Real-Time Speaker Verification with a Microphone Array

Gang Mei, Roger Xu, Debang Lao and
Chiman Kwan
Intelligent Automation, Inc.
Rockville, MD, USA

Vincent Stanford
National Institute of Standards and
Technology
Gaithersburg, MD, USA

Abstract - Real-time speaker verification, with speech acquired using the NIST Mk-III microphone array and an autodirective beamforming algorithm, is demonstrated. The software and hardware backbone of the demonstration is the NIST Smart Flow System and Mk-III Array, both developed by National Institute of Standards and Technology in support of multimodal research communities. A microphone array acquires speech signals; a steered response beamformer calculates the direction of arrival (DoA) of the dominant signal; and a speaker verification component determines whether the signal is speech from a specific privileged speaker. If so, a camera will slew to the DoA of the privileged speaker; but not other speakers, or other kinds of sound. Novel approaches were taken to the design of the basic components to obtain good realtime demonstration performance.

Keywords: Speaker Verification, Smart Flow, Gaussian Mixture Model

1.0 Introduction

The Smart Flow System developed by NIST provides a platform to support standards that promote the interoperability of multimodal sensing devices and classification algorithms produced by different manufacturers. The NIST system can acquire, transport, time tag, and archive, multiple sensor data streams, such as voice and video in real-time, for subsequent distributed processing. Many signal, image, and speech processing applications such as beamforming, speaker verification, face identification, or head trackers, can be implemented within the framework of the NIST Smart Flow System.

Under the support of a NIST Small Business Innovative Research grant, we (Intelligent Automation Incorporated) developed proof of concept for a user sensitive interface that performs real time speaker verification. In this system, the two most important components include an improved delay-and-sum beamformer and a state machine based speaker verification algorithm.

The beamforming algorithm has the following features: first, it is real time; second, it compensates near-field effects; third, it is stable; and fourth, it provides good tracking performance. Taken together, these comprise a novel and unique implementation for autodirective speech acquisition.

The speaker verification algorithm we have developed segments speech signals using a state machine. Compared with the energy threshold based approach, the state machine-based speech segmentation has faster tracking performance in real time verification; less training time is required; and modeling is more accurate.

Previously, we implemented a Gaussian Mixture Model based speaker verification algorithm. Realtime tests proved the effectiveness of this basic algorithm at a fixed speaker bearing using microphone-acquired speech. We improved the verification performance and system flexibility with several design upgrades. First, we enhanced the speech signals from the beamformer with the purpose of reducing the residual noise even further from that of a far-field delay and sum beamformer. Second, we improved the speaker verification performance via discriminative training of the population model. As a result, the privileged speaker's voice can more precisely match the corresponding model. Third, we performed speaker verification using utterance level segmentation in order to achieve high performance.

We will first present an overview of the demonstration system in Sec. 2.0. An improved near-field delay-and-sum beamforming algorithm will be then described in Sec. 3.0. The speaker verification algorithm will be explained in Sec. 4.0. In the last section, conclusions and future directions will be given.

2.0 Overview of the Demonstration System

The NIST system can acquire and process multiple sensor data streams, such as voice and image, in real-time. Here we use it to construct a proof-of-concept of the *user sensitive interface* proposed by Stanford in [1] and discussed further by Flanagan and Stanford in [2]. The data flow diagram of the demonstration system is shown in Fig. 1. To verify the performance of the speaker identification and bearing estimation algorithms, a camera steered to the privileged speaker is used. The *Visca camera control* client, shown in the diagram, controls the orientation of the pan-tilt camera. The objective of the demo is to control the camera so as to track a privileged speaker based on his or her voice as he or she moves through the room. When the privileged speaker moves while he she is speaking, the camera will follow him or her. When non-privileged speakers talk, the camera doesn't respond. Moreover, the system blocks all non-privileged speakers' speech and only forwards the privileged speaker's speech signals.

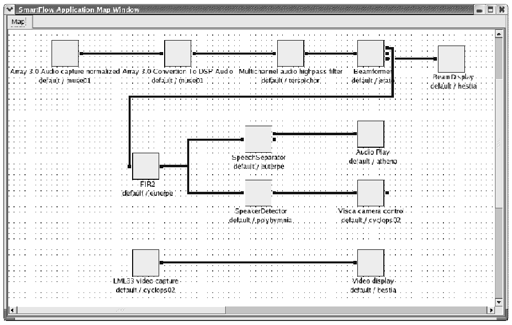

Fig. 1: Demonstration system as shown in a NIST Smart Flow Application Map.

