論文 / 著書情報
Article / Book Information

Title	Noise-robust speaker verification using F0 features
Authors	Koji Iwano, Taichi Asami, Sadaoki Furui
Citation	Interspeech2004-ICSLP, Vol. 2, , pp. 1417-1420,
Pub. date	2004, 10
Copyright	(c) 2004 International Speech Communication Association, ISCA
DOI	http://dx.doi.org/

Noise-Robust Speaker Verification Using F_0 Features

Koji Iwano, Taichi Asami, and Sadaoki Furui

Department of Computer Science, Tokyo Institute of Technology
2-12-1 Ookayama, Meguro-ku, Tokyo, 152-8552 Japan
{iwano,taichi,furui}@furui.cs.titech.ac.jp

Abstract

This paper proposes a noise-robust speaker verification method augmented by fundamental frequency (F_0). The paper first describes a noise-robust F_0 extraction method using the Hough transform. Then, it proposes a robust speaker verification method using multi-stream HMMs which fuse the extracted F_0 and cepstral features. Experiments are conducted using four-connected-digit utterances of Japanese by 37 male speakers recorded at five sessions over a half year period. The utterances are contaminated with white noise at various SNR levels. Experimental results show that the F_0 features improve the verification performance in all SNR conditions.

1. Introduction

In order to exploit high-performance speaker recognition systems, various methods using fundamental frequency (F_0) in combination with spectral/cepstral features have been proposed[1-9]. Since F_0 features are less sensitive to channel distortions or additive noise than spectral/cepstral features, they are expected to be useful for increasing the robustness of speaker recognition. [2] proposed a robust speaker recognition method using F_0 features to cope with the effect of handset variation on telephone speech. [4] showed that F_0 features increased robustness of VQ-based speaker identification against additive noise. However, F_0 features have not been fully exploited for increasing noise-robustness in speaker verification.

In this paper, we propose a noise-robust HMM-based speaker verification method using F_0 features. In our previous work on noise-robust speech recognition[10], multi-stream HMMs were used for fusing F_0 and cepstral features, and the Hough transform[11], one robust image processing technique, was used for reliably extracting F_0 values. Since this method was effective for improving the speech recognition performance in various kinds of noise and SNR conditions, we have applied the same strategy to speaker verification as reported in this paper.

This paper is organized as follows: Section 2 explains a robust F_0 extraction method using the Hough transform. In Section 3, our noise robust speaker verification method using multi-stream HMMs is explained. Experimental results are reported in Section 4, and Section 5 concludes this paper.

2. Noise-Robust F_0 Extraction Using the Hough Transform

2.1. Hough transform

The Hough transform is a technique to robustly extract parametric patterns, such as lines, circles, and ellipses, from a noisy image[11].

This paper uses the Hough transform method to extract linear transitional patterns of F_0 values. The method for extracting a significant line from an image on the x–y plane can be formulated as follows. Suppose the image consists of n pixels at (x_i, y_i) $(i = 1, \cdots, n)$. Every pixel on the x–y plane is transformed to a line on the m–c plane as

$$c = -x_i m + y_i \quad (i = 1, \cdots, n) \tag{1}$$

Brightness values of pixels on the x–y plane are accumulated at every point on the line. This process is called "voting" to the m–c plane. After voting of all pixels has been completed, the maximum accumulated voting value on the m–c plane is detected, and the peak point (m, c) is transformed to a line on the x–y plane by the following equation:

$$y = mx + c \tag{2}$$

2.2. F_0 extraction using the Hough transform

Although F_0 contours have temporal continuity in the voiced period, cepstral peaks which have been widely used to extract F_0 values often cause errors, including half pitch, double pitch and drop outs, due to noise effects. To take advantage of the continuity, the Hough transform is applied to time-cepstrum images of noisy speech.

Speech waveforms are sampled at 16kHz and transformed to 256 dimensional cepstra. A 32ms-long Hamming window is used to extract frames every 10ms. To the time-cepstrum image, a nine-frame moving window is applied at every frame interval to extract an image for line information detection. The time-cepstrum image is used as the pixel brightness image for the Hough transform. An F_0 value is obtained from a cepstrum index of the center point of the detected line. Since the moving window has nine frames, the time continuity for 90ms is taken into account in this method.

