GOLDEN MANDARIN(II) - AN IMPROVED SINGLE-CHIP REAL-TIME MANDARIN DICTATION MACHINE FOR CHINESE LANGUAGE WITH VERY LARGE VOCABULARY

Lin-shan Lee[1,2,3], Chiu-yu Tseng[4], Keh-Jiann Chen[3], I-Jung Hung[1], Ming-Yu Lee[1], Lee-Feng Chien[2], Yumin Lee[1], Renyuan Lyu[1] , Hsin-min Wang[1], Yung-Chuan Wu[1], Tung-Sheng Lin[1], Hung-yan Gu[2], Chi-ping Nee[1], Chun-Yi Liao[2], Yeng-Ju Yang[2], Yuan-Cheng Chang[2], Rung-chiung Yang[2]

National Taiwan University and Academia Sinica, Taipei, Taiwan, Republic of China *

ABSTRACT

Golden Mandarin (II) is an improved single-chip real-time Mandarin dictation machine for Chinese language with very large vocabulary for the input of unlimited Chinese sentences into computers using voice. In this dictation machine only a single chip Motorola DSP 96002D on an Ariel DSP-96 card is used, with a preliminary character correct rate around 95% in speaker dependent mode at a speed of 0.36 sec per character. This is achieved by many new techniques, primarily a segmental probability modeling technique for syllable recognition specially considering the characteristics of Mandarin syllables, and a word-lattice-based Chinese character bigram for character identification specially considering the structure of Chinese language.

I. INTRODUCTION

Today, the input of Chinese characters into computers is still a very difficult and unsolved problem. This is the basic motivation for the development of a Mandarin dictation machine. We defined the scope of this research by following limitations. The input speech is in the form of isolated syllables. The machine is speaker dependent. Reasonable errors are acceptable because they can be found on the screen and corrected from the keyboard by the user very easily. But the machine has to be able to recognize Mandarin speech with very large vocabulary and unlimited texts, because the input to computers can be arbitrary Chinese texts. Also, the machine has to work in real-time for computer input applications. A previous version of such a machine, Golden Mandarin (I), has been developed in 1990 [1][2], but the highly computation-intensive algorithms for Golden Mandarin (I) require 10 TMS 320C25 chips operating in parallel on 9 special hardware boards to meet the real time requirements. This is why the present machine is developed using completely different algorithms.

There are at least 10^5 commonly used Chinese words, each composed of one to several characters. There are at least 10^4 commonly used Chinese characters, all mono-syllabic. However, the total number of different syllables in Mandarin speech is only 1302. Based on such observation, the use of syllable as the dictation unit becomes a very natural choice. Another very special feature of Mandarin Chinese is that it is a tonal language. Every syllable is assigned a tone in general. There are basically four lexical tones and one neutral tone in Mandarin. It has been shown that the primary difference for the tones is in the pitch contours, and the tones are essentially independent of the other acoustic properties of the syllables. If the differences in tones are disregarded, only 408 base syllables (each bearing different tones) are required for Mandarin Chinese. This means the recognition of the syllables can be divided into two parallel procedures, the recognition of the tones, and of the 408 base syllables disregarding the tones. Based on the above considerations, the overall system structure for the Golden Mandarin (II) dictation machine is shown in Fig. 1. The system is basically divided into two subsystems. The first is to recognize the Mandarin syllables, and the second is to transform the series of syllables into Chinese characters, because every syllable can be shared by many homonym characters. For the first subsystem of syllable recognition, the base syllable (disregarding the tones) and the tone are recognized independently in parallel. For the second subsystem of language model, we need to first obtain all possible word hypothesis to construct a Chinese word lattice, and then use a word-lattice-based Chinese character bigram to select the most probable concatenation of word hypotheses as the output sentence.

