STUDIA PHONOLOGICA XXIV (1990)

Effect of Time Duration and Intrinsic Features for English Phoneme Recognition

Yasuo ARIKI

SUMMARY

This paper describes methods to improve the performance of English phoneme recognition from linguistic view points. The methods include exploiting time duration information in hidden Markov model (HMM), intrinsic feature space for vowel. The time duration constraint imposed on states of the phoneme HMM can improve its recognition rate significantly for phoneme data in continuous pseech. As intrinsic feature spaces for vowel, formants and the time derivative are employed. They improve the phoneme recognition rate considerably compared with the commonly used LPC cepstral coefficients.

I. INTRODUCTION

The speech signal exhibits variability in both time duration and spectrum even by the same speaker. This variability has been causing difficulty in the realisation of speech recognition by machine. Hidden Markov modelling (HMM) was proposed to absorb the variability, by virture of its computational time alignment capability coupled with use of spectral probability distribution functions (pdf) [1]. The HMM made it feasible to recognise continuous speech with large vocabulary by modelling sub-word speech units. The subword models can be easily concatenated into word models owing to the probabilistic representation of the HMM [2][3] [4].

Using HMM sub-word modelling as bases for experimentation, we investigated several methods of increasing its performance. In this paper following two methods to improve phoneme recognition rate are described:

(1) Incorporation of time duration modelling into the phoneme HMM.

(2) Employment of specific feature space intrinsic to vowels and consonants.

The HMM can be thought as a time varying information source which can be modelled as Markov process with the pdf at each state. Speech data are used to estimate the pdf at each state and state transition probability of the Markov model. However, it lacks information about how long the speech data can be likely to stay at each state [5][6]. In this paper, we formalise the time duration modelling in the HMM and show the recognition improvement of the phoneme in continuous speech,

Yasuo ARIKI (有木康雄): Associated Professor, Department of Electronics and informatics, Ryukoku University

compared to the conventional HMM without time duration [7].

In phoneme recognition by HMM, a single acoustic feature space, such as LPC cepstral coefficients, has been employed so far, irrespective of the phoneme classes: consonants or vowels. However, their features may be well represented in intrinsic feature space, since vowels and consonants are produced through different vocal mechanisms. For example, consonants can be characterised largely by spectral envelope and their temporal movement. On the other hand, vowels can be characterised by formant frequencies rather than spectral envelope. In this paper· we demonstrate considerable improvement of the vowel recognition by employing formant frequency instead of LPC cepstral coefficients [8].

In order to formalise the time duration modelling in the HMM, we describe EM algorithm for parameter estimation of the HMM in section II, then its application to the discrete HMM is described in section III and IV. Section V formalises the time duration HMM. Phoneme recognition using the discrete HMM and the effect of time duration HMM are described in section VI and VII respectively. Section VIII describes the experimental results of phoneme recognition in intrinsic feature space.

II. EM Algorithm

We suppose here that observed speech sample x comes out according to probabilistic structure y with the probability density of $f(x,y)$. Information such as from which probabilistic structure (class, state) a data sample x comes is unobservable and only the speech sample x is observed. Observable data are called incomplete data because they are missing the unobservable data y, and data composed of observable and unobservable data are called complete data. The purpose of the EM algorithm is to maximise the log-likelihood of incomplete data, by iteratively maximising the expectation of log-likelihood of complete data. The name of the EM algorithm comes from E for *expectation* and M for *maximisation*. It can be said that the EM algorithm is a maximum likelihood estimation method, but its computation is less complex than the conventional maximum likelihood estimation method.

Let $L(x, \bar{\lambda})$ denote the log-likelihood of the observed speech sample x. Here $\bar{\lambda}$ is the parameters of the probabilistic structure y. The log-likelihood $L(x, \bar{\lambda})$ is decomposed into two functions; $Q(\lambda, \bar{\lambda})$ and $H(\lambda, \bar{\lambda})$ as follows.

$$L(x, \bar{\lambda}) = Q(\lambda, \bar{\lambda}) - H(\lambda, \bar{\lambda}) \tag{1}$$

where λ and $\bar{\lambda}$ indicate the parameters of the probabilistic sturcture y already estimated and to be estimated at this iteration step respectively. The H function has the property that $H(\lambda, \bar{\lambda}) \leq H(\lambda, \lambda)$, then the log-likelihood L increases if the function Q increases for the newly estimated parameters $\bar{\lambda}$. The function Q is

expressed as follows:

$$Q(\lambda, \bar{\lambda}) = \frac{1}{f(x|\lambda)} \sum_{y} f(x, y|\lambda) \log f(x, y|\bar{\lambda}) \tag{2}$$

The procedure of the EM algorithm can be summarised in the following way.