Several software clients are involved in the demonstration system as shown in Fig. 1. The *Array 3.0 Audio Capture Normalized* client receives data packets from the microphone array. It also performs normalization of the data, which is required by the beamformer client. According to the specifications of the Panasonic microphones used in the array, the gain of the microphones can vary in the range of 6 dB. Therefore, microphone signals for all channels must be normalized to compensate these normal gain variations. The *Array 3.0 Conversion to DSP Audio* client converts the output data format of the data capture client to from 24bit to 16 bit for the downstream clients in the Smart Flow System. The *Multichannel audio highpass filter* client is a 64-channel high pass IIR filter. The *Beamformer* client calculates the power distribution versus directions, or steered response, based on several frames of the input data. It also makes decisions about whether the maximum power direction is the direction of interest. Due to the random and intermittent nature of speech signals, it is a challenging task to select the direction of the sound of interest. Many modifications have been made to this client and several techniques have been implemented to improve the robustness and response time of the client. The *BeamDisplay* client displays the beamspace power distribution calculated by the Beamformer client for visualization purposes. The *SpeechSeparator* client separates the speech of the primary speaker from other sound while the SpeakerDetector client detects the primary speaker and controls the camera. Both clients consist of speech activity detection, feature extraction and speaker verification. The speaker verification client we developed is based on state machines that identify speech onset and offset points. The *Camera control* client is the interface between the system and camera. It receives position information from the speaker detection client and uses the information to slew the camera.

In the following two sections we will present details of the improved delay-and-sum beamforming algorithm and the state machine based speaker verification algorithm.

3.0 An Improved Delay-and-Sum Beamforming Algorithm

The default beamforming algorithm in the NIST Smart Flow System is very basic and was intended only to test the data flow clients and illustrate the principals of programming of the data flow system. It scans a range bearing angles under the assumption that incoming signal is a plane wave, i.e. originating in the far field, and chooses the bearing angle which gives the maximum averaged signal energy.

However, for normal conference room applications, the speakers usually stand within a few meters at most of the microphone array, so the near field effect is no longer negligible given the fact that the microphone array is about 1.3 meters in length. This often puts the speakers within three diameters of the array. Therefore, near field compensation becomes necessary, and will improve the beamforming performance significantly. Beams of typical human speech are shown in Fig. 2. Our new beamforming algorithm with near field compensation and hierarchy scanning includes the following steps:

- Initialization: Calculate the theoretical delay of all sensors for all possible angles at some selected distance interval using simple geometry, and round the result to units of samples.

- Real-time beamforming: For a segment of new data, generate 10 beams using 15 degrees interval from -69 degree to 69 degree at the infinite distance (plane wave) by first shifting 64 channels by corresponding delay value in the delay array then sum them together.

- Pick the beam with the maximum energy as the course DoA.

- Scan again using 3 degrees interval from DoA-9 degree to DoA+9 degree at the infinite distance (plane wave), and pick the beam with the maximum energy as the accurate DoA.

- Given the DoA, scan though different distance cases to pick the beam with maximum energy as the course distance.

- Calculate the theoretical delay at current DoA and around the course distance value using finer distance interval, then scan through these finer intervals to pick the accurate distance.

- Generate the final near field beamforming output with the accurate DoA and accurate distance.

