# Probabilistic Fusion of Sorted Score Sequences for Robust Speaker Verification

Ming-Cheung Cheung[1], Man-Wai Mak[1] and Sun-Yuan Kung[2]

[1] Center for Multimedia Signal Processing,
Dept. of Electronic and Information Engineering,
The Hong Kong Polytechnic University, Hong Kong
[2] Department of Electrical Engineering, Princeton University, USA

**Abstract.** Fusion techniques have been widely used in multi-modal biometric authentication systems. While these techniques are mainly applied to combine the outputs of modality-dependent classifiers, they can also be applied to fuse the decisions or scores from a single modality. The idea is to consider the multiple samples extracted from a single modality as independent but coming from the same source. In this chapter, we propose a single-source, multi-sample data-dependent fusion algorithm for speaker verification. The algorithm is data-dependent in that the fusion weights are dependent on the verification scores and the prior score statistics of claimed speakers and background speakers. To obtain the best out of the speaker's scores, scores from multiple utterances are sorted before they are probabilistically combined. Evaluations based on 150 speakers from a GSM-transcoded corpus are presented. Results show that data-dependent fusion of speaker's scores is significantly better than the conventional score averaging approach. It was also found that the proposed fusion algorithm can be further enhanced by sorting the score sequences before they are probabilistically combined.

**Keywords**: Decision Fusion; Speaker Verification; Feature Transformation; GSM-Transcoded Speech.

## 1 Introduction

In recent years, research has focused on using fusion techniques to improve the performance of speaker verification systems. One popular approach is to fuse the scores obtained from modality-specific classifiers. For example, in [1][2] the scores from a lip recognizer are fused with those from a speaker recognizer, and in [3] a face classifier is combined with a voice classifier using a variety of combination rules. These types of systems, however, require multiple sensors, which tend to increase system costs and require extra cooperation from users, e.g. users may need to present their faces as well as to utter a sentence to support their claim. While this requirement can be alleviated by

fusing different speech features from the same utterance [4], the effectiveness of this approach relies on the degree of independence among these features.

This chapter investigates the fusion of scores from multiple utterances to improve the performance of speaker verification from GSM-transcoded speech. The simplest way to achieve this goal is to average the scores obtained from multiple utterances, as in [5]. While score averaging is a reasonable approach to combining the scores, the approach weighs the contribution of speech patterns from multiple utterances equally, which may not produce optimal fused scores. In our previous work [6][7], we computed the optimal fusion weights based on the score distribution of the utterances and on the prior score statistics determined from enrollment data. To further enhance the fusion algorithm, we propose in this chapter to sort the score sequences before fusion takes place. With this arrangement, the contribution of some erroneous scores of one utterance can be compensated by the scores of another utterance. Compared with the conventional equal-weight approach, the new algorithm is able to reduce the equal error rate by 23%.

The remainder of the chapter is organized as follows. In Section 2, the data-dependent decision fusion algorithm proposed in [6] is briefly reviewed and the proposed score sorting approach is introduced. This is followed by a theoretical analysis demonstrating the benefit of the proposed score sorting approach. The proposed method is further evaluated in Section 3 via a speaker verification experiment using GSM-transcoded speech. Finally, in Section 4, concluding remarks are provided.

## 2 Data-Dependent Decision Fusion Model

### 2.1 Architecture

Assume that $K$ streams of features vectors (e.g. MFCCs) can be extracted from $K$ independent utterances $\mathcal{U} = \{\mathcal{U}_1, \ldots, \mathcal{U}_K\}$. Let us denote the observation sequence corresponding to utterance $\mathcal{U}_k$ by

$$\mathcal{O}^{(k)} = \{\mathbf{o}_t^{(k)} \in \Re^D; t = 1, \ldots, T_k\} \qquad k = 1, \ldots, K \tag{1}$$

where $D$ and $T_k$ are respectively the dimensionality of $\mathbf{o}_t^{(k)}$ and the number of observations in $\mathcal{O}^{(k)}$. We further define a normalized score function

$$s(\mathbf{o}_t^{(k)}; \Lambda) \equiv \log p(\mathbf{o}_t^{(k)}|\Lambda_{\omega_c}) - \log p(\mathbf{o}_t^{(k)}|\Lambda_{\omega_b}) \tag{2}$$

where $\Lambda = \{\Lambda_{\omega_c}, \Lambda_{\omega_b}\}$ contains the Gaussian mixture models (GMMs) characterizing the client speaker ($\omega_c$) and the background speakers ($\omega_b$), and $\log p(\mathbf{o}_t^{(k)}|\Lambda_\omega)$ is the output of $\Lambda_\omega$, $\omega \in \{\omega_c, \omega_b\}$, given observation $\mathbf{o}_t^{(k)}$.

In [6], frame-level fused scores are computed as

$$s(\mathbf{o}_t^{(1)}, \ldots, \mathbf{o}_t^{(K)}; \Lambda) = s(\mathbf{O}_t; \Lambda) = \sum_{k=1}^{K} \alpha_t^{(k)} s(\mathbf{o}_t^{(k)}; \Lambda) \tag{3}$$

$$\alpha_t^{(k)} \in [0, 1] \text{ and } \sum_{k=1}^{K} \alpha_t^{(k)} = 1,$$

where $\mathbf{O}_t = \{\mathbf{o}_t^{(1)}, \ldots, \mathbf{o}_t^{(K)}\}$ contains the $K$ observations from the $K$ utterances at time $t$ and $\alpha_t^{(k)}$ represents the confidence (reliability) of the observation $\mathbf{o}_t^{(k)}$.

During enrollment, the mean score of each client speaker ($\tilde{\mu}_c$) and of the background speakers ($\tilde{\mu}_b$) are determined. Then, the prior score and prior variance are respectively computed as follows:

$$\tilde{\mu}_p = \frac{K_c \tilde{\mu}_c + K_b \tilde{\mu}_b}{K_c + K_b} \qquad \text{and} \qquad \tilde{\sigma}_p^2 = \frac{1}{K_c + K_b} \sum_{n=1}^{K_c+K_b} \left[ s(\tilde{\mathcal{O}}^{(n)}; \Lambda) - \tilde{\mu}_p \right]^2 \tag{4}$$

where $s(\tilde{\mathcal{O}}^{(n)}; \Lambda) = \frac{1}{T_n} \sum_{t=1}^{T_n} s(\tilde{\mathbf{o}}_t^{(n)}; \Lambda)$ is the mean score of the $n$-th training utterance and $K_c$ and $K_b$ are respectively the numbers of client speaker's utterances and background speakers' utterances. Then, during verification, the claimant is asked to utter $K$ utterances, and the data-dependent fusion weights are computed as:

$$\alpha_t^{(k)} = \frac{\exp\left\{ [s_t^{(k)} - \tilde{\mu}_p]^2 \Big/ 2\tilde{\sigma}_p^2 \right\}}{\sum_{l=1}^{K} \exp\left\{ [s_t^{(l)} - \tilde{\mu}_p]^2 \Big/ 2\tilde{\sigma}_p^2 \right\}} \qquad k = 1, \ldots, K. \tag{5}$$

where for ease of presentation, we have defined $s_t^{(k)} \equiv s(\mathbf{o}_t^{(k)}; \Lambda)$. The mean fused score

$$s(\mathcal{U}; \Lambda) = \frac{1}{T} \sum_{t=1}^{T} s(\mathbf{O}_t; \Lambda) \tag{6}$$

is compared against a decision threshold for decision making. Fig. 1 depicts the architecture of the fusion model.