1. Choose an initial estimate λ.
2. E-step. Compute $Q(\lambda, \bar{\lambda})$ based on the given λ.
3. M-step. Choose $\bar{\lambda} \in \underset{\bar{\lambda}}{argmax}\ Q(\lambda, \bar{\lambda})$. Here, $\underset{\bar{\lambda}}{argmax}\ Q(\lambda, \bar{\lambda})$ denotes the set of values $\bar{\lambda}$ which maximise $Q(\lambda, \bar{\lambda})$.
4. Set $\lambda = \bar{\lambda}$, repeat from step 2 until convergence.

The Q function expressed in $EQ.(2)$ is concerned with a single observed incomplete data x_k. Applying it to multiple observed incomplete data $X = \{x_1, \cdots, x_N\}$, the Q function is extended to:

$$\begin{aligned} Q(\lambda, \bar{\lambda}) &= \sum_{k=1}^{N} Q_k(\lambda, \bar{\lambda}) \\ &= \sum_{k=1}^{N} \sum_{y} \frac{f(x_k, y|\lambda)}{f(x_k|\lambda)} \log f(x_k, y|\bar{\lambda}) \end{aligned} \tag{3}$$

where N denotes the number of observed data.

The EM algorithm is used in applications which permit easy maximisation of the Q-function instead of maximising $L(x, \bar{\lambda})$ directly. In such applications, the M-step maximisation of $Q(\lambda, \bar{\lambda})$ is easily carried out.

III. Application of EM Algorithm to HMM

H dden Markov modelling is the probabilistic modelling to deal with a time sequence of speech data. Stable state of the speech data is modelled by one simple probabilistic model and the time sequence is modelled by the transition from one probabilistic model to anther model.[7] Let $X = x_1 \cdots x_t \cdots x_T$ denote a time sequnece of speech data and $S = s_1 \cdots s_t \cdots s_T$ denote a time sequence of stable states of the model. The x_t is referred to as frame data hereafter. The complete data is expressed as (X, S). In order to apply the EM algorithm to the HMM, it is necessary to formalise the probability density function (pdf) of the complete data. Then the Q function expressed in $EQ.(2)$ can be applied. The pdf of the complete data is expressed as:

$$\begin{aligned} f(X, S|\lambda) &= f(x_1 \cdots x_t \cdots x_T, s_1 \cdots s_t \cdots s_T|\lambda) \\ &= f(x_1, s_1|\lambda) \cdots f(x_t, s_t|X_1^{t-1}, S_1^{t-1}, \lambda) \\ &\qquad \cdots f(x_T, s_T|X_1^{T-1}, S_1^{T-1}, \lambda) \end{aligned} \tag{4}$$

where X_1^{t-1} and S_1^{t-1} denote the partial sequence from time 1 to $t-1$ of the speech data X and state sequence S. Here, we call $f(x_t, s_t|X_1^{t-1}, S_1^{t-1}, \lambda)$ conditional pdf of the complete data at time t. We introduce following hypothesis into the model.

The conditional pdf of the complete data at time t does not depend on the previous frame data sequence X_1^{t-1}, but only depends on the previous stable state s_{t-1} Under this hypothesis, the conditional pdf of the complete data at time t is simplified as follows:

$$\begin{aligned} f(x_t, s_t | X_1^{t-1}, S_1^{t-1}, \lambda) &= f(x_t, s_t | s_{t-1}, \lambda) \\ &= Pr(s_t | s_{t-1}, \lambda) f(x_t | s_{t-1}, s_t, \lambda) \end{aligned} \tag{5}$$

We introduce the following terms to make the model more clear.