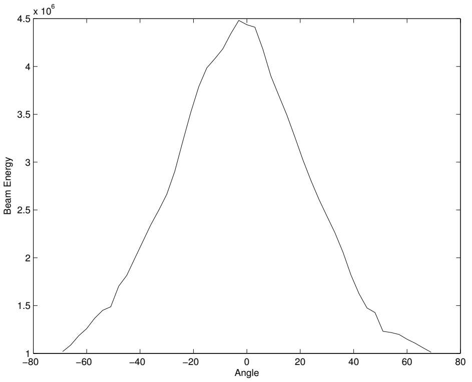

Fig. 2: Beam energy vs. Angle

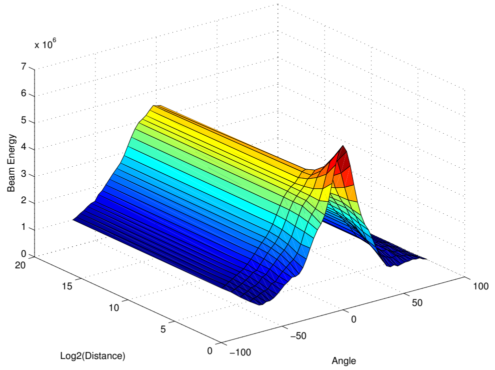

Fig. 3: Beam Energy vs. Angle & Distance

To demonstrate the near field compensation scheme, we recorded a segment of speech at nearfield proximity to the array. For this recorded speech, Fig. 3 shows the beam energy at different angles and distances. We can see that without near field compensation, we will not be able to achieve the maximum beam energy at the closer distance.

The difference in the beamforming output signal is shown in Fig. 4 and Fig. 5, where we can see that after near field compensation, the SNR (signal to noise ratio) is increased. For the input signal shown in Fig. 4 and Fig. 5, SNR is increased from 36.2dB to 38.4dB, and 2.2dB gain is obtained from the near field compensation alone. The SNR of the single channel signal is 30.1dB.

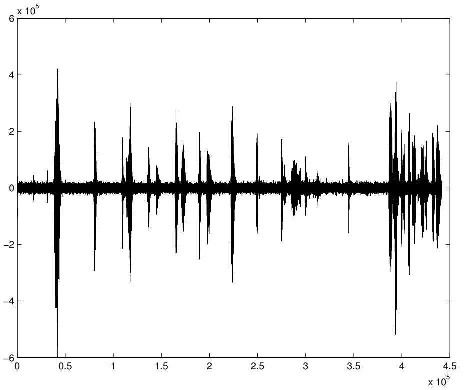

Fig. 4: Beamforming output without near field compensation

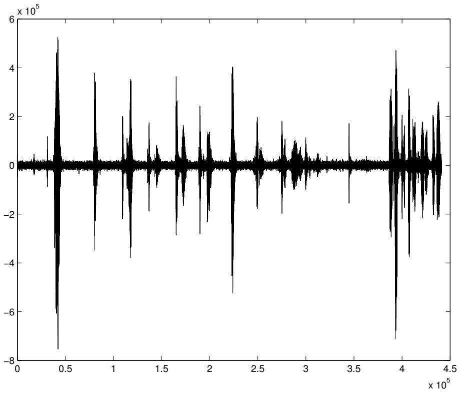

Fig. 5: Beamforming output with near field compensation

4.0 A State Machine-based Speaker Verification Algorithm

Gaussian Mixture Models (GMMs) are commonly used for speaker verification applications. In our demonstration, two GMM models are trained and used for speaker verification. One model is trained by the privileged speaker's speech data and the other, called the population GMM, is trained by speech data of a group of male and female speakers. In speaker verification, cepstral coefficients of acquired speech signals are scored against these two models. If the privileged speaker's GMM yields the higher likelihood, the signal is imputed to come from the privileged speaker.

Our state machine-based speaker verification algorithm includes a state machine based speech segmentation algorithm, discriminative training of the world GMM models, and a novel decision making algorithm.

4.1 State machine based speech segmentation algorithm

In order to accurately train the GMM models for various speakers, we should use the valid speech data only and exclude the background noise and the silence from the training data. In a previous application [3], we used an energy thresholding algorithm to separate the speech signal from the background noise. The energy threshold algorithm will consider the incoming frame as speech if and only if the signal energy of this frame exceeds a preset threshold times the background noise energy level.

The energy thresholding algorithm has two major disadvantages. First, in order to exclude the background noise from the speech data, the threshold has to be set quite high. As a result, only the speech data with the highest energy is considered as speech, and much of the speech with low energy is lost. Second, discontinuous speech segments caused by energy thresholding will adversely affect the calculation of first- and second-order cepstral coefficients.

Therefore, in the current version of *SpeakerDetector* and *SpeechSeparator* clients, we use a state machine based speech segmentation algorithm to extract the whole utterances from the speech signal. Three states are defined: speech, sub-speech and silence. The speech state represents the signal period with the highest energy, the silence state represents the signal period with the lowest energy, and the sub-speech state represents the transition period from the silence state to the speech state. The corresponding state transition diagram is shown in Fig. 6.