Note that this method requires the $K$ utterances to contain the same number of feature vectors, i.e. $T_k = T \; \forall \; k = 1, \ldots, K$. If it is not the case, we may move some of the vectors from the tail of the longer utterances to the tail of the shorter utterances to make the number of vectors in all utterance equal.[3]

[3] As it is likely that the utterances are obtained from the same speaker under the same environment in a verification session, moving feature vectors from utterances to utterances will have the same effect as partitioning a long utterance into a number of equal-length short utterances.

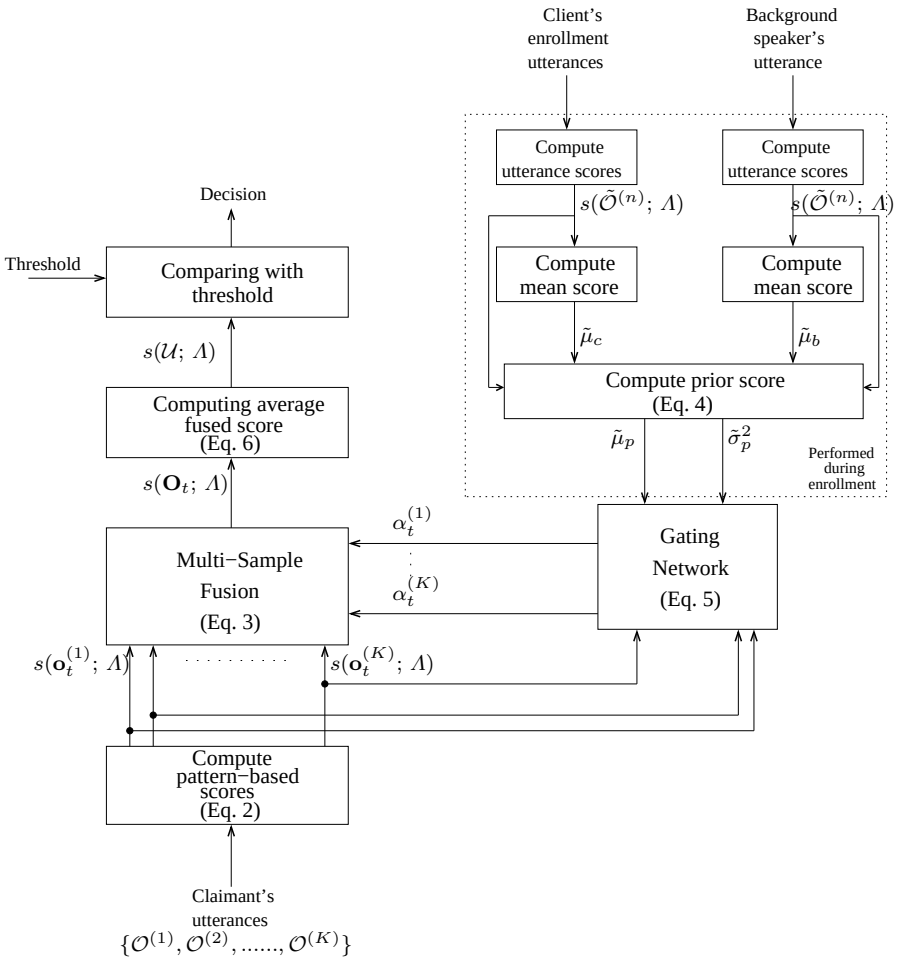



**Fig. 1.** Architecture of the multi-sample decision fusion model.

### 2.2 Gaussian Example

Fig. 2 illustrates an example where the distributions of the client speaker scores and the impostor scores are assumed to be Gaussian. It is also assumed that both the client and impostor utter two utterances. The client speaker's mean scores for the first and second utterances are equal to 1.2 and 0.8 respectively. Likewise, the impostor's mean scores for the two utterances are equal to $-1.3$ and $-0.7$. Obviously, equal-weight fusion will produce a mean speaker score of 1.0 and a mean impostor score of $-1.0$, resulting in a score dispersion of 2.0. These two mean scores ($-1.0$ and 1.0) are indicated by the two vertical lines in Fig. 2(b). We can see from Fig. 2(b) and Fig. 2(c) that when the prior

score $\tilde{\mu}_p$ is set to a value between these two means (i.e. between the vertical lines), the scores dispersion can be larger than 2.0.

### 2.3 Theoretical Analysis

Here we provide a theoretical analysis of the fusion algorithm. Through this analysis, we will be able to explain how and why the fusion algorithm achieves better performance as compared to the equal-weight fusion approach. The reason behind the increase in the score dispersion in Fig. 2 can also be explained.

We consider the case where the score sequences of two independent utterances are fused, i.e. $K = 2$ in (3). The extension to multiple sequences is trivial. As the two utterances are independent, their scores $s_t^{(1)}$ and $s_t^{(2)}$ are also independent. Differentiating both side of (5) with respect to $s_t^{(k)}$ and using the independence between $s_t^{(1)}$ and $s_t^{(2)}$, we obtain

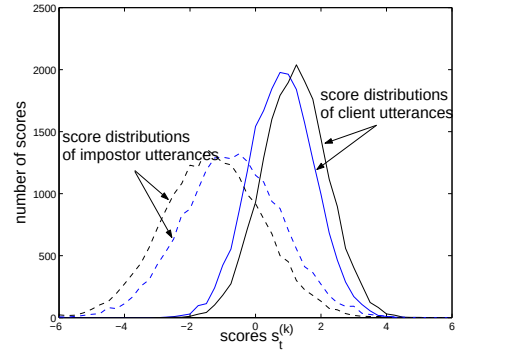
$$\frac{\partial \alpha_t^{(k)}}{\partial s_t^{(k)}} = \frac{C(s_t^{(k)} - \tilde{\mu}_p)}{\tilde{\sigma}_p^2} \left[ \frac{e^{\{(s_t^{(k)} - \tilde{\mu}_p)^2 / 2\tilde{\sigma}_p^2\}}}{\left( \sum_{l=1}^{2} e^{\{(s_t^{(l)} - \tilde{\mu}_p)^2 / 2\tilde{\sigma}_p^2\}} \right)^2} \right] \qquad k = 1, 2. \quad (7)$$

where $C = \sum_{l \neq k} e^{\{(s_t^{(l)} - \tilde{\mu}_p)^2 / 2\tilde{\sigma}_p^2\}} > 0$. Equation (7) suggests that when $s_t^{(k)} > \tilde{\mu}_p$, $\partial \alpha_t^{(k)} / \partial s_t^{(k)} > 0$, and vice versa for $s_t^{(k)} < \tilde{\mu}_p$.