3. Noise-Robust Speaker Verification Using Multi-Stream HMMs

3.1. Japanese connected digit speech

The proposed method was evaluated using four-connected-digit speech in Japanese. In Japanese connected digit speech, two consecutive digits usually make one prosodic phrase. Figure 1 shows an example of an F_0 contour of four-connected-digit speech. The first two digits make the first prosodic phrase, and the latter two digits make the second prosodic phrase. The transition of F_0 is represented by CV syllabic units, and each CV syllable can be prosodically labeled as a "rising" or "falling" F_0 part.

3.2. Integration of segmental and prosodic features

Each segmental feature vector has 25 elements consisting of 12 MFCC, their deltas, and the delta log energy. The window

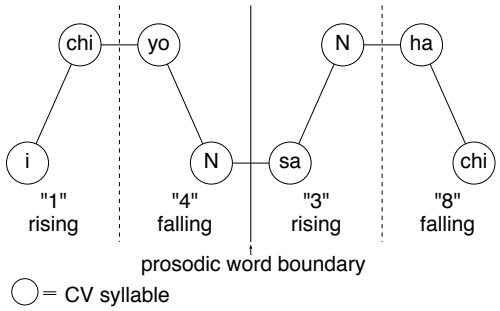

Figure 1: An example of F_0 contour of four connected digit speech in Japanese.

length is 25ms and the frame interval is 10ms. Cepstral mean subtraction (CMS) is applied to each utterance.

Two kinds of prosodic features are extracted; $\log F_0$ and $\Delta \log F_0$. Prosodic feature vectors consist of both or either of the two features. The segmental and prosodic feature vectors are combined for each frame to build a segmental-prosodic feature vector.

3.3. Multi-stream syllable HMMs

3.3.1. Basic structure of syllable HMMs

Since timing of the change of F_0 transitions, such as "rising" and "falling", is highly related to that of CV syllable transitions, segmental and prosodic features are integrated in our method using syllabic unit HMMs.

The integrated syllable HMM denoted by "SP-HMM (Segmental-Prosodic HMM)" is modeled by taking both the context and the F_0 transitions into account. Each Japanese digit uttered continuously with other digits can be modeled by a concatenation of two syllables (morae). Even "2" (/ni/) and "5" (/go/) can be modeled by two syllables, since their final vowel is usually lengthened as /ni:/ and /go:/. The context of each syllable is considered only within each digit in our experiment. Therefore, the SP-HMM can be denoted by either a left-context dependent syllable "`LC-SYL,PM`" or a right-context dependent syllable "`SYL+RC,PM`", where "`PM`" indicates an F_0 transition pattern which is either rising (`U`) or falling(`D`). For example, the first syllable /i/ of "1" (/ichi/) which has rising F_0 transition is denoted as "`i+chi,U`". Each SP-HMM has a standard left-to-right topology with $n \times 3$ states, where n is the number of phonemes in the syllable.

3.3.2. Multi-stream modeling

SP-HMMs are modeled as multi-stream HMMs. In recognition, the probability $b_j(\boldsymbol{O}_{SP})$ of generating segmental-prosodic observation $\boldsymbol{O}_{SP}$ at state j is calculated by:

$$b_j(\boldsymbol{O}_{SP}) = b_j(\boldsymbol{O}_S)^{\lambda_S} \cdot b_j(\boldsymbol{O}_P)^{\lambda_P} \tag{3}$$

where $b_j(\boldsymbol{O}_S)$ is the probability of generating segmental feature vectors $\boldsymbol{O}_S$, and $b_j(\boldsymbol{O}_P)$ is the probability of generating prosodic feature vectors $\boldsymbol{O}_P$. λ_S and λ_P are weighting factors for the segmental stream and the prosodic stream, respectively. They are constrained by $\lambda_S + \lambda_P = 1 \ \ (0 \leq \lambda_S, \lambda_P \leq 1)$.

3.3.3. Building SP-HMMs

Syllable HMMs for segmental and prosodic feature vectors are separately made and combined to build SP-HMMs using a tied-mixture technique as follows:

1. "S-HMMs (Segmental HMMs)" are trained by segmental features only. They can be denoted by either "`LC-SYL,*`" or "`SYL+RC,*`". Here, "`*`" (wild card) means that HMMs are built without considering the F_0 transitions, "`U`" and "`D`". The total number of S-HMM states is the same as the number of SP-HMM states.
2. Training utterances are segmented into syllables by the forced-alignment technique using the S-HMMs, and one of the F_0 transition labels, "`U`" or "`D`", is given to each segment according to the actual F_0 pattern.
3. "P-HMMs (Prosodic HMMs)" are trained by prosodic feature vectors within these segments, according to the F_0 transition label. Five separate models, "`*-*,U`", "`*+*,U`", "`*-*,D`", "`*+*,D`", and "`sil`", are made. Each P-HMM has a single state.
4. The S-HMMs and P-HMMs are combined to make SP-HMMs. Gaussian mixtures in the segmental stream of SP-HMMs are tied with corresponding S-HMM mixtures, while the mixtures in the prosodic stream are tied with corresponding P-HMM mixtures. Figure 2 shows the integration process. In this example, the mixtures of SP-HMM "`i+chi,U`" are tied with S-HMM "`i+chi,*`" and P-HMM "`*+*,U`".