$$\begin{array}{lll} \textit{Transition probability} & a_{s_{t-1}s_t} = Pr(s_t | s_{t-1}, \lambda) & (2 \leq t \leq T) \\ \textit{initial probability} & \pi_{s_1} = Pr(s_1 | \lambda) & (t = 1) \\ \textit{output probability} & b_{s_{t-1}s_t}(x_t) = f(x_t | s_{t-1}, s_t, \lambda) & (1 \leq t \leq T) \end{array} \tag{6}$$

The output probability $b_{s_{t-1}s_t}(x_t)$ is the probability to output the data x_t at the transition from the state s_{t-1} to s_t. If all the output probabilities from any state to the state s_t are tied, the output probability is expressed as:

$$b_{s_t}(x_t) = f(x_t | s_t, \lambda) \quad (1 \leq t \leq T) \tag{7}$$

Then the conditional pdf of the complete data at time t is expressed as:

$$f(x_t, s_t | X_1^{t-1}, S_1^{t-1}, \lambda) = \begin{cases} a_{s_{t-1}s_t} b_{s_t}(x_t) & (2 \leq t \leq T) \\ \pi_{s_1} b_{s_1}(x_1) & (t = 1) \end{cases} \tag{8}$$

From the EQ.(4) and EQ.(8), the pdf of the complete data $f(X, S | \lambda)$ is obtained as:

$$\begin{aligned} f(X, S | \lambda) &= \pi_{s_1} b_{s_1}(x_1) \prod_{t=2}^{T} a_{s_{t-1}s_t} b_{s_t}(x_t) \\ &= \pi_{s_1} \prod_{t=1}^{T-1} a_{s_t s_{t+1}} \prod_{t=1}^{T} b_{s_t}(x_t) \end{aligned} \tag{9}$$

By substituting the EQ.(9) to the EQ.(2), the HMM parameters are iteratively obtained by the EM algorithm under the hypothesis of data independence and one state dependency (Markov chain).

IV. Discrete HMM

In discrete HMM, frame data x_t in speech data X is quantised into finite set of repersentatives, called codewords. The process of quantisation is called vector quantisation. In the application of the EM algorithm to the quantised speech data $O = o_1 \cdots o_t \cdots o_T$, the pdf of the complete data (O, S) is substituted into the EQ.(2). Using the notation of the initial probability π_{s_1}, state transition probability $a_{s_t s_{t+1}}$ and output probability $b_{s_t}(o_t)$, the log-likelihood (log pdf) of the complete data is expressed as follows, applying log operation to the EQ.(9):

$$\log Pr(O, S|\bar{\lambda}) = \log \bar{\pi}_{s_1} + \sum_{t=1}^{T-1} \log \bar{a}_{s_t s_{t+1}} + \sum_{t=1}^{T} \log \bar{b}_{s_t}(o_t) \tag{10}$$

Then $Q(\lambda, \bar{\lambda})$ function is expressed as:

$$\begin{aligned} Q(\lambda, \bar{\lambda}) &= \frac{1}{Pr(O|\lambda)} \sum_{S} Pr(O, S|\lambda) \log \bar{\pi}_{s_1} \\ &+ \frac{1}{Pr(O|\lambda)} \sum_{S} Pr(O, S|\lambda) \sum_{t=1}^{T-1} \log \bar{a}_{s_t s_{t+1}} \\ &+ \frac{1}{Pr(O|\lambda)} \sum_{S} Pr(O, S|\lambda) \sum_{t=1}^{T} \log \bar{b}_{s_t}(o_t) \end{aligned} \tag{11}$$

By maximising the first term of the right hand side in terms of $\bar{\pi}_{s_1}$, the initial probability estimation is obtained.

$$\bar{\pi}_i = Pr(s_1 = i|O, \lambda) = \gamma_1(i) \tag{12}$$

By maximising the second term of the right hand side in terms of $\bar{a}_{s_t s_{t+1}}$, the transition probability estimation is obtained.