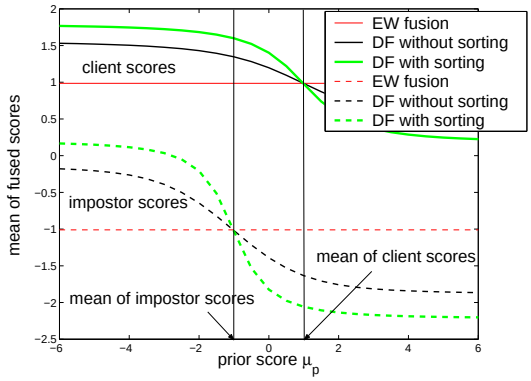
Let us consider two scenarios:

Scenario A: $\tilde{\mu}_p < \mu$, where $\mu$ is the mean score of the two utterances. For example, $\mu = 1$ for the two client utterances in Fig. 2(a). In this scenario, the claimant is more likely to be a client speaker than an impostor because the two utterances produce many large pattern-based scores to make $\mu > \tilde{\mu}_p$. Since the majority of the pattern-based scores ($s_t^{(k)}$ and $s_t^{(l)}$) are large, we have the following conditions (see Fig. 3 for an illustration):

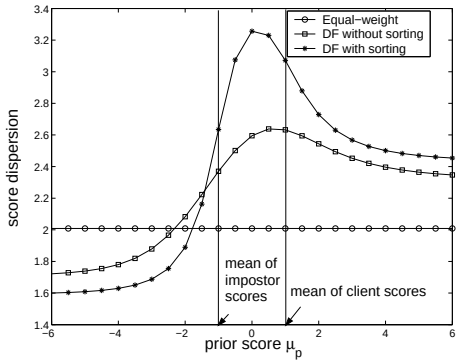
Condition A-1: $P(s_t^{(k)} > \tilde{\mu}_p) > P(s_t^{(k)} < \tilde{\mu}_p) \quad k \in \{1, 2\}$




(a)




(b)




(c)

**Fig. 2.** (a) Distributions of client scores and impostor scores as a result of two utterances: one from a client speaker and another from an impostor. The means of client scores are 0.8 and 1.2, and the means of impostor scores are $-1.3$ and $-0.7$. (b) The mean of fused client scores and the mean of fused impostor scores versus the prior score $\tilde{\mu}_p$. (c) Difference between the mean of fused client scores and the mean of fused impostor scores under different values of prior scores $\tilde{\mu}_p$'s based on equal-weight fusion and data-dependent fusion (DF) with and without score sorting.

Condition A-2:

$$
\begin{aligned}
& P(\text{emphasizing large scores}) \\
= {} & P(\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3) \\
= {} & P(\{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p > \tilde{\mu}_p - s_t^{(l)})\} \cup \\
& \quad \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} < s_t^{(l)} - \tilde{\mu}_p)\}) \\
= {} & P(\{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} + s_t^{(l)} > 2\tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (2\tilde{\mu}_p < s_t^{(k)} + s_t^{(l)})\}) \\
> {} & P(\{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} + s_t^{(l)} < 2\tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (2\tilde{\mu}_p > s_t^{(k)} + s_t^{(l)})\}) \\
& \quad \text{from Condition A-1} \\
= {} & P(\{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p < \tilde{\mu}_p - s_t^{(l)})\} \cup \\
& \quad \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} > s_t^{(l)} - \tilde{\mu}_p)\}) \\
= {} & P(\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6) \\
= {} & P(\text{emphasizing small scores})
\end{aligned}
$$

where $P(\mathcal{S})$ stands for the probability of having the scores fall on the set $\mathcal{S}$, and its value can be obtained by integrating the 2-D Gaussian density function over the region defined by $\mathcal{S}$. As the peak of the 2-D density function falls on $\mathcal{S}1$ (see Fig. 3(b)), the volume under $\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3$ should be larger than that under $\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6$. This observation agrees with the inequality in Condition A-2.

The above argument shows that $P(\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3) > P(\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6)$. Here, we explain why $P(\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3)$ is the probability of emphasizing large scores and $P(\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6)$ is the probability of emphasizing small scores. In $\mathcal{S}1$, since both $s_t^{(k)}$ and $s_t^{(l)}$ are larger than the prior score $\tilde{\mu}_p$, (5) will emphasize the larger score only. In $\mathcal{S}2$, although $s_t^{(l)}$ is smaller than the prior score $\tilde{\mu}_p$, (5) will still emphasize the larger score (i.e. $s_t^{(k)}$ in this set) as the difference between $s_t^{(k)}$ and $\tilde{\mu}_p$ is larger than that between $s_t^{(l)}$ and $\tilde{\mu}_p$. The situation in $\mathcal{S}3$ is similar to that in $\mathcal{S}2$, except the large score is $s_t^{(l)}$ and the small score is $s_t^{(k)}$; again the larger score $s_t^{(l)}$ is emphasized because it is further away from $\tilde{\mu}_p$ than $s_t^{(k)}$ is. Therefore, by merging these three sets together, we can obtain

the probability of emphasizing large scores. Similar arguments can be applied to $\mathcal{S}4$, $\mathcal{S}5$ and $\mathcal{S}6$ to obtain the probability of emphasizing the small scores.

Scenario A suggests that when the majority of scores are greater than the prior score $\tilde{\mu}_p$, the fusion algorithm has a higher chance of emphasizing large scores. Meanwhile (7) suggests that if $s_t^{(k)}$ increases, the fusion weight $\alpha_t^{(k)}$ for the corresponding score will also increase (because $\partial\alpha_t^{(k)}/\partial s_t^{(k)} > 0$). Putting these two observations together suggests that the mean fused score should be larger than the mean scores of the two utterances.