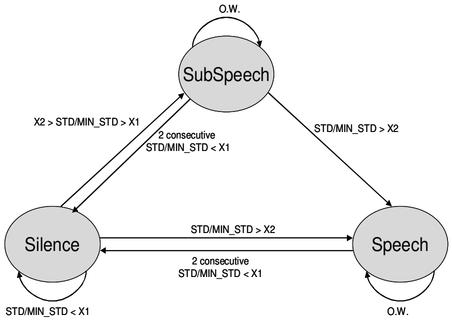

Fig. 6: State transition diagram for state machine based speech segmentation algorithm

The detailed procedure of the algorithm is as follows:

- If the standard deviation of one frame is more than X1 (a preset threshold) times the back ground noise level, the state is changed from the initial silence state to the sub-speech state and this frame is copied to the speech buffer.

  If the standard deviation of one frame is more than X2 (a preset threshold > X1) times the back ground noise level, the state is changed from the initial silence state directly to the speech state and this frame is copied to the speech buffer.

- If current state is sub-speech state and the standard deviation of two consecutive frames are both below X1 times of back ground noise level, the state is changed back to the silence state and the speech buffer is cleared without any decision making.

  If current state is sub-speech state and the standard deviation of the incoming frame is more than X2 times the back ground noise level, the state is changed to the speech state and this frame is copied to the speech buffer. Otherwise, the state remains sub-speech state and the incoming frame will be copied to the speech buffer.

- If current state is speech state and the standard deviation of two consecutive frames are both below X1 times of back ground noise level, the state is changed back to the silence state and the speech buffer is cleared after all data is sent to the decision making function. Otherwise, the state remains speech state and the incoming frame will be copied to the speech buffer.

Fig. 7 shows the DET curve showing the performance of state machine based speech segmentation algorithm and engery thresholding algorithm for the same set of test data.

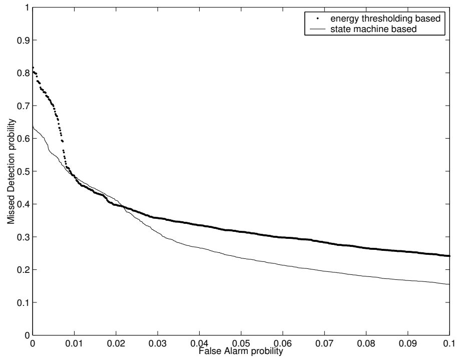

Fig. 7: Performance comparison of thresholding vs. state machine based speech segmentation algorithm

4.2 The discriminative training of the world GMM model

Prior to this work, the composition of the speech data used to train the population GMM model is independent of the selection of the privileged speaker and the distribution of speech data from different speakers is uniform. The basic architecture of this technique is described in [4,5].

Nevertheless, empirically it is very common that a particular speaker in the population model is easier to trigger the false alarm in the speaker verification system than other speakers due to the similarity of his/her voice to the privileged speaker's. Therefore, although the uniformly distributed population model described above is simple and intuitive, it may not have the optimal performance.

In order to boost the verification performance we use a discriminative training method to train the population model. The discriminative training method reorganizes the distribution of the population model by increasing the percentage of speakers' training data, which is more similar to the selected privileged speaker's data. Fig. 8 shows the performance improvement introduced by the reorganization of the population model compared to the uniformly distributed population model.

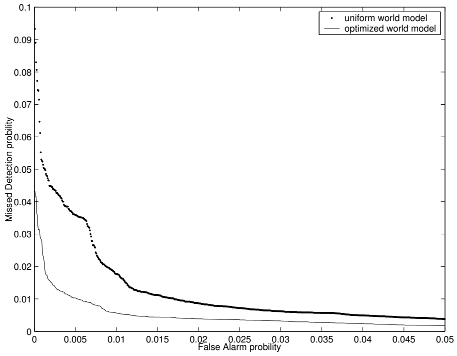

Fig. 8: Performance comparison of optimized world model vs. uniform world model

4.3 The novel decision-making algorithm

In order to decrease the false alarm rate without sacrificing the response time, we only perform speaker verification at the boundary of separate utterances based on the assumption that each utterance can only come from one speaker. Thus, after each new utterance generated from the speech segmentation algorithm, we check the total size of the speech buffer. If it is big enough, we will make one verification decision.