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} Pr(s_t=i, s_{t+1}=j|O, \lambda)}{\sum_{j} \sum_{t=1}^{T-1} Pr(s_t=i, s_{t+1}=j|O, \lambda)} = \frac{\sum_{t=1}^{T-1} \gamma_t(i,j)}{\sum_{j} \sum_{t=1}^{T-1} \gamma_t(i,j)} \tag{13}$$

By maximising the third term of the right hand side in terms of $\bar{b}_{s_t}(o_t)$, the output probability estimation is obtained.

$$\bar{b}_i(k) = \frac{\sum_{t \in o_t = v_k} Pr(s_t=i|O, \lambda)}{\sum_{t} Pr(s_t=i|O, \lambda)} = \frac{\sum_{t \in o_t = v_k} \gamma_t(i)}{\sum_{t} \gamma_t(i)} \tag{14}$$

where v_k is the k-th codeword. $\gamma_t(i) = Pr(s_t=i|O, \lambda)$ is the probability that state is i at time t after having observed speech data O, then;

$$\gamma_t(i) = \alpha_t(i)\, \beta_t(i) / Pr(O|\lambda) \tag{15}$$

$\alpha_t(i)$, $\beta_t(i)$ are called forward probability and backward probability respectively, and expressed as followed:

$$\alpha_t(i) = \sum_{j} \alpha_{t-1}(j)\, a_{ji}\, b_i(o_t) \tag{16}$$

$$\beta_t(i) = \sum a_{ij}\, \beta_{t+1}(j)\, b_j(o_{t+1}) \tag{17}$$

$\gamma_t(i,j) = Pr(s_t=i, s_{t+1}=j|O, \lambda)$ is the transition probability from the state i to the state j at time t after having observed the speech data O and is expressed as followed:

$$\gamma_t(i,j) = \alpha_t(i)\, a_{ij}\, \beta_{t+1}(j)\, b_j(o_{t+1}) / Pr(O|\lambda) \tag{18}$$

V. Time Duration Modelling

In a basic HMM, the probability that speech data can be produced from one state decreases exponentially with time t according to the expression of $p(1-p)^t$, where the p is a transition probability from the state to the next state. This does not present the true property of the state because the short duration is always preferred. Ideally, there should be more representative limits on duration time, preventing both short durations as well as long durations in the state. This types of time duration probability can be modelled by a Gaussian distribution.

The conventional discrete HMM, whose output probability is computed on vector quantised codewords, can be extended to use time duration probabilities which are also represented in discrete form. The HMM parameters such as transition probability $\bar{a}_{ij}$, output probability $\bar{b}_j(k)$, and time duration probability $\bar{d}_j(\tau)$ can be directly obtained from the EM algorithm.[9]

Let $O=o_1\cdots o_t\cdots o_T$, $S=s_1\cdots s_k\cdots s_N$ and $D=\pi_1\cdots\pi_k\cdots\tau_N$ denote the quantised speech data, state sequence and time duration sequence respectively. The states included in the state sequence are different each other, unlike the discrete HMM without time duration. Then, the state s_k denotes the k-th state in the state sequence, not the state at time k. The $b_{s_k}(o_t)$ is the probability that the frame data o_t is produced from the state s_k. The $a_{s_k s_{k+1}}$ and π_{s_k} is the transition probability and initial probability defined in the same way as discrete HMM. The $d_k(\tau)$ is the probability that the state s_k is occupied for duration τ. Notation t_k is used to present the starting time of the k-th state so that the pair (t_k, τ_k) denotes the time information at the k-th state. Using these notations, the pdf of the complete data (O, S, D) is expressed as:

$$\begin{aligned} Pr(O,S,D|\lambda) &= Pr(O_{t_1}^{t_1+\tau_1-1}, s_k, \tau_1|\lambda) \\ &\cdots Pr(O_{t_k}^{t_k+\tau_k-1}, s_k, \tau_k|\lambda, O_1^{t_k-1}, s_1^{k-1}, t_k-1) \\ &\cdots Pr(O_{t_N}^{T}, s_N, \tau_N|\lambda, O_1^{t_N-1}, S_1^{N-1}, t_N-1) \end{aligned} \tag{19}$$