II. MANDARIN SYLLABLE RECOGNITION

The recognition of the Mandarin syllables includes two parts: recognition of the 408 base syllables (disregarding the tones) and recognition of the tones. The tone recognition is not too difficult. Discrete Hidden Markov Models based on feature vectors of pitch frequency, difference pitch frequency, energy and difference energy are used, and the syllable durational cues are further applied to distinguish the neutral tone from the 4 lexical tones. The recognition of the 408 base syllables (disregarding the tones), however, is very difficult, because there exist 38 confusing sets in this vocabulary. A good example is the A-set, { a, ja, cha, sha, dsa, tsa, sa, ga, ka, ha, da, ta, na, la, ba, pa, ma, fa }. Specially trained continuous density Hidden Markov Models (HMM's)[3][4] for cepstral coefficients were used in the previous version machine [1][2], which are highly computation-intensive. Considering the fact that Mandarin mono-syllables have relatively simple phonetic structure and the primary problem in base syllable recognition is to distinguish the very confusing initial consonants instead of matching the entire template, it is therefore believed that the time warping functions of the state transition probabilities of HMM's are not very important. Because state transition path searching process in HMM's is highly computation-intensive, a segmental probability model (SPM) specially for Mandarin base syllables was therefore developed, which is very similar to continuous den-

*1. Dept. of Electrical Engineering, National Taiwan University

2. Dept. of Computer Science and Information Engineering, National Taiwan University

3. Institute of Information Science, Academia Sinica

4. Institute of History and Philology, Academia Sinica

0-7803-0946-4/93 $3.00 © 1993 IEEE

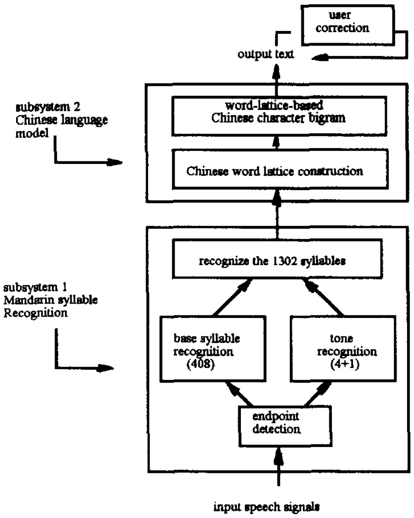

Figure 1: The overall system structure for the Golden Mandarin(II) dictation machine

sity HMM, but the state transition probabilities are deleted and the N states equally segment the syllable utterance.

In more detail, each utterance of syllable α is equally divided into N segments (or states), and each segment is modeled by M Gaussian mixtures. Each of the mixture is characterized by a mean vector $\hat{\mu}_{ij}$ and a covariance matrix $\hat{\sigma}_{ij}$, where $i = 1, 2, \ldots, N$ is the segment index, and $j = 1, 2, \ldots, M$ is the mixture index. The SPM of a syllable α is therefore represented by

$$S_{NM}(\alpha) = \{(\hat{\mu}_{ij}, \hat{\sigma}_{ij}), i = 1, 2, \ldots, N, j = 1, 2, \ldots, M\}$$

In the training phase, all training utterances for the syllable α are equally divided into N segments, and the feature vectors from the i-th segment of all training utterances are used together to train the parameters $(\hat{\mu}_{ij}, \hat{\sigma}_{ij})$, $j = 1, 2, \ldots, M$, for the i-th segment. They are first vector quantized into M clusters, and the feature vectors in the j-th cluster are used to obtain $(\hat{\mu}_{ij}, \hat{\sigma}_{ij})$. The covariance matrices $\hat{\sigma}_{ij}$ are assumed diagonal. In the recognition phase, the observation probability function $b_i(\hat{o})$ for an observed feature vector $\hat{o}$ with respect to the i-th segment of the syllable α is simply

$$b_i(\hat{o}) = \mathop{\text{Max}}\limits_{j=1,2,\ldots,M} \{b_{ij}(\hat{o})\}$$

where $b_{ij}(\hat{o})$ is the Gaussian distribution function defined by ($\hat{\mu}_{ij}, \hat{\sigma}_{ij}$) . In the recognition phase, an unknown utterance U is first equally divided into N segments, assuming each with n feature vectors,

$$U = \{\hat{o}_{ik}, i = 1, 2, \ldots, N, k = 1, 2, \ldots, n\}$$

where i is the segment index and k is the vector number in a segment. The observation probability of this unknown utterance U with respect to the SPM model of a syllable $\alpha, S_{NM}(\alpha)$, is then

$$Prob(U|S_{NM}(\alpha)) = \prod_{i=1}^{N}\left[\prod_{k=1}^{n} b_i(\hat{o}_{ik})\right] \equiv Prob(U|\alpha)$$

Apparently the syllable model giving the highest observation probability for U is the recognition output. In this way, both the training and recognition processes are simplified tremendously as compared to the continuous density HMM's.