Scenario B: $\tilde{\mu}_p > \mu$, where $\mu$ is the mean score of the two utterances. For example, $\mu = -1$ for the two impostor utterances in Fig. 2(a). In this scenario, the claimant is more likely to be an impostor because the two utterances produce many small pattern-based scores to make $\mu < \tilde{\mu}_p$. Since the majority of the pattern-based scores ($s_t^{(k)}$ and $s_t^{(l)}$) are small, we have the following conditions (see Fig. 4 for an illustration):

Condition B-1: $P(s_t^{(k)} < \tilde{\mu}_p) > P(s_t^{(k)} > \tilde{\mu}_p) \quad k \in \{1, 2\}$

Condition B-2:

$$
\begin{aligned}
& P(\text{emphasizing small scores}) \\
&= P(\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6) \\
&= P(\{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p)\} \cup \\
&\qquad \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p < \tilde{\mu}_p - s_t^{(l)})\} \cup \\
&\qquad \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} > s_t^{(l)} - \tilde{\mu}_p)\}) \\
&> P(\{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p)\} \cup \\
&\qquad \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p > \tilde{\mu}_p - s_t^{(l)})\} \cup \\
&\qquad \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} < s_t^{(l)} - \tilde{\mu}_p)\}) \\
&\qquad \text{from Condition B-1} \\
&= P(\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3) \\
&= P(\text{emphasizing large scores})
\end{aligned}
$$

where $P(\mathcal{S})$ stands for the probability of having the scores fall on the set $\mathcal{S}$, and its value can be obtained by integrating the 2-D Gaussian density function over the region defined by $\mathcal{S}$. As the peak of the 2-D density function falls on $\mathcal{S}4$ (see Fig. 4(b)), the volume under $\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6$ should be larger than that under $\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3$. This observation agrees with the inequality in Condition B-2.

Scenario B suggests that when the majority of scores are smaller than the prior score $\tilde{\mu}_p$, the fusion algorithm has a higher chance of emphasizing

small scores. At the same time, we can observe from (7) that when $s_t^{(k)}$ decreases, the fusion weight $\alpha_t^{(k)}$ for the corresponding score will also increases (because $\partial\alpha_t^{(k)}/\partial s_t^{(k)} < 0$). These two observations lead to the conclusion that the mean fused score of the two utterances should be smaller than the utterances' mean score.

Based on the above analysis, we can see that if the claimant is more likely to be a client speaker, the fusion algorithm will increase his/her mean fused score and vice versa if he/she is an impostor. This has the effect of increasing the score dispersion, as demonstrated in Fig. 2(c).

### 2.4 Fusion of Sorted Scores

As the proposed fusion algorithm depends on the pattern-based scores of individual utterances, the positions of scores in the score sequence also affect the final fused scores. Moreover, as illustrated in Section 2.3, the emphasis of large speaker scores under Scenario A and the de-emphasis of small impostor scores under Scenario B are probabilistic, i.e. there is no guarantee that these situations will always occur. In order to overcome this limitation, we have proposed to sort scores before fusion so that small scores will always be fused with large scores [8].

Here, we provide a theoretical analysis to explain the benefit of sorting the scores before fusion. We assume that there are two sorted score sequences with equal mean ($\mu$), one being arranged in ascending order and the other in descending order. We further assume that the scores in the sequences follow a Gaussian distribution. If the numbers of scores in the sequences are sufficiently large, we can obtain the following relationship:

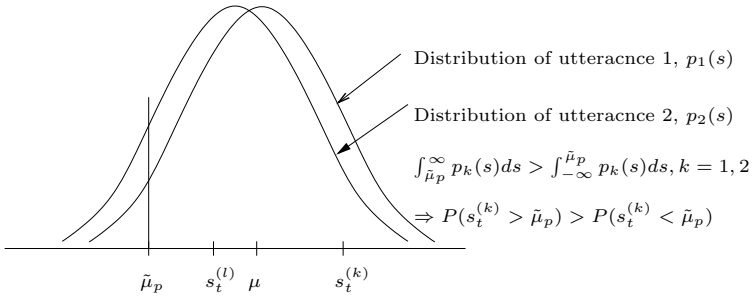
$$\mu - s_t^{(1)} \approx s_t^{(2)} - \mu, \text{ i.e., } s_t^{(2)} \approx 2\mu - s_t^{(1)} \tag{8}$$

where $s_t^{(1)}$ and $s_t^{(2)}$ respectively represent the scores less than and greater than the score mean $\mu$. Without loss of generality, we denote the smaller score as $s_t^{(1)}$ and the larger one as $s_t^{(2)}$, i.e. $s_t^{(1)} < s_t^{(2)}$. Substituting (8) into (5), the fusion weight for the small scores $s_t^{(1)}$ can be expressed as

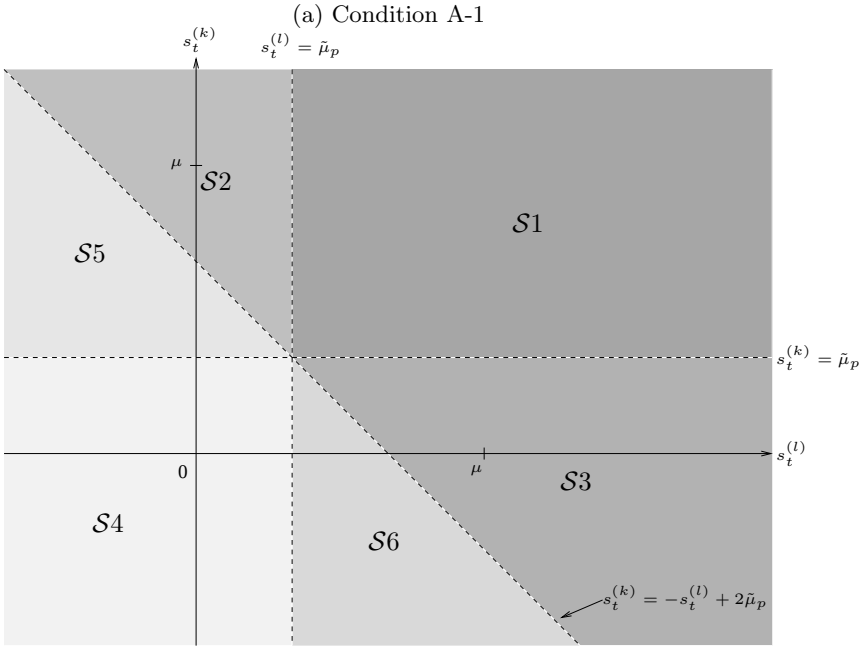
$$\begin{aligned}\alpha_t^{(1)} &= \frac{\exp\{(s_t^{(1)} - \tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}{\sum_{l=1}^{2}\exp\{(s_t^{(l)} - \tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} \\ &= \frac{\exp\{(s_t^{(1)} - \tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}{\exp\{(s_t^{(1)} - \tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\} + \exp\{(2\mu - s_t^{(1)} - \tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}.\end{aligned} \tag{9}$$