In the same way as the discrete HMM, we call $Pr(O_{t_k}^{t_k+\tau_k-1}, s_k, \tau_k|\lambda, O_1^{t_k-1}, s_1^{k-1}, t_k-1)$ the conditional pdf of the complete data at the state s_k. Under the assumption of data independency, one previous state dependency, and absolute time independency, the conditional pdf of the complete dats at the state s_k is simplified as follows:

$$\begin{aligned} &Pr(O_{t_k}^{t_k+\tau_k+1}, s_k, \tau_k|\lambda, O_1^{t_k-1}, s_1^{k-1}, t_k-1) \\ &\quad = Pr(s_k|\lambda, s_{k-1})\, Pr(O_{t_k}^{t_k+\tau_k-1}|\lambda, s_{k-1}, s_k)\, Pr(\tau_k|\lambda, s_{k-1}, s_k) \end{aligned} \tag{20}$$

We introduce the following terms to make the model more clear.

$$\begin{aligned} &\textit{Transition probability} && a_{s_{k-1}s_k} && = Pr(s_k|s_{k-1}, \lambda) && (2\le k\le T) \\ &\textit{initial probability} && \pi_{s_1} && = Pr(s_1|\lambda) && (k=1) \\ &\textit{output probability} && b_{s_k}(o_{t_k+n-1}) && = Pr(o_{t_k+n-1}|s_k, \lambda) && (1\le k\le N)(1\le n\le \tau_k) \\ &\textit{duration probability} && d_{s_k}(\tau_k) && = Pr(\tau_k|s_{k-1}, s_k, \lambda) && (1\le k\le N) \end{aligned} \tag{21}$$

Then the log-likelihood of the complete data is expressed as:

$$\log Pr(O, S, D|\lambda) = \log \pi_{s_1} + \sum_{k=1}^{N} \sum_{n=1}^{\tau_k} \log b_{s_k}(o_{t_k+n-1}) + \sum_{k=1}^{N} \log d_{s_k}(\tau_k) + \sum_{k=1}^{N-1} \log a_{s_k s_{k+1}} \tag{22}$$

Then Q function is expressed as:

$$\begin{aligned} Q(\lambda, \bar{\lambda}) &= \frac{1}{Pr(O|\lambda)} \sum_S \sum_D Pr(O, S, D|\lambda) \log Pr(O, S, D|\bar{\lambda}) \\ &= \frac{1}{Pr(O|\lambda)} \sum_S \sum_D Pr(O, S, D|\lambda) \log \bar{\pi}_{s_1} \\ &+ \frac{1}{Pr(O|\lambda)} \sum_S \sum_D Pr(O, S, D|\lambda) \sum_{k=1}^{N-1} \log \bar{a}_{s_k s_{k+1}} \\ &+ \frac{1}{Pr(O|\lambda)} \sum_S \sum_D Pr(O, S, D|\lambda) \sum_{k=1}^{N} \sum_{n=1}^{\tau_k} \log \bar{b}_{s_k}(o_{t_k+n-1}) \\ &+ \frac{1}{Pr(O|\lambda)} \sum_S \sum_D Pr(O, S, D|\lambda) \sum_{k=1}^{N} \log \bar{d}_{s_k}(\tau_k) \end{aligned} \tag{23}$$

By maximising the first term of the right hand side in terms of $\bar{\pi}_{s_1}$, the initial probability estimation is obtained.

$$\bar{\pi}_i = Pr(s_1 = i|O, \lambda) = \sum_\tau r_{0,\tau}(i) \tag{24}$$

By maximising the second term of the right hand side in terms of $\bar{a}_{s_k s_{k+1}}$, the transition probability estimation is obtained.

$$\bar{a}_{ij} = \frac{\sum_\tau \sum_t r_{t,\tau}(i,j)}{\sum_j \sum_\tau \sum_t r_{t,\tau}(i,j)} \tag{25}$$

By maximising the third term of the right hand side in terms of $\bar{b}_{s_k}(o_{t_k+n-1})$, the output probability estimation is obtained.

$$\bar{b}_i(l) = \frac{\sum_t \sum_\tau r_{t,\tau}(i)\, c_{t,\tau}(l)}{\sum_l \sum_t \sum_\tau r_{t,\tau}(i)\, c_{t,\tau}(l)} = \frac{\sum_t \sum_\tau r_{t,\tau}(i)\, c_{t,\tau}(l)}{\sum_t \sum_\tau r_{t,\tau}(i)\, \tau} \tag{26}$$

where $c_{t,\tau}(k)$ is the number of vector quantised output symbols v_k from time $t+1$ to $t+\tau$. The $r_{t,\tau}(i,j)$ is the time duration probability that speech data can be produced from the state s_j for duration τ, after transition from the state s_i to s_j at time t.