3.4. Verification score

A posterior probability is used as the score for verification decisions. The posterior probability of being the claimed speaker S^c after observing a feature set x is denoted by $p(S^c|x)$.

$$p(S^c|x) = \frac{p(x|S^c)p(S^c)}{p(x)} \tag{4}$$

where $p(x|S^c)$ is a likelihood value with claimed speaker's SP-HMM. The probability $p(x)$ is approximated by using a likelihood value with general speaker's SP-HMM $p(x|S^g)$:

$$\begin{aligned} p(S^c|x) &= \frac{p(x|S^c)p(S^c)}{p(s|S^g)p(S^g)} \\ &\propto \frac{p(x|S^c)}{p(x|S^g)} \end{aligned} \tag{5}$$

The right term of equation (5) is calculated as follows:

$$\begin{aligned} \frac{p(x|S^c)}{p(x|S^g)} &= \frac{\Sigma_w p(x|S^c, w)p(w)}{\Sigma_w p(x|S^g, w)p(w)} \\ &\approx \frac{\max_w p(x|S^c, w)}{\max_w p(x|S^g, w)} \end{aligned} \tag{6}$$

where w is a string of four connected digits. Equation (6) means that our method uses two likelihoods calculated by usual speech recognition processes using the speaker dependent (SD) and independent (SI) SP-HMMs. The verification score is denoted by $\log p(S^c|x)$. If the score is larger than a threshold value, the speaker is accepted as the claimed speaker.

3.5. Dictionary and Grammar

Speaker verification is performed in the text-independent framework. In the dictionary, each digit has three variations according to the F_0 transitions. For instance, variations of "1" are "`i+chi,U i-chi,U sp`" and "`i+chi,D i-chi,D sp`". This means that the F_0 transition pattern does not change within the period of each digit. In the grammar, all digits are allowed to be connected with no restrictions.

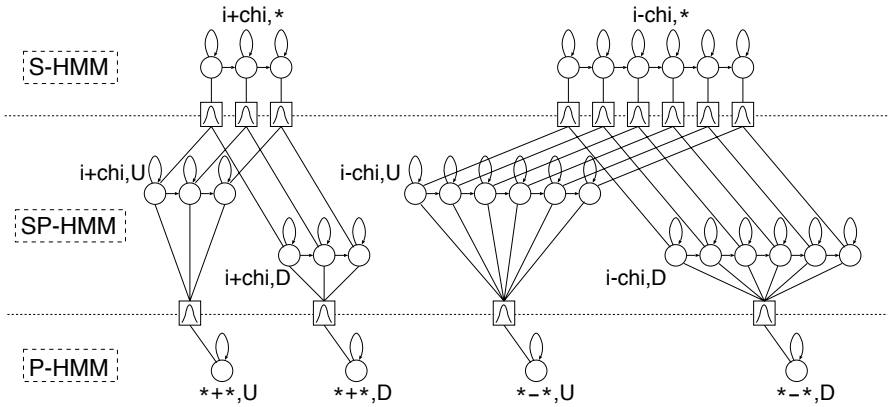

Figure 2: Building SP-HMMs using a tied-mixture technique. S-HMMs and P-HMMs are trained using segmental features and prosodic features, respectively.

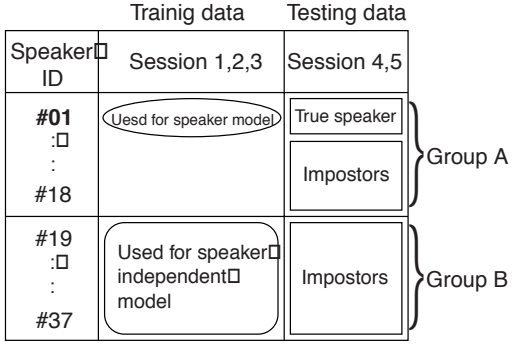

Figure 3: Training and testing data for the verification experiment when the speaker #01 is the claimed speaker.