Differentiate both side of (9) with respect to $s_t^{(1)}$, we obtain

$$\frac{\partial\alpha_t^{(1)}}{\partial s_t^{(1)}} = \frac{\frac{2(\mu-\tilde{\mu}_p)}{\tilde{\sigma}_p^2} e^{\{(s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} e^{\{(2\mu-s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}}{\left[e^{\{(s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} + e^{\{(2\mu-s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right]^2}.$$



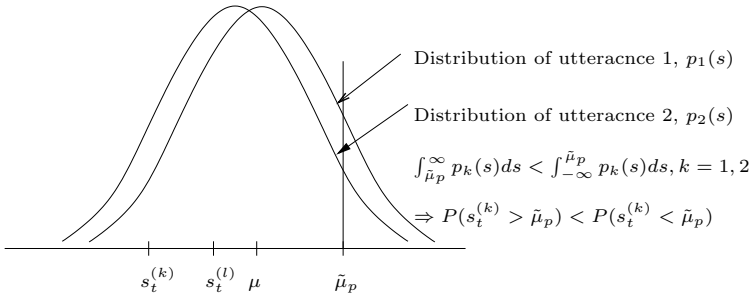

(a) Condition A-1



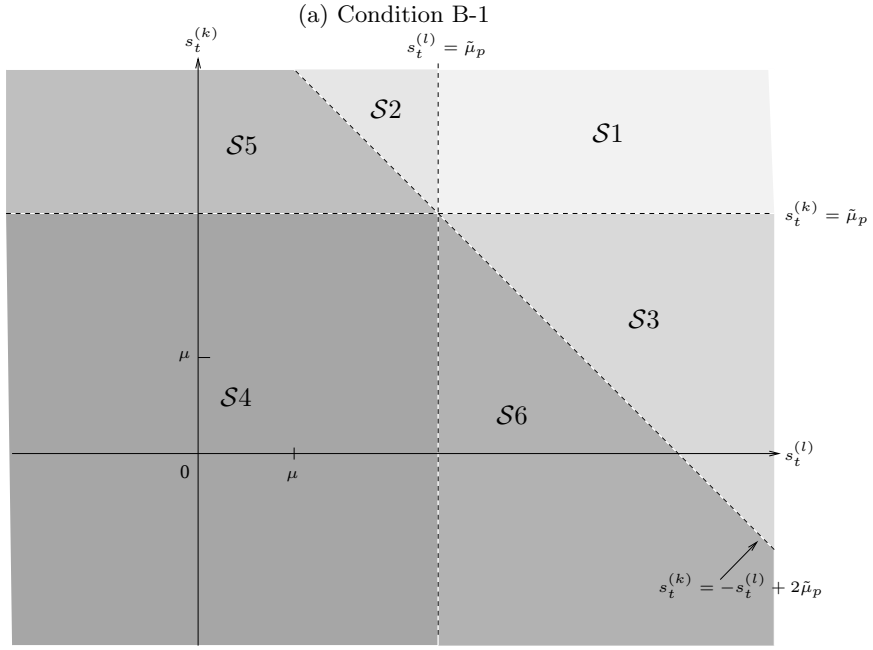

$\mathcal{S}1 = \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p)\}$
$\mathcal{S}2 = \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p > \tilde{\mu}_p - s_t^{(l)})\}$
$\mathcal{S}3 = \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} < s_t^{(l)} - \tilde{\mu}_p)\}$
$\mathcal{S}4 = \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p)\}$
$\mathcal{S}5 = \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p < \tilde{\mu}_p - s_t^{(l)})\}$
$\mathcal{S}6 = \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} > s_t^{(l)} - \tilde{\mu}_p)\}$

(b) Condition A-2

**Fig. 3.** Illustration of the two conditions in Scenario A ($\tilde{\mu}_p < \mu$). (a)$P(s_t^{(k)} > \tilde{\mu}_p) > P(s_t^{(k)} < \tilde{\mu}_p)$; (b)$P$(emphasizing large scores) $> P$(emphasizing small scores).




(a) Condition B-1




$$\mathcal{S}1 = \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p)\}$$
$$\mathcal{S}2 = \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p > \tilde{\mu}_p - s_t^{(l)})\}$$
$$\mathcal{S}3 = \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} < s_t^{(l)} - \tilde{\mu}_p)\}$$
$$\mathcal{S}4 = \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p)\}$$
$$\mathcal{S}5 = \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p < \tilde{\mu}_p - s_t^{(l)})\}$$
$$\mathcal{S}6 = \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} > s_t^{(l)} - \tilde{\mu}_p)\}$$

(b) Condition B-2

**Fig. 4.** Illustration of the two conditions in Scenario B ($\tilde{\mu}_p > \mu$). (a)$P(s_t^{(k)} > \tilde{\mu}_p) < P(s_t^{(k)} < \tilde{\mu}_p)$; (b)$P$(emphasizing large scores) $< P$(emphasizing small scores).

Therefore, we have

$$\frac{\partial \alpha_t^{(1)}}{\partial s_t^{(1)}} \begin{cases} < 0 \text{ when } \mu < \tilde{\mu}_p \ , \\ = 0 \text{ when } \mu = \tilde{\mu}_p \ , \\ > 0 \text{ when } \mu > \tilde{\mu}_p \ . \end{cases} \tag{10}$$

Similarly, we can show that

$$\frac{\partial \alpha_t^{(2)}}{\partial s_t^{(2)}} \begin{cases} < 0 \text{ when } \mu < \tilde{\mu}_p \ , \\ = 0 \text{ when } \mu = \tilde{\mu}_p \ , \\ > 0 \text{ when } \mu > \tilde{\mu}_p \ . \end{cases} \tag{11}$$