The training data used in the experiments include 5 utterances for each of the 1302 syllables for each speaker, and the results below are the average scores obtained for two speakers. The recognition rates are listed in Table 1. The experiment (1) in the first row is for the continuous density HMM's used in the previous version machine [1][2], and experiment (2) in the second row is the initial test for SPM where N=7 and M=3, such that the number of states and mixtures are exactly the same in experiments (1) and (2) for parallel comparison. It can be seen that SPM gives an top 1 rate more than 20% lower than continuos density HMM's, apparently because SPM is a much more simplified version. However, the following series of improvements in experiments (3)~(6) in fact indicate the very high potential of SPM for Mandarin syllable recognition, if special characteristics of Mandarin syllables can be more carefully considered. Because the primary problem for Mandarin syllable recognition is to distinguish the very confusing initial consonants and some errors are often caused by the syllable ending (such as /an/ and /ang/),in experiment (3) smaller shift between adjacent speech frames was used in the first 20% and last 10% of the syllable utterances, such that finer signal characteristics can be extracted for the initial consonants and syllable ending. In the third row of Table 1, the top 1 rate was improved in this way from 69.85% to 75.49%. In experiments (4) and (5) optimal values of N and M were further found empirically and the linear prediction order P was increased from 10 to 14. Note that 3 segments (N=3) gives better results than 7 segments (N=7), probably because all the Mandarin syllables have relatively simple structure, composed of at most 3 to 4 phonemes. Without the time warping function of the state transition probabilities, too many segments (or states) may in fact cause interference among adjacent segments in the SPM. Therefore roughly one phoneme per segment turns out to be the best choice for SPM, although in HMM's 7 states gives the best results. On the other hand, because the computation load for SPM recognition is very low, increase of linear prediction order P from 10 to 14 can be easily achieved but is highly rewarding, as was indicated by the significant improvements in the top 1 rate, from 77.45% to 88.97%. Still further improvements can be achieved by a two-stage SPM approach in experiment (6) as shown in Fig. 2, in which the first stage SPM used cepstral coefficients, while the second stage SPM used regression coefficients obtained from cepstral coefficients, and the parameters M, N for the two stages can be separately optimized. The first stage selected the top L candidates $\alpha_1, \alpha_2, \ldots, \alpha_L$ and passed them to the second stage, together with the corresponding observation probabilities $P_1(U|\alpha_j), j = 1, 2, \ldots, L$. The final score of each of the L candidates is then the weighted sum of the observation probabilities obtained in the two stages. The last row of Table 1 indicates that in this way the top 1 rate can be as high as 96.57%, and the top 3 rate can be 99.75%. Note that the computation requirements in experiment (6) are still much less than those of continuous density HMM's used in experiment (1), but the performance is much more better. These results are also summarized in Fig. 3.

III. CHINESE LANGUAGE MODEL

After the base syllables and tones are recognized by the subsystem 1, the high degree of ambiguity caused by the large number of homonym characters still remain to be solved. The subsystem 2 thus acts as a linguistic decoder to identify the characters using context information. In the previous version machine [1][2], a relatively simple Chinese character bigram trained by primary

