	0
Mic3	-0.866	-0.5	0
Mic4	0.866	-0.5	0

The array microphones were made by Knowles Acoustics, model BL-1994. A 24-bit PAR4CH A/D converter made by Symmetric Research was used to digitize the signals at a sampling rate of 1001.602 Hz.

4. CLASSIFICATION

Both parametric and non-parametric classifiers were implemented to provide a thorough comparison between beamforming and the alternative methods. To evaluate the classifier performance, a standard 10-fold cross-validation method [3] was implemented.

4.1. Feature Extraction

Feature vectors of dimension m were extracted from the original time series to provide more discriminable and lower dimensional information to the classifiers. In speech recognition applications, cepstral processing techniques are typically employed to extract useful signal features for classifica-

tion. In previous experiments with this database, we considered several cepstral based features including: standard linear frequency cepstrum, minimum-phase cepstrum, averaged minimum-phase cepstrum, delta cepstrum and cepstral mean normalized versions of each [5]. The minimum-phase cepstrum is found by computing the cepstrum on the minimum-phase component of a signal after a minimum-phase/all-pass decomposition. (This is useful in many practical scenarios because the minimum-phase portion of a time series has been shown to be less affected by reverberation [6]). Averaging the minimum phase cepstrum is another feature extraction technique used to help remove echos and reverberation effects [7]. Delta features are found by concatenating time-differenced cepstral features with the original cepstrum to provide a feature vector with presumed perceptually useful temporal signal information [4]. Cepstral mean normalization is a processing technique used to remove unknown linear filtering operations.

From the above feature extraction techniques, delta cepstrum features were shown to provide superior classification results for this database, and are used in this paper.

4.2. Classifiers

4.2.1. GMM

Statistical parametric classifiers fit a probability distribution in feature space to the training points of each class. We assumed the classes in this database might not be represented by a single Gaussian distribution, hence a Gaussian mixture model (GMM) was employed. GMMs generate probability distributions by summing several m-dimensional weighted multivariate Gaussian distributions (called "mixture components"). The weights and Gaussian parameters are fit with an expectation maximization algorithm during the classifier training process. More detailed information about this classifier can be found in [8].

4.2.2. LIME

Linear interpolation with maximum entropy (LIME) is a non-parametric classifier that generalizes the k-nearest-neighbors (k-NN) classifier by applying a non-uniform weight to each neighbor [9]. LIME has been shown through simulations to achieve a lower bias than other neighborhood methods, and to consistently outperform other weighted k-nearest-neighbor classifiers.

The class prediction equation for LIME is the same as for k-NN:

$$\hat{y} = \underset{g}{\operatorname{argmax}} \sum_{i \in \boldsymbol{J_k}} w_i I_{y_i = g}, \tag{5}$$

where $\sum_{i \in \boldsymbol{J_k}} w_i = 1$, and $\boldsymbol{J_k}$ denotes the set of k neighbor training points. The weights w_i are computed by minimizing the distance between the test point and a linear interpolation of the training points. For the solution to be unique, a scalable maximum entropy term is incorporated to force the weights toward a uniform distribution. The LIME weights $\boldsymbol{w}^*$ solve the following objective function,

$$\boldsymbol{w}^* = \underset{\boldsymbol{w}}{\operatorname{argmin}} \left(||\boldsymbol{w}^T \boldsymbol{X} - \boldsymbol{x}^T||_2^2 + \lambda \sum_{i=1}^{k} w_i \log w_i \right), \tag{6}$$

where $\boldsymbol{X}$ is a k-by-m matrix with the training points as rows, $\boldsymbol{x}$ is a test point (in m dimensions), and λ is a trade-off parameter. Because the LIME objective function (6) is convex, standard convex optimization methods can be used to solve for $\boldsymbol{w}^*$.

When λ is nearly zero, the weights are found such that they minimize the distance between the test point and the linear interpolation of the training points. When λ is large, the focus is put on maximizing the entropy (achieving a uniform weighting like k-NN).

The probability associated with the prediction $\hat{y}$ will be defined as

$$p = \sum_{i \in J_k} w_i I_{y_i = \hat{y}}, \tag{7}$$

which simply sums the weights of the training points associated with the predicted class. More information about the theoretical properties and performance of LIME can be found in [9, 10].

4.3. Parameter Optimization

When building a classifier, there are often several parameters that need to be tuned to optimize classification performance. The candidate parameter sets for GMM and LIME are shown below. For each classification scenario (e.g. 2-class, GMM, maximum-likelihood method) a single value from each set is found to jointly minimize the classification error-rate. This was done using 10-fold cross-validation and an exhaustive search over these candidate parameters. More details on the choice of candidate parameter values can be found in [5].

Two parameters are optimized for the GMM classifier:

1. $m = \{10, 20, 30, 40, 50, 60\}$,
2. number of mixture components $= \{2, 4, 6, \ldots, 20, 22\}$.