4. Experiments

4.1. Database

Speech data were recorded at five sessions with intervals of approximately one month. The data were collected from 37 male speakers and sampled at 16kHz with a 16bit resolution. Each speaker uttered 50 strings of four connected digits in Japanese at each session.

The set of data recorded at sessions $1 \sim 3$ was used for training and data recorded at sessions 4 and 5 was used for testing. The database was separated into two groups in terms of speakers as shown in Figure 3. This figure shows the case where speaker #01 was used as the claimed speaker. The SI model was trained using utterances by all the speakers in speaker group B, which did not include the claimed speaker. When one of the speakers in speaker group B was used as the claimed speaker, utterances by speaker group A were used for the SI model training. In this way, the SI model was always trained using the data of a speaker group not including the claimed speaker. All the speakers in both speaker groups A and B, except for the claimed speaker himself, were used as impostors.

White noise was added to the training data at a 30dB SNR level to increase the robustness against noisy speech, and testing data were contaminated with white noise at 5, 10, 15, 20, and 30dB SNR conditions.

Table 1: Three kinds of prosodic feature vectors extracted by Hough transform.

Prosodic feature vector	Vector component (dim.)
H-L	$\log F_0$ (1)
H-D	$\Delta \log F_0$ (1)
H-LD	$\log F_0, \Delta \log F_0$ (2)

4.2. Experimental results

In our preliminary experiments using S-HMMs in 30dB noise condition, the best verification performance was obtained when the number of mixtures in S-HMMs was four. Accordingly, we used this mixture condition for S-HMMs in the following experiments.

4.2.1. Effectiveness of prosodic feature vectors

We first investigated speaker verification performance in various conditions of prosodic feature vectors. Table 1 explains three kinds of prosodic feature vectors, **H-L**, **H-D**, and **H-LD**, built using the $\log F_0$ and $\Delta \log F_0$ extracted by the Hough transform. The number of mixtures in prosodic stream (P-HMMs) in SP-HMMs tied to the four mixture S-HMMs is optimized for each prosodic feature vector at 30dB SNR condition.

Figure 4 shows equal error rates (EER) using each prosodic feature vector at various SNR conditions. It was found that the best number of mixtures in P-HMMs was four, irrespective of the kind of prosodic feature vector used. All prosodic feature vectors were effective in improving verification performance in noisy environments. **H-L** yielded better performance than **H-D**. Since the best improvement was obtained when using **H-LD**, we used this feature vector in subsequent experiments. The best improvement by using **H-LD** was observed at SNR = 10dB; the error rate was reduced by 39.8% from the baseline (S-HMMs) method.

4.2.2. Effectiveness of the Hough transform

For examining the effect of the Hough transform on verification performance, a two-dimensional prosodic feature vector **NH-LD** was prepared without using the Hough transform; it consisted of $\log F_0$, extracted by choosing highest cepstral peaks, and $\Delta \log F_0$, computed by linear smoothing of the $\log F_0$ values within a 90ms window.

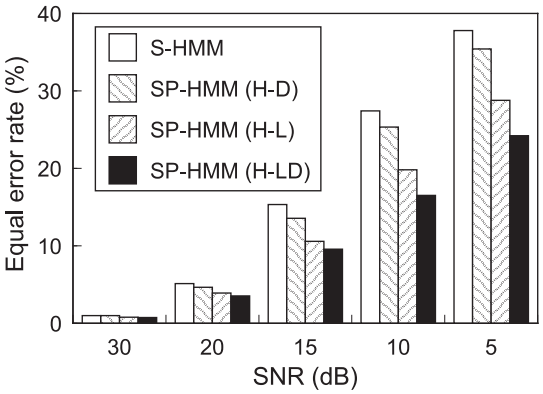

Figure 4: Speaker verification results in various conditions of prosodic feature vectors.

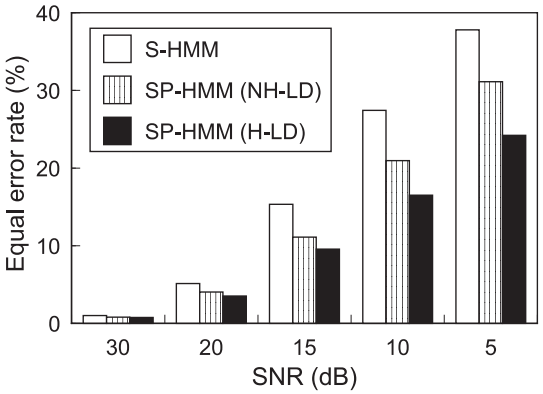

Figure 5: Comparison of the EERs when using the prosodic feature vector with/without the Hough transform.