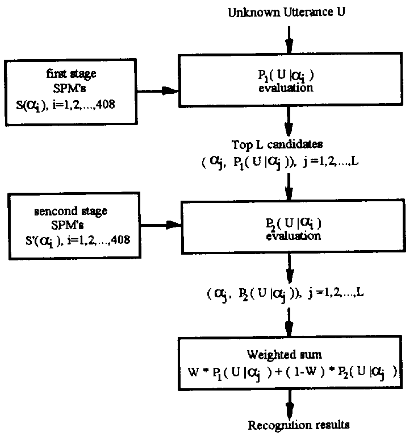
Unknown Utterance U
first stage
SPM's
$S(\alpha_i)$, i=1,2,...,408
$P_1(U|\alpha_i)$
evaluation
Top L candidates
$(\alpha_j, P_1(U|\alpha_j))$, j=1,2,...,L
sencond stage
SPM's
$S'(\alpha_i)$, i=1,2,...,408
$P_2(U|\alpha_i)$
evaluation
$(\alpha_j, P_2(U|\alpha_j))$, j=1,2,...,L
Weighted sum
$W * P_1(U|\alpha_j) + (1-W) * P_2(U|\alpha_j)$
Recognition results

Figure 2: The two-stage approach for SPM

school Chinese textbooks [5][6] was used, whose function was in fact limited. In Chinese language every word is composed of from one to several characters and there is no blanks between two adjacent words, thus a sentence can be considered as a sequence of words, or a sequence of characters. The 10^4 characters or 10^5 words require a character bigram of $10^4 \times 10^4$ probabilities or a word bigram of $10^5 \times 10^5$ probabilities respectively. Preliminary tests indicated that the word bigram is much more powerful than the character bigram [5], probably because the Chinese sentences are really built by words rather than by characters. But the word bigram is difficult to train and implement on a single chip because of the much larger size. A new approach considering the special structure of Chinese language using a word-lattice-based Chinese character bigram was thus developed to solve this problem. In this approach, the sequence of syllables obtained from the subsystem 1 is first matched with the words in a lexicon of 10^5 words to find all possible word hypotheses to construct a word lattice, with the help of a set of lexical rules. A word lattice is a graph of all possible paths connecting all word hypotheses, a simple example is shown in Fig. 4. The paths on the word lattice are then searched through by a word-lattice-based character bigram. The path with the highest probability is then chosen as the result, just as shown in Fig. 1.

Previous study [7][8] showed that grammatical information such as word formation are very helpful to statistical language models in grouping legal combinations of words while filtering out illegal ones. In Chinese language many compound words can be established by combining two or more words with simple rules, so they don't have to be stored in the lexicon. For example, the words " pig(豬) " and " Meat(肉) " can form a new word " Pork(豬肉) ", etc. These are the lexical rules mentioned above to help reduce the size of the lexicon. By matching the input syllable sequence with the words in the lexicon with the help of the lexical rules, all possible word hypotheses can be obtained and constructed in the word lattice. The function of the statistical language model can then be significantly reduced and simplified. For example, many noisy syllables or characters such as incorrectly recognized syllables or homonym characters which can't form word hypotheses with adjacent syllables or characters or can't be used as a mono-character word will be automatically deleted. On the other hand, if a set of adjacent word hypotheses can be grouped together earlier into a single compound word hypothesis in the word lattice, the number of possible paths connecting the word hypothesis in the lattice can be reduced. Also, it has been observed in preliminary experiments that a longer word hypothesis is usually more reliable, and in fact very probably it is exactly the correct answer if it is really long. Therefore the establishment of such a word lattice can not only reject the interference from many noisy syllables and characters, but significantly reduce the search space of the statistical language model and improve the overall accuracy.

Experiments	top1	top 2	top 3	top 4	top 5
(1) : previous HMM	91.67	98.53	99.51	99.75	99.75
(2):initial SPM	69.85	79.66	85.04	88.73	89.71
(3): finer framing	75.49	84.80	87.01	88.24	88.48
(4) :N=3, M=4	77.45	87.50	90.44	91.91	92.16
(5): P=14	88.97	97.06	99.02	99.51	99.51
(6): Two-stage	96.57	99.26	99.75	100	100

Table 1: Base syllable recognition rates for the previous HMM and the new SPM techniques