Three parameters are optimized for the LIME classifier:

1. $m = \{10, 20, 30, 40, 50, 60\}$,
2. $k = \{10, 20, 30, 40\}$,
3. $\lambda = \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$.

5. RESULTS

The optimal classification error-rates achieved for each classification scenario are shown in Table 3. A chance performance is also included showing the error-rate for a classifier that always predicts the most abundant class (largest prior). For each row in the table, the beamforming method is always inferior to the best LDCC method (in bold-face). The maximum-summation-likelihood method provides the lowest error-rate

Table 3. Classification error-rates (%).

		Baselines		LDCC		
		Chance	Beamform	Vote	MaxSum	Max
2 Class	GMM	39.9	19.5	17.7	**15.8**	18.6
	LIME	39.9	17.9	15.8	15.8	**15.5**
3 Class	GMM	16.8	14.3	10.2	**9.1**	9.4
	LIME	16.8	9.6	11.2	9.7	**8.4**
6 Class	GMM	50.3	29.3	24.0	**21.0**	24.3
	LIME	50.3	24.6	22.6	**21.0**	21.1

for the GMM classifier; however, the maximum-likelihood method appears more promising for the LIME classifier.

As with the previous study in speech recognition, we are able to achieve higher performance when using the alternative channel combination methods. However, a test was not conducted in [1] to verify the statistical significance of the the resulting error-rate differences between beamforming and the alternative processing methods. We chose to measure the statistical significance using a two-tailed binomial test [11]. For each row, a significance value (p-value) was obtained by a pairwise comparison of the best LDCC method with the baseline beamforming method. The statistical significance level is interpreted as the probability that the differences in the observed classification performances is due to chance. Therefore, low levels indicate a strong difference (high significance) between methods. Typically, p-values less than 0.05 are considered significant. Table 4 shows the error-rate difference and the p-value between beamforming and the best LDCC method.

Table 4. LDCC error-rate improvement over beamforming with associated p-values.

		Improvement	p-value
2 Class	GMM	3.7%	1.8e-10
	LIME	2.4%	6.5e-02
3 Class	GMM	5.2%	1.1e-10
	LIME	1.2%	1.4e-01
6 Class	GMM	8.3%	2.5e-12
	LIME	3.6%	1.1e-02

As shown in Table 4, with a standard p-value significance level of $\alpha = 0.05$, four out of the six results are statistically significant, and the others are within 10% of being significant. It is interesting to note that the GMM p-values are much smaller than those for LIME. Much of this has to do with the fact that the GMM classifier exhibits more improvement than LIME, even though LIME is almost always a superior classifier for this database.

6. CONCLUSIONS

Through comprehensive classification tests, we have verified that beamforming is not always the optimal channel combination technique for multi-channel signal classification. Channel combination methods proposed for speech recognition have been shown to generalize to a much different class of signals. Furthermore, the additional method proposed in this paper outperformed all other techniques in several classification tests.

Investigating the theory to explain the performance enhancements we have shown would be an interesting next step.

7. REFERENCES

[1] L. Wang, N. Kitaoka, and S. Nakagawa, "Robust distant speech recognition by combining multiple microphone-array processing with position-dependent CMN," *EURASIP*, pp. 1–11, 2006.

[2] M. Brandstein and D. Ward, Eds., *Microphone arrays: signal processing techniques and applications*, Springer, 2001.

[3] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, Springer, 2001.

[4] X. Huang, A. Acero, and H. Hsiao-Wuen, *Spoken language processing: a guide to theory, algorithm, and system development*, Prentice-Hall PTR, 2001.

[5] B. Smith, "Feature extraction for transient acoustic event classification," M.S. thesis, University of Washington, 2006.

[6] J. L. Sanchez-Bote, J. Gonzalez-Rodriguez, and J. Ortega-Garcia, "A new approach to dereverberation and noise reduction with microphone arrays," *Proc. EUSIFCO*, pp. 183–186, 2000.

[7] Q. Liu, B. Champagne, and P. Kabal, "Room speech dereverberation via minimum-phase and all-pass component processing of multi-microphone signals," *IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing*, pp. 571–574, 1995.

[8] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: a review," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37, 2000.

[9] M. R. Gupta, R. M. Gray, and R. A. Olshen, "Nonparametric supervised learning by linear interpolation with maximum entropy," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 28, pp. 766–781, 2006.

[10] M. P. Friedlander and M. R. Gupta, "On minimizing distortion and relative entropy," *IEEE Trans. on Information Theory*, vol. 52, no. 1, pp. 238–245, 2006.

[11] S. L. Salzberg, "On comparing classifiers: pitfalls to avoid and a recommended approach," *Data Mining and Knowledge Discovery*, vol. 1, pp. 317–328, 1997.