Equations (10) and (11) suggest that when $\mu < \tilde{\mu}_p$ (i.e. most of the scores are smaller than the prior score $\tilde{\mu}_p$), the fusion weights for small scores $\alpha_t^{(1)}$ increase when $s_t^{(1)}$ decreases, and the fusion weights for large scores $\alpha_t^{(2)}$ decrease when $s_t^{(2)}$ increases. This implies that (3) and (5) will emphasize small scores and thus decrease the mean fused score. In Fig. 2(b), the right vertical line represents the mean of client scores and the left vertical line the mean of impostor scores. We can notice that both the mean of the fused client scores and that of the fused imposter scores decrease when the prior score $\tilde{\mu}_p$ is greater than the respective mean, i.e. $\tilde{\mu}_p > 1.0$ for the client and $\tilde{\mu}_p > -1.0$ for the impostor. Similarly, when $\mu > \tilde{\mu}_p$ (i.e. most of the scores are larger than the prior score $\tilde{\mu}_p$), the fusion weights for small scores $\alpha_t^{(1)}$ decrease when $s_t^{(1)}$ decreases and the fusion weights for large scores $\alpha_t^{(2)}$ increase when $s_t^{(2)}$ increases. As a result, the proposed fusion algorithm ((3) and (5)) favors larger scores only when $\mu > \tilde{\mu}_p$, which has the effect of increasing the mean fused scores. We can also notice from Fig. 2(b) that both the mean of the fused client scores and that of the fused imposter scores increase when the prior score $\tilde{\mu}_p$ is smaller than the respective mean, i.e. $\tilde{\mu}_p < 1.0$ for the client and $\tilde{\mu}_p < -1.0$ for the impostor. Finally, when $\mu = \tilde{\mu}_p$, the proposed fusion approach will be equivalent to equal-weight fusion. This can be observed from Fig. 2(b) where the fused mean scores are equal to $\tilde{\mu}_p$'s, i.e. $\tilde{\mu}_p = 1.0$ for the client and $\tilde{\mu}_p = -1.0$ for the impostor. The curves intersect each other when the prior score $\tilde{\mu}_p$ is equal to the mean of impostor scores. This suggests that the mean of fused scores is equal regardless of the fusion algorithm used.

To conclude, our fusion algorithm will either increase or decrease the mean of fused scores depending on the value of the prior score $\tilde{\mu}_p$ and the score mean $\mu$ before fusion. We can observe from Fig. 2(b) that when the prior score is set between the means of client scores and impostor scores (i.e. between the two vertical lines), theoretically the mean of fused client scores increases and the mean of fused impostor scores decreases. This has the effect of increasing the difference between the means of fused client scores and that of the fused impostor scores, as demonstrated in Fig. 2(c). As the mean of fused scores is used to make the final decision, increasing the score dispersion can decrease the speaker verification error rate.

### 2.5 Comparison between Fusion of Sorted and Unsorted Scores

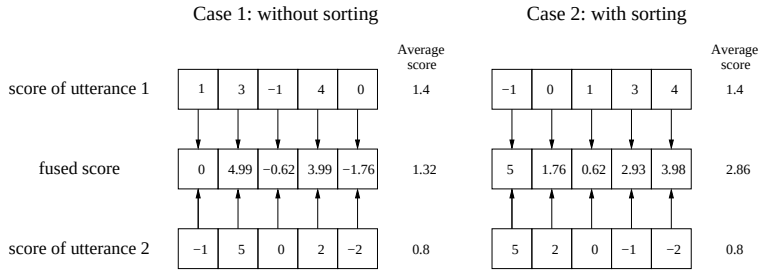
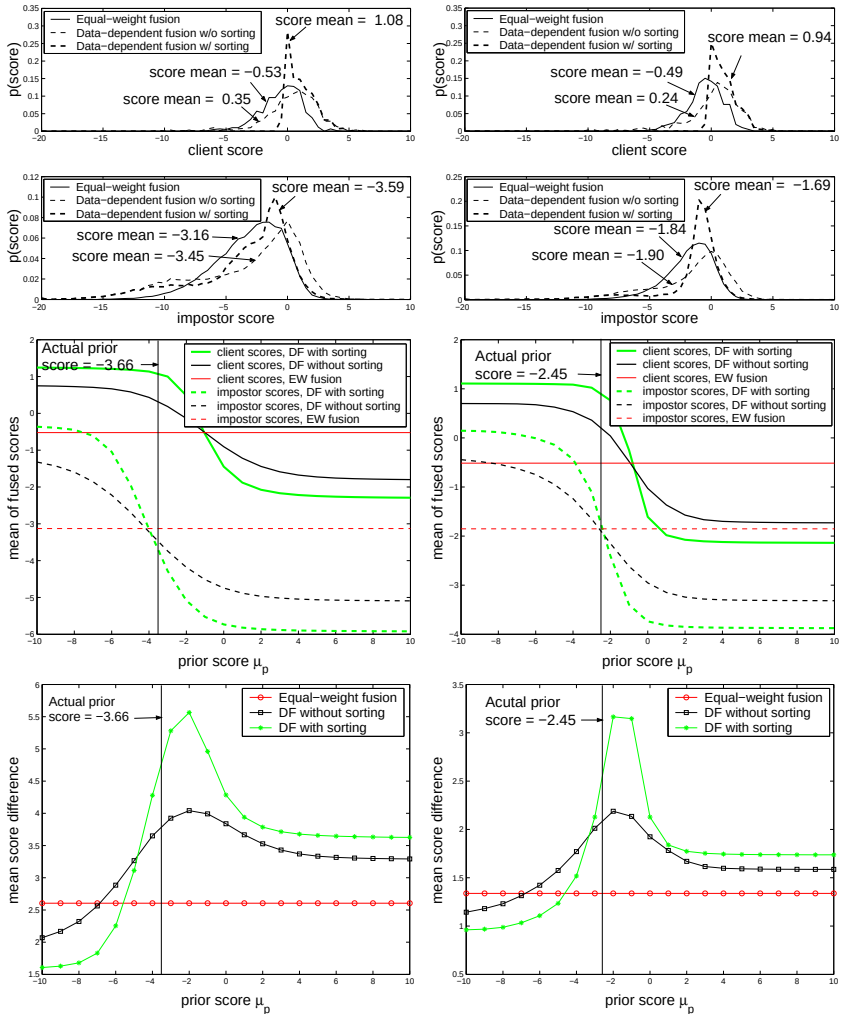



**Fig. 5.** Fused scores derived from unsorted (left figure) and sorted (right figure) score sequences obtained from a client speaker. Here we assume $\tilde{\mu}_p = 0$ and $\tilde{\sigma}_p^2 = 1$ in (5).