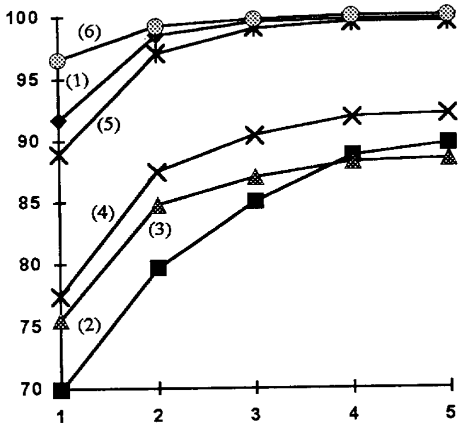

Figure 3: Recognition rates for the base syllables

After a word lattice was constructed as discussed above, a specially designed word-lattice-based Chinese character bigram was used to search through the word lattice to obtain the maximum likelihood output sentence. For each word hypothesis sequence $W = W_1 W_2 ... W_m$, where W_i is the i-th component word hypothesis, let $W_i = C_{i1} C_{i2} ... C_{is_i}$, where C_{ik} is the k-th component character of W_i and s_i is the number of characters in W_i , recalling that a Chinese word is composed of several characters. Then

$$
\begin{aligned}
P(W) &= P(W_1, W_2, ..., W_m) \\
&= P(\underbrace{C_{11} C_{12} \cdots C_{1S_1}}_{W_1}, \cdots, \underbrace{C_{i1} \cdots C_{iS_i}}_{W_i}, \cdots, \underbrace{C_{m1} C_{m2} \cdots C_{mS_m}}_{W_m})
\end{aligned}
$$

This probability can be approximated by

$$
\begin{aligned}
P(W) &= P(C_{11}) P(C_{21} | C_{1S_1}) \cdots P(C_{m1} | C_{(m-1)S_{m-1}}) \\
&= P(C_{11}) \cdot \prod_{i=2}^{m} P(C_{i1} | C_{(i-1)S_{i-1}})
\end{aligned}
$$

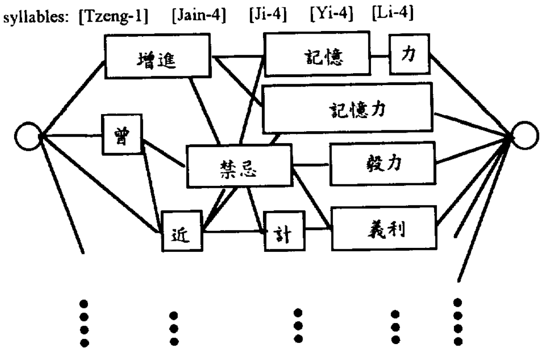

Figure 4: A partial list of an example word lattice. Each rectangle is a multi-character word, while each square is a mono-character word.

As can be found in Fig.5, this probability only considers the conditional probabilities for boundary characters in those adjacent word hypothesis, but ignores the conditional probabilities for characters within a word hypothesis, $P(C_{ik}|C_{i(k-1)}), 2 \leq k \leq S_i$ within W_i . This is because the characters within a word hypothesis are fixed and known, the conditional probability for adjacent characters can thus be assumed to be unity, when the word hypothesis is already constructed on the word lattice. In this way, the sentence hypotheses including longer word hypotheses will have higher probabilities. Therefore the longer word hypotheses will automatically have higher priority to be chosen, because they are in fact more reliable as mentioned previously. The word-lattice-based character bigram was trained in a similar way, i.e., the words in the training corpus were first segmented, and the bigram probabilities for the boundary characters were then estimated. Note that such a character bigram of $10^4 \bullet 10^4$ probabilities is relatively easy to handle and relatively robust with respect to insufficient training corpus, but the word lattice discussed previously can effectively enhance the capabilities of the character bigram to approximate a word bigram. In Golden Mandarin (II), the character bigram was trained by a corpus of 6 million characters taken from newspapers, magazines, and so on, and the top several base syllables and tones from the subsystem 1 are included in the word lattice.