By maximising the fourth term of the right hand side in terms of $\bar{d}_{s_k}(\tau_k)$, the duration probability estimation is obtained.

$$\bar{d}_i(\tau) = \frac{\sum_t r_{t,\tau}(i)}{\sum_\tau \sum_t r_{t,\tau}(i)} \tag{27}$$

Here, $\gamma_{t,\tau}(i,j)$ is the probability that the state transition occurred from state i to state j at time t and the state j is occupied for duration τ, after having observed speech data O. It is calculated using the following expression:

$$\gamma_{t,\tau}(i,j) = \sum_{k=1}^{N-1} Pr(s_k = i, s_{k+1} = j, t_{k+1} = t, \tau_{k+1} = \tau \mid O, \lambda)$$
$$= \alpha_{t-1}(i)\, a_{ij}\, d_j(\tau) \prod_{n=1}^{\tau} b_j(o_{t+n-1})\, \beta_{t+\tau-1}(j) \tag{28}$$

and

$$\gamma_{t,\tau}(j) = \sum_i \gamma_{t,\tau}(i,j) \tag{29}$$

The $\alpha_t(j)$ and $\beta_t(i)$ is the forward probability and backward probability. They are computed by the following expression.

$$\alpha_t(j) = \sum_{\tau} \sum_{\substack{i \\ i \neq j}} \alpha_{t-\tau}(i)\, a_{ij}\, d_j(\tau) \prod_{n=1}^{\tau} b_j(o_{t-\tau+n}) \tag{30}$$

and

$$\beta_t(i) = \sum_{\tau} \sum_{\substack{j \\ j \neq i}} a_{ij}\, d_j(\tau) \prod_{n=1}^{\tau} b_j(o_{t+n})\, \beta_{t+\tau}(j) \tag{31}$$

In the case where uniform probability distribution is used as the initial time duration probability for $d_j(\tau)$ and the final probability is estimated this results in the Fergusson model.[5] In the case where a Gaussian probability distribution is used, this is termed here as the Gaussian model. The Gaussian model can prevent errors produced by short token durations or long token durations by virtue of its statistical nature.

VI. Phoneme Recognition Experiment

As a guide line for comparison, phoneme recognition experiments were carri-

Table 1, Conditions for acoustic analysis (AA), vector quantisation (VQ) and Hidden Markov Modelling (HMM).

AA	Sampling frequency	16kHz
	High-pass filter	$x_t - 0.97x_{t-1}$
	Feature parameter	LPC cepstrum (20th)
	Frame length	20 ms
	Frame shift	5 ms
	Window type	Hamming window
VQ	Codebook size	256 codewords
	Distance measure	Euclidean distance
	Data amount	100 frames/phoneme
HMM	Number of states	3 states
	Learning method	Baum-Welch Algorithm
	Recognition method	Viterbi Algorithm

ed out using basic discrete HMM without time duration modelling. The experi mental speech data used here consisted of 98 sentences which were spoken twice by a single speaker: these were hand-labelled to a phoneme level. Half of the sentences were used as training data and the remaining half were used for testing. The conditions of the acoustic analysis, vector quantisation and Hidden Markov Modelling are shown in Table 1.

The phoneme recognition results are shown in Table 2, organised by phoneme:

Table 2. Phoneme recognition results by HMM without time duration modelling.