In addition, we added one level of filtering to the speaker verification decisions described above. We define an integer number with an initial value of 0. With each positive raw speaker verification decision (privileged speaker), this number will increase by 1 until it reaches a preset upper limit. Once the upper limit is reached, the decision-making algorithm will make a final positive decision. For each negative raw speaker verification decision, this number will decrease by 1 until it reaches a preset lower limit. Once the lower limit is reached, the decision-making algorithm will make a final negative decision. When the privileged speaker talks continuously, the integer number will remain at the upper limit, and for each new raw positive decision, our decision-making algorithm will make another final positive decision. Therefore, the response time will be satisfactory. If a non-privileged speaker is speaking, continuous raw negative decisions will make the integer number remain at the lower limit. Thus, even once a while we have a wrong raw decision, our decision-making algorithm will not make positive final decision. Therefore, false alarms can be mitigated

5.0 Conclusion and Future Directions

In this paper, we have presented a real-time speaker verification demonstration based on the NIST Smart Flow System infrastructure with improved signal acquisition and classification algorithms. The demonstration involves development of several novel speech processing algorithms. The experimental results have shown that the improved beamforming algorithm with near field compensation increases the SNR by about 2.2 dB over a basic far-field delay and sum beamformer. The state machine-based speaker verification with discriminative training results in faster responses and lower false alarm rates.

The following are some highlights of future research directions:

- *Noise floor reduction for the microphone array.* An improved microphone board has been contributed by the EU CHIL project and has shown a significantly lower noise floor.
- *Improvement of the DOA estimation algorithm.* Higher resolution beamforming methods may be helpful in this regard, as may minimum power adaptive beamformers.
- *Source separation.* Significant progress has been made in blind source separation and array beamforming could be combined in some way with existing techniques of source separation.
- *Explorations of different microphone array geometries.* It is possible that linear arrays, while manageable are not the optimal configuration.
- *Integration of speech recognition with the Smart Flow System.* The modular data flow architecture of the NIST Data Flow system will accommodate improved algorithms and new capabilities.

These and other improvements may make it feasible to create a new generation of computer interface that can respond to specific individuals in a context sensitive manner.

Acknowledgements

The authors from Intelligent Automation wish to acknowledge support by the National Institute of Standards Information Technology Laboratory, and its Technology Small Business Innovative Research Program, under contract number: SB1341-02-W-1140 for this work [6].

Disclaimer

Employees of the Federal Government, in the course of their official duties, developed the NIST Smart Data Flow software at the National Institute of Standards and Technology. Pursuant to title 17 Section 105 of the United States Code this software is not subject to copyright protection and is in the public domain. Commercial products may have been identified in order to adequately specify or describe the subject matter of this work. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products identified are necessarily the best available for the purpose. The Smart Data Flow is an experimental system. NIST assumes no responsibility whatsoever for its use by other parties, and makes no guarantees, expressed or implied, about its quality, reliability, or any other characteristic. The National Institute of Standards and Technology and the Smart Space Project would appreciate acknowledgements from those using the tools.

References

[1] V. Stanford. "Smart Space Scenario"; Proceedings of the 1998 DARPA/NIST Smart Spaces Workshop, July 30-31, Gaithersburg, MD (1998) 1.1-1.2

[2] J. Flanagan and V. Stanford. Situation Awareness in Smart Spaces. Proceedings of the 1998 DARPA/NIST Smart Spaces Workshop, July 30-31, 1998, Gaithersburg, MD (1998) 3.1-3.13

[3] R. Xu, G. Mei, Z. Ren, and C. Kwan. "A Real Time Speaker Verification Demonstration on the Smart Flow System" 2004 International Symposium on Intelligent Multimedia, Video & Speech Processing (ISIMP 2004), Hong Kong, October 2004

[4] D. A. Reynolds and R. C. Rose. "Robust Text-Independent Speaker Verification Using Gaussian Mixture Speaker Models," IEEE Trans. Speech and Audio Processing, vol. 3, no. 1, 1995.

[5] Q. Li, B.H. Juang, Q. Zhou, and C. Lee. "Automatic Verbal Information Verification for User Authentication"; IEEE Trans. Speech and Audio Processing, vol. 8, No. 5, 2000.

[6] C. Kwan et al., Phase 2 final report for NIST SB1341-02-W-1140 SBIR contract, submitted to NIST, August 2005.