The comparison of the EERs when using the feature vector **H-LD** and **NH-LD** is shown in Figure 5. **H-LD** yielded better performance than **NH-LD**, indicating that the Hough transform is effective in F_0 extraction in noise-robust speaker verification.

4.2.3. Effects of the prosodic stream weight

Figure 6 shows the EERs as a function of the prosodic stream weight λ_P at each SNR. Improvements from baseline ($\lambda_P = 0$) are observed over a wide range: $0.0 < \lambda_P < 0.9$ in all SNR conditions. This means that the proposed method is not sensitive to the change of the stream weight.

5. Conclusions

This paper has proposed a speaker verification method using multi-stream HMMs which combine segmental and prosodic features. The prosodic features are extracted by an F_0 feature extraction technique based on the Hough transform. Experimental results using Japanese connected digit speech show that: 1) the Hough transform is effective for increasing robustness in extracting F_0 features in the proposed verification method; 2) the best verification performance is obtained when using both $\log F_0$ and $\Delta \log F_0$ as prosodic features; and 3) our method is not sensitive to the change of the stream weight.

Our future works include: 1) investigating useful prosodic features other than F_0-based features; 2) improving the SP (segmental-prosodic)-HMM topology; 3) effectively using voiced/unvoiced information; and 4) developing an automatic method for optimizing the stream weight.

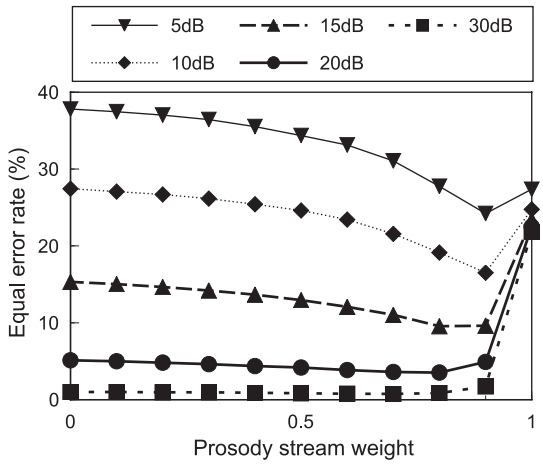

Figure 6: EERs as a function of the prosodic stream weight (λ_P) in each SNR condition

6. References

[1] T. Matsui and S. Furui, "Text-independent speaker recognition using vocal tract and pitch information," *Proc. ICSLP90*, vol.1, pp.137-140, Kobe (1990-11).

[2] M.J. Carey, et al., "Robust prosodic features for speaker identification," *Proc. ICSLP96*, vol.3, pp.1800-1803, Philadelphia, Pennsylvania (1996-10).

[3] M.K. Sönmez, et al., "A lognormal tied mixture model of pitch for prosody-based speaker recognition," *Proc. Eurospeech97*, vol.3, pp.1391-1394, Rhodes (1997-9).

[4] Y.-J. Kyung and H.-S. Lee, "Text independent speaker recognition using micro-prosody," *Proc. ICSLP98*, vol.1, pp.157-160, Sydney (1998-12).

[5] Y. Cheng and H.-C. Leung, "Speaker verification using fundamental frequency," *Proc. ICSLP98*, vol.1, pp.161-164, Sydney (1998-12).

[6] K.P. Markov and S. Nakagawa, "Text-independent speaker recognition using multiple information sources," *Proc. ICSLP98*, vol.1, pp.173-176, Sydney (1998-12).

[7] C. Miyajima, et al., "Text-independent speaker identification using Gaussian mixture models based on multi-space probability distribution," *IEICE Trans. Inf. & Syst.*, vol.E84-D, no.7, pp.847-855 (2001-7).

[8] F. Weber, et al., "Using prosodic and lexical information for speaker identification," *Proc. ICASSP2002*, vol.1, pp.141-144, Orlando, Florida (2002-5).

[9] D. Reynolds, et al., "The SuperSID project: Exploiting high-level information for high-accuracy speaker recognition," *Proc. ICASSP2003*, vol.4, pp.784-787, Hong Kong (2003-4).

[10] K. Iwano, et al., "Noise robust speech recognition using F_0 contour extracted by Hough transform," *Proc. ICSLP2002*, vol.2, pp.941-944, Denver, Colorado (2002-9).

[11] P.V.C. Hough, "Method and means for recognizing complex patterns," U.S. Patent #3069654 (1962).