Phoneme		Rate (%)	Phoneme		Rate (%)
@	(191)	53.4	n	(111)	64.9
i	(96)	34.4	l	(80)	56.3
ii	(68)	79.4	t	(76)	84.2
ai	(56)	73.2	m	(72)	70.8
a	(44)	31.8	r	(58)	89.7
ei	(43)	46.5	d	(60)	60.0
uu	(47)	72.3	w	(53)	77.4
ou	(38)	42.1	y	(47)	91.5
e	(38)	5.3	p	(47)	91.5
uh	(33)	36.4	z	(41)	73.2
oo	(32)	43.8	b	(39)	89.7
o	(21)	47.6	k	(36)	61.1
aa	(15)	20.0	s	(29)	96.6
u	(10)	10.0	ng	(27)	25.9
au	(8)	12.5	ch	(27)	63.0
i@	(6)	66.7	jh	(26)	61.5
@@	(5)	0.0	g	(24)	45.8
u@	(3)	0.0	dh	(23)	43.5
oi	(2)	50.0	v	(23)	56.5
e@	(3)	0.0	f	(19)	89.5
			th	(19)	73.7
			sh	(18)	38.9
			h	(17)	35.3
			zh	(17)	23.5
Total	(759)	59.8 (%)			

consonants on the right hand side, vowels on the left hand side. The number in the parentheses is the number of the data tokens used in training and testing. The recognition rate up to the first candidate was 59.8% as shown in Table 2. The recognition rate up to the second, third, fourth and fifth candidates were 76.3%, 84.3%, 88.6% and 91.4% respectively.

VII. Time Duration Effect

Phoneme recognition experiments were carried out by using discrete HMM with time duration modelling described in section V. In this experiment, we applied following, three kinds of time duration modelling according to the type of initial time duration probability $d_j(\tau)$.

(1) *Fergusson model*

The initial time duration probability $d_j(\tau)$ is set to be uniform distribution, and the EM algorithm is applied to estimate the parameters. In the Fergusson model, the time duration is limited up to 64 frames.

(2) *Gaussian model*

The initial time duration probability $d_j(\tau)$ is set to be Gaussian distribution, whose initial mean and standard deviation are computed from the training data. In the EM algorithm, time duration probabilities are updated and remodelled by Gaussian distributions at each iteration.

(3) *Enhanced Graussian Model*

The time duration probability computed in (2) is raised to the fifth power after setting the highest probability equal to unity only in the decoding process.

The result is shown in Table 3. The enhanced Gaussian model showed the highest phoneme recognition rates of 68.0%. The main reason for this increase is attributable to the time duration constraint which enables the most probable path to be optimised in the Viterbi decoding algorithm.

The recognition rates of vowels, consonants and total phonemes by time duration modelling are listed in Table 4, together with the result without time duration

Table 3, Comparison of discrete time duration modelling in phoneme recognition (%).

Without time	Fergusson	Gaussian	Enhanced Gaussian
59.8	61.2	62.8	68.0

Table 4. Phoneme recognition results by time duration modelling (%).

	Vowel	Consonant	Total phoneme
Without time	54.3	77.5	59.8
Time duration	63.9	80.2	68.0

modelling. In vowel and consonant recognition, separate codebooks were used. The time duration modelling increased vowel recognition rates by 9.6%. It is directly related to the total phoneme recognition rate. On the other hand, consonant accuracy increased by only 2.7%. The improvement of vowel recognition is attributed to the successful allocation of the respective states of vowel HMMs to stationary part of the vowel data by the time duration constraint.

VIII. Feature Subspace Effect

In Table 4, it can be seen that vowel recognition rate is 63.9% and is still lower than consonant rate by 16.3%, though it was improved by time duration modelling. In order to improve the vowel recognition rate still further, the most suitable feature space should be prepared for vowel recognition, instead of using LPC cepstral coefficients which are commonly used for both vowel and consonant recognition.

The formants are the most representative feautres for vowel because the resonance at the vocal tract causes the formants and produces the vowel sound. Vowel recognition was carried out using three formant frequencies: the first, second and third formants [10].

Table 5 shows the details of vowel recognition in the following feature spaces:

(1) Formant (F).
(2) Mel-formant (MF) which is the formant frequency after mel-transformation and has the effect to increase the frequency resolution around 1kHz.
(3) Mel-formant and its time derivative (MFD) which includes the mel-formant and its frequency derivatives between consecutive frames (5 ms).

In the Table, LPC cepstral coefficients with and without time duration modelling are listed for comparison. From the tablel time duration modelling is effective for the almost all vowels, and formant representation is most effective for short vowels like /a/, /e/ and /o/. The table also shows that the mel-formant representation is effective for diphthong and long vowels like /ai/, /ei/ and /ou/. The highest recognition rate 75.4% is achieved by the mel-formant and its derivative, mainly due to the non-linear transformation of the mel-frequency and its dynamic features.