IV. REAL TIME IMPLEMENTATION AND CONCLUDING REMARKS

In the real-time implementation of the Golden Mandarin (II) all necessary computation is performed in a single chip Motorola DSP 96002D, and the complete machine is implemented on an Ariel DSP-96 card inserted into an IBM PC/AT, while a ProPort Model/656 acts as the front end for acoustic signals. The waveform of the input unknown syllable is filtered and sampled in ProPort and transformed into 16-bit integer format, DSP96002D then sponsors all the following processes including endpoint detection, pre-emphasis, Mandarin syllable recognition and the Chinese language model. Preliminary tests indicate that in average it takes 0.36 sec for the machine to dictate a character, which is exactly real-time, and the character correct rate is around 95%.

Golden Mandarin (II) is the second version prototype system developed in a long term project, in which the goal is to solve the difficult problem of input of arbitrary Chinese text into computers using Mandarin speech. As compared to the first version, this machine is based on a single chip with all algorithms significantly simplified, but provides improved character correct rate at higher speed. This is achieved by many new techniques, primarily a segmental probability modeling technique for syllable recognition specially considering the characteristics of Mandarin syllables, and a word-lattice-based Chinese character bigram for character identification specially considering the structure of Chinese language.

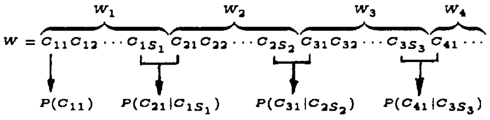
w_1
w_2
w_3
w_4
$W = c_{11}c_{12}\cdots c_{1S_1}c_{21}c_{22}\cdots c_{2S_2}c_{31}c_{32}\cdots c_{3S_3}c_{41}\cdots$
$P(c_{11})$
$P(c_{21}|c_{1S_1})$
$P(c_{31}|c_{2S_2})$
$P(c_{41}|c_{3S_3})$

Figure 5: The probability estimation in the word-lattice-based Chinese character bigram

ACKNOWLEDGMENT

The work described in this paper is supported by the National Science Council of Republic of China from 1984 to 1993, and performed in the Speech Laboratory of National Taiwan University in Taipei.

References

[1] L. S. Lee, C. Y. Tseng, H. Y. Gu, K. J. Chen, F. H. Liu, C. H. Chang, S. H. Hsieh, C. H. Chen, "A Real-time Mandarin Dictation Machine for Chinese Language with Unlimited Texts and Very Large Vocabulary," *ICASSP* , Apr 1990, Albuquerque, NM, USA, pp.65-68.

[2] L. S. Lee, C. Y. Tseng, H. Y. Gu, F. H. Liu, C. H. Chang, Y. H. Lin, Y. Lee, S. L. Tu, S. H. Hsieh, C. H. Chen,"Golden Mandarin (I) - A Real-time Mandarin Speech Dictation Machine for Chinese Language with Very Large Vocabulary," to appear in *IEEE Transactions on Speech and Audio Processing*, Vol. 1, NO. 2, Apr 1993.

[3] L. S. Lee, C. Y. Tseng, F. H. Liu, C. H. Chang, H. Y. Gu, S. H. Hsieh, C. H. Chen, "Special Speech Recognition Approaches for the Highly Confusing Mandarin Syllables Based on Hidden Markov Models," *Computer Speech and Language*, Vol. 5, No. 2, Apr 1991, pp.181-201.

[4] B. H. Juang and L. R. Rabiner, "Mixture Autoregressive Hidden Markov Models for Speech Signals,"*IEEE Transactions on ASSP*, pp. 1404-1413, 1985.

[5] H. Y. Gu, C. Y. Tseng and L. S. Lee, "Markov Modeling of Mandarin Chinese for Decoding the Phonetic Sequence into Chinese Characters," *Computer Speech and Language*, Vol. 5, No. 4, Oct 1991, pp.363-377.

[6] S. M. Katz, "Estimation of Probabilities from Sparse Data for the Language Model component of a Speech Recognizer," *IEEE Transactions on ASSP*, Vol. 35, pp.400- 411, 1987.

[7] L. F. Chien, K. C. Chen and L. S. Lee, "A Best-first Language Processing Model Integrating the Unification Grammar and Markov Language Model for Speech Recognition Applications, " to appear on *IEEE Transactions on Speech and Audio Processing* , Vol. 1, NO. 2, Apr 1993.