In the previous subsection, we have argued that the fusion of sorted score sequences increases the score dispersion. Here, we compare the fusion of unsorted scores with the fusion of sorted scores in terms of verification performance. Fig. 5 shows a hypothetical situation in which the scores were obtained from two client utterances. For client utterances, we would prefer (5) to favor large scores and de-emphasize small scores. However, Case 1 in Fig. 5 clearly shows that the fifth score ($-2$, which is very small) in utterance 2 is emphasized by a relatively larger score in utterance 1. This is because the fifth score of utterance 1 is identical to the prior score ($\tilde{\mu}_p = 0$), which makes the fused score dominated by the fifth score of utterance 2. The influence of these extremely small client scores on the final mean fused score can be reduced by sorting the scores of the two utterances in opposite order before fusion such that small scores will always be fused with large scores. With this arrangement, the contribution of some extremely small client scores in one utterance can be compensated by the large scores of another utterance. As a result, the mean of the fused client scores will be increased. Fig. 5 shows that the mean of fused scores increases from 1.32 to 2.86 after sorting the scores. Likewise, if this sorting approach is applied to the scores of impostor utterances with a proper prior score $\tilde{\mu}_p$ (i.e. greater than the mean of impostor scores, see Fig. 2(b)), the contribution of some extremely large impostor scores in one utterance can be greatly reduced by the small scores in another utterance, which has the net effect of minimizing the mean of the fused impostor scores. Therefore, this score sorting approach can further increase the dispersion between client scores and impostor scores, resulting in a lower error rate. This is demonstrated in Fig. 2(c) where the score dispersion achieved by data-dependent fusion with score sorting is significantly larger than that without score sorting.




(a) From client speaker "mdac0" (b) From client speaker "faem0"

**Fig. 6.** Distributions of pattern-by-pattern client scores (figures in the first row) and impostor scores (figures in the second row), the mean of fused client scores and the mean of fused impostor scores (figures in the third row), and difference between the mean of fused client scores and the mean of fused impostor scores (figures in the fourth row) based on equal-weight fusion (score averaging) and data-dependent fusion with and without score sorting. The means of speaker scores and impostor scores obtained by both fusion approaches are also shown.

To further demonstrate this phenomenon, we select two client speakers (faem0 and mdac0) from the HTIMIT corpus [9] and plot the distributions of the fused speaker scores and fused impostor scores in Fig. 6. In (4), we use the overall mean $\tilde{\mu}_p$ as the prior score. However, as the number of background speakers' utterances is usually much larger than that of client speaker's utterances during the training phase, the overall mean is very close to the mean score of background speakers, i.e. $\tilde{\mu}_p \approx \tilde{\mu}_b$. According to Fig. 2(b) and the third row of Fig. 6, when $\tilde{\mu}_p \approx \tilde{\mu}_b$, the mean of fused impostor scores are almost identical regardless of the fusion algorithm used. However, the same $\tilde{\mu}_p$ will increase the mean of fused client scores significantly, especially when the client scores were sorted before fusion.

Fig. 6(a) shows that the mean of client scores increases from 0.35 to 1.08 and the mean of impostor scores decreases from $-3.45$ to $-3.59$ after sorting the score sequences.[4] Therefore, the dispersion between the mean client score and the mean impostor score increases from 3.80 to 4.67. We can notice from Fig. 6(b) that both the mean of client scores and the mean of impostor scores increase. This is because the means of impostor scores obtained from verification utterances are greater than the prior score $\tilde{\mu}_p$. This results in the increase of the mean of fused impostor scores. However, as the increase in the mean client scores is still greater than the increase in the mean impostor scores, there is still a net increase in the score dispersion. Specifically, the dispersion in Fig. 6(b) increases from $2.14 (= 0.24 - (-1.90))$ to $2.63 (= 0.94 - (-1.69))$. As verification decision is based on the mean scores, the wider the dispersion between the mean client scores and the mean impostor scores, the lower the error rate.

## 3 Speaker Verification Experiments

The proposed fusion algorithm was applied to telephone-based speaker verification. We used a GSM speech codec to transcode the HTIMIT corpus [9] and applied the resulting transcoded speech in a speaker verification experiment similar to [10] and [11]. HTIMIT was obtained by playing a subset of the TIMIT corpus through 9 different telephone handsets and one Sennheizer head-mounted microphone. Speakers in the corpus were divided into a speaker set (50 male and 50 female) and an impostor set (25 male and 25 female).

Sequences of 12th order MFCCs were extracted from 28ms speech frames of uncoded and GSM-transcoded utterances at a frame rate of 71 Hz. During enrollment, we used the SA and SX utterances from handset "senh" of the uncoded HTIMIT to create a 32-center GMM for each speaker. A 64-center universal background GMM [12] was also created based on the speech of 100

[4] The decrease in the mean of fused impostor scores is due to the fact that the prior score $\tilde{\mu}_p$ is greater than the mean of the un-fused impostor scores, see fourth row of Fig. 6(a).

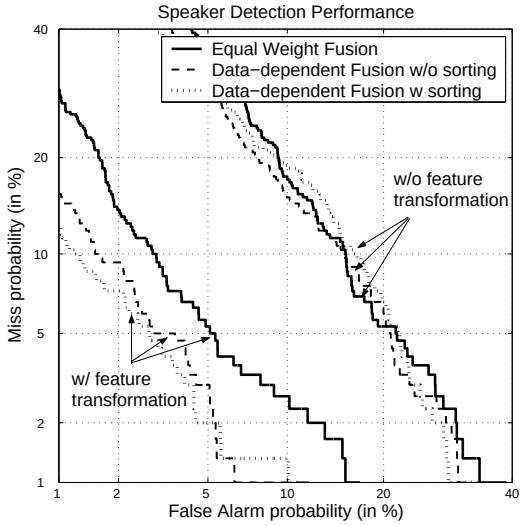



**Fig. 7.** DET curves for equal-weight fusion (score averaging) and data-dependent fusion with and without score sorting. The curves were obtained by using the utterances of handset "cb1" as verification speech.

client speakers recorded from handset "senh". The background model was shared among all client speakers in all verification sessions.

For verification, we used the GSM-transcoded speech from all ten handsets in HTIMIT. As a result, there were handset- and coder- mismatches between speaker models and verification utterances. We used stochastic feature transformation with handset identification [10][13] to compensate the mismatches. We assumed that a claimant will be asked to utter two sentences during a verification session. Therefore, for each client speaker and each impostor, we applied the proposed fusion algorithm to fuse two independent streams of scores obtained from his/her SI sentences. As the fusion algorithm requires the two utterances to have an identical number of feature vectors (length), we computed the average length of the two utterances and appended the extra patterns in the longer utterance to the end of the shorter utterance. Then, we sorted the score sequences in opposite order and fused the sorted scores according to (3) and (5).

Fig. 7 depicts the detection error trade-off curves [14] based on 100 client speakers and 50 impostors using utterances from handset "cb1" for verification. Fig. 7 clearly shows that with feature transformation, data-dependent fusion is able to reduce the error rate significantly, and sorting the scores before fusion can reduce the error rate further. However, without feature transformation, the performance of data-dependent fusion with score sorting is not significantly better than that of equal-weight fusion. This is caused by the mismatch between the prior scores $\tilde{\mu}_p$'s in (5) and the scores of the distorted

features. Therefore, it is very important to use feature transformation to reduce the mismatch between the enrollment data and verification data.