IX. Conclusion

Two methods to improve the HMM-based phoneme recognition were described considering EM algorithm and its application to ordinary basic discrete HMM, the time duration modelling in the HMM frame work is analysed and formalised. The experimental results based on the formalisation of time duration significantly improved the phoneme recognition rate. Formants play an important role compared to the LPC coefficients in vowel recognition and showed considerable improvement of vowel recognition. The further work will be to integrate the recognition

Table 5. Detailed results of vowel recognition (%).

Phoneme		CEP	CEP time	F time	MF time	MFD time
@	(191)	58.1	75.4	62.8	60.7	70.7
i	(96)	52.1	52.1	53.1	58.2	66.7
ii	(68)	82.4	88.2	91.2	88.2	91.2
ai	(56)	85.7	94.6	92.9	98.2	100.0
a	(44)	36.4	38.6	86.4	86.4	90.9
ei	(43)	58.1	74.4	76.7	90.7	90.7
uu	(47)	70.2	80.9	83.0	85.1	85.1
ou	(38)	36.8	63.2	81.6	86.8	92.1
e	(38)	7.9	15.8	44.7	52.6	52.6
uh	(33)	30.3	36.4	42.4	54.6	51.5
oo	(32)	90.6	93.8	93.8	96.9	96.9
o	(21)	52.4	57.1	81.0	90.5	90.5
aa	(15)	13.3	13.3	66.7	66.7	46.7
u	(10)	0.0	0.0	10.0	20.0	0.0
au	(8)	12.5	12.5	25.0	25.0	12.5
i@	(6)	33.3	33.3	66.7	66.7	66.7
@@	(5)	0.0	20.0	20.0	40.0	40.0
u@	(3)	0.0	0.0	0.0	0.0	0.0
oi	(2)	50.0	50.0	50.0	0.0	0.0
e@	(3)	0.0	0.0	0.0	0.0	0.0
Total	(759)	54.3	63.9	68.9	71.8	75.4

CEP: Cepstral coefficients of 20th order.
F: Formant.
MF: Mel-formant.
MFD: Mel-formant and its derivative.
time: With time duration.
( ): The number of tokens.

results of vowel and consonants. Also the application of time duration HMM to phoneme or word spotting will be left for further work.

References

[1] F. Jelinek, "Continuous Speech Recognition by Statistical Methods," Proc. IEEE, Vol. 64, No. 4, pp. 532–556, 1976.

[2] A. Averbuch, et al., "Experiments with the TANGORA 20,000 Word Speech Recognizer," ICASSP87, pp. 701–704, 1987.

[3] Y.L. Chow, et al., "BYBLOS: The BBN Continuous Speech Recognition System," ICASSP 87, pp. 89–92, 1987.
[4] K.F. Lee and H.W. Hon, "Large-Vocabulary Speaker-Independent Continuous Speech Recognition using HMM," ICASSP88, pp. 123–126, 1988.
[5] S.E. Levinson, "Continuously Variable Duration Hidden Markov Models for Automatic Speech Recognition," Computer Speech and Language, No. 1, pp. 29–45, 1986.
[6] M.J. Russel and A.E. Cook, "Experimental Evaluation of Duration Modelling Techniques for Automatic Speech Recognition," ICASSP87, pp. 2376–2379, 1987.
[7] Y. Ariki and M.A. Jack, "Enhanced Time Duration Constraints in Hidden Markov Modelling for Phoneme Recognition," IEE Electronics Letters, Vol. 25, No. 13, pp. 824–825, 1989.
[8] Y. Ariki, F.R. McInnes and M.A. Jack, "Hierarchical Phoneme Discrimination by Hidden Markov Modelling using Cepstrum and Formant Information," ICASSP89, pp. 663–666, 1989.
[9] H.D. Huang, Y. Ariki and M.A. Jack, "Hidden Markov Models for Speech Recognition," Edinburgh University Press, 1990.
[10] A. Crowe and M.A. Jack, "Globally Optimising Formant Tracker using Generalised Centroids," IEE Electronics Letters, Vol. 23, No. 19, pp. 1019–1020, 1987.