Fig. 8 shows the detection error trade-off curves based on 100 client speakers and 50 impostors using all the scores from ten handsets. It shows that data-dependent fusion with score sorting outperforms equal-weight fusion for all operating points and by 23% in terms of equal error rate.

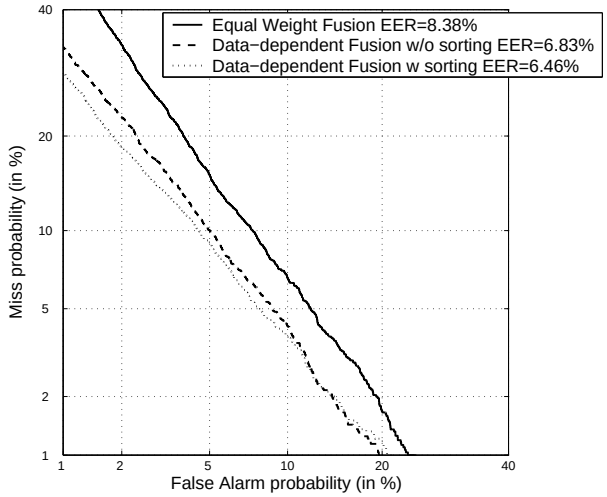



**Fig. 8.** DET curves for equal-weight fusion (score averaging) and data-dependent fusion with and without score sorting. The curves were obtained by concatenating the scores from ten handsets.

Table 1 shows the speaker detection performance of 100 speakers and 50 impostors for the equal-weight fusion approach and the proposed fusion approach with and without sorting the score sequences. Table 1 clearly shows that our proposed fusion approach outperforms the equal-weight fusion. In particular, after the score sequences have been sorted, the equal error rate is further reduced.

## 4 Conclusions

We have presented a decision fusion algorithm that makes use of prior score statistics and the distribution of the recognition data. The fusion algorithm was combined with feature transformation for speaker verification using GSM-transcoded speech. Results show that the proposed fusion algorithm outperforms equal-weight fusion. It was also found that performance can be further improved by the fusion of sorted scores.

**Table 1.** Equal error rates achieved by different fusion approaches, using utterances from 10 different handsets for verification. Each figure is based on the average of 100 speakers, each impersonated by 50 impostors. DF stands for data-dependent fusion. "No fusion" means the verification results were obtained from using single utterance per verification session. "average" is the average EER of 10 handsets.

| | **Equal Error Rate (%)** | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Fusion Method | cb1 | cb2 | cb3 | cb4 | el1 | el2 | el3 | el4 | pt1 | senh | **average** |
| No fusion | 6.31 | 6.36 | 19.96 | 15.35 | 5.89 | 10.83 | 11.79 | 8.18 | 9.38 | 4.90 | **9.90** |
| Equal-weight fusion | 5.11 | 4.33 | 19.15 | 12.89 | 4.42 | 8.31 | 9.96 | 6.29 | 7.57 | 2.99 | **8.10** |
| DF w/o sorting | 4.01 | 3.27 | 15.92 | 10.55 | 3.04 | 6.51 | 8.67 | 4.75 | 7.51 | 2.32 | **6.67** |
| DF w/ sorting | 3.60 | 2.86 | 15.30 | 9.91 | 3.49 | 4.65 | 6.81 | 4.02 | 6.59 | 1.99 | **5.92** |

## 5 Acknowledgement

This work was supported by the Hong Kong Polytechnic University Grant No. G-T860 and HKSAR RGC Project No. PolyU 5131/02E.

## References

1. Wark, T., Sridharan, S. (2001) Adaptive fusion of speech and lip information for robust speaker identification. Digital Signal Processing, vol. 11, pp. 169–186
2. Jourlin, P., Luettin, J., Genoud, D., Wassner, H. (1997) Acoustic-labial speaker verification. Pattern recognition letters, vol. 18, no. 9, pp. 853–858
3. Kittler, J., Hatef, M., Duin, R.P.W., Matas, J. (1998) On combining classifiers. IEEE Trans. on Pattern Anal. Machine Intell., vol. 20, no. 3, pp. 226–239
4. Sanderson, C., Paliwal, K. K. (2001) Joint cohort normalization in a multi-feature speaker verification system. The 10th IEEE International Conference on Fuzzy Systems 2001, vol. 1, pp. 232–235
5. Poh, N., Bengio, S., Korczak, J. (2002) A multi-sample multi-source model for biometric authentication. Proc. IEEE 12th Workshop on Neural Networks for Signal Processing, pp. 375–384
6. Mak, M.W., Cheung, M.C., Kung, S.Y. (2003) Robust speaker verification from GSM-transcoded speech based on decision fusion and feature transformation. Proc. IEEE ICASSP'03, pp. II745–II748
7. Cheung, M.C., Mak, M.W., Kung, S.Y. (2003) Adaptive decision fusion for multi-sample speaker verification over GSM networks. Eurospeech'03, pp. 1681–1684
8. Cheung, M.C., Mak, M.W., Kung, S.Y. (2004) Multi-sample data-dependent fusion of sorted score sequences for biometric verification. Proc. IEEE ICASSP04, pp. V681–V684
9. Reynolds, D.A. (1997) HTIMIT and LLHDB: speech corpora for the study of handset transducer effects. Proc. IEEE ICASSP'97, pp. II1535–II1538
10. Mak, M.W., Kung, S.Y. (2002) Combining stochastic feature transformation and handset identification for telephone-based speaker verification. Proc. IEEE ICASSP'2002, pp. I701–I704

11. Yu, W.M., Mak, M.W., Kung, S.Y. (2002) Speaker verification from coded telephone speech using stochastic feature transformation and handset identification. The 3rd IEEE Pacific-Rim Conference on Multimedia 2002, pp. 598–606.
12. Reynolds, D.A., Quatieri, T.F., Dunn, R.B. (2000) Speaker verification using adapted gaussian mixture models. Digital Signal Processing, vol. 10, pp. 19–41
13. Tsang, C.L., Mak, M.W., Kung, S.Y. (2002) Divergence-based out-of-class rejection for telephone handset identification. Proc. ICSLP'02, pp. 2329–2332
14. Martin, A., Doddington, G., Kamm, T., Ordowski, M., Przybocki, M. (1997) The DET curve in assessment of detection task performance. Proc. Eurospeech '97, pp. 1895–1898