EXPERIMENTS ON CHINESE SPEECH RECOGNITION WITH TONAL MODELS AND PITCH ESTIMATION USING THE MANDARIN SPEECON DATA

Ying Sun[*†], *Daniel Willett*[*], *Raymond Brueckner*[*], *Rainer Gruhn*[*†], *Dirk Bühler*[†]

[*]Harman/Becker Automotive Systems, Speech Dialog Systems, Ulm, Germany
[†]Department of Information Technology, University of Ulm, Ulm, Germany
ysun@harmanbecker.com

ABSTRACT

Automatic speech recognition of a tonal and syllabic language such as Chinese Mandarin poses new challenges but also offers new opportunities. We present approaches and experimental results concerning the choice of base units for acoustic modeling, pitch estimation and how to integrate pitch estimates into the modeling framework. The experimental evaluations are carried out both on rather clean headset data and on noisy and reverberant distant talking speech data. Results show that tonal base units offer a word error rate reduction of more than 30% compared to toneless base units. This holds for both phoneme models and initial-final models. The integration of pitch as an additional feature stream yields another remarkable improvement of more than 20% over the best tonal baseline system. In a two-stream modeling approach, the pitch stream distributions can be strongly tied such that the overall model size increases only very moderately.

Index Terms: Chinese ASR, base unit selection, pitch

1. INTRODUCTION

Chinese Mandarin is a tonal syllabic language. Each Chinese character represents a syllable comprising a specific tone of the five tones, which are defined by characteristic pitch contours. For automatic speech recognition (ASR) this raises several interesting questions. One concerns the choice of the base model units, i.e. either phoneme-based or based on syllables, and which of the base units should be allowed to be tone-dependent. Another question regards the usefulness of explicitly extracting pitch features and how to best integrate them into the modeling framework.

In this paper we present a thorough investigation concerning these questions. Acoustic modeling is performed with HTK [1] and experimental evaluations are carried out on the Mandarin Speecon data base [2], which allows identical experiments on clean headset data as well as on noisy and reverberant distant talking microphone data.

2. CHINESE SPEECH RECOGNITION

Studies have been performed on Chinese speech recognition in many aspects. Concerning the choice of base model units, Xu et al. [3] have made experiments on three base unit sets (syllable, phoneme and Initial-Final) and found that the Initial-Final (IF) set shows best performance in pure Chinese speech recognition. Chen et al. [4] have defined a new base unit set consisting of premes and tonemes. A *premes* is a combination of the initial consonant with the glide of the final's vowel (if present). For example, the Initial L corresponds to four premes: $L\ LI\ LU\ LYU$. The intuition is that the shapes of the mouth for the four premes, even from the beginning, are very different; thus it might make sense to treat them as different base units. A *toneme* is the remaining part of the final including one of the five tones. In this representation the word $LUAN3$ is represented as $LU + AN3$ (as compared to $L + UAN3$ in a conventional IF system). However, the paper does not compare this base unit to the ordinary IF definition with respect to its ASR performance.

Concerning pitch extraction and incorporation into ASR systems, Chen et al. [4] have used pitch as an acoustic parameter. In order to compensate for the unvoiced sections in the pitch contour, they have applied a continuation algorithm based on a running average. In [5], investigations on the effect of pitch in large vocabulary continuous speech recognition (LVCSR) and isolated word recognition are reported. Considerable improvements are achieved once tone context and normalized pitch are incorporated into the ASR system. However, applying a language model to the baseline system is found to achieve comparable character accuracy. Thus, in LVCSR, pitch does not seem to substantially help improve the performance. Huang et al. [6] present an efficient real-time pitch tracker and a set of tone features, resulting in a vast 30% reduction of the character error rate compared to the non-tonal baseline. They show that both pitch and its derivative as well as the degree of voicing are useful features for tone recognition. However, long-term pitch normalization to a moving average is necessary to remove the negative influence of the speaker bias and the phrase effect of slowly falling pitch within the sentence.

3. THE CHOICE OF BASE UNITS

Table 1 and Table 2 show the base unit inventory of the phoneme- and the IF-based systems utilized in this study. The phonetic inventory is based on the SAMPA proposal as used in the Speecon lexicon [2].

Phonemes	_n, f, k, k_h, l, m, N, n, p, p_h, s, s=, s`, t, t_h, ts, ts=, ts=_h, ts_h, ts`, ts`_h, x, z`, @, {, X7, A a, a_?, E, e, E_r, i, I, i=, i`, o, U, u, y, w, j, H, @`, @~`, X7`, a`, a~`,E_r`, o`, u~`, u`
Initials	_I, _a, _e, _o, _u, _v, b, c, ch, d, f, g, h, j, k, l, m, n, p, q, r, s, sh, t, x, z, zh
Finals	a, ai, air, an, ang, angr, ao, aor, ar, e, ei, en, eng, engr, enr, er, i, ia, iai, ian, iang, iangr, iao, iaor, iar, ib, ie, ier, if, in, ing, ingr, inr, iong, iou, iour, ir, o, ong, ongr, or, ou, our, u, ua, uai, uair, uan, uang, uangr, uanr, uei, ueir, uen, ueng, uo, uor, ur, v, van, vanr, ve, ver, vn, vr, @

Table 1. Toneless base units used for acoustic modeling.

Phonemes	_n, f, k, e0, e1, e2, e3, e4, u0, u1, u2, u3, u4, ...
Initials	_I, _a, _e, _o, _u, _v, b, c, ch, d, f, g, h,...
Finals	a0, a1, a2, a3, a4, ai0, ai1, ai2, ai3, ai4,...

Table 2. Tonal base units used for acoustic modeling (initials are not marked with tone); 0 indicates neutral tone.

The IF inventory is based upon [3] whereas dummy initial models for initial-less syllables are applied ("ang = _a ang") to ensure that each syllable consists of an initial and a final in order to reduce the number of possible triphone combinations. In the tonal systems, only vowels and final models are split into up to five tone-dependent versions, which limits the number of base units. Consonants and initials will acquire tone-dependence anyway when turning to context-dependent models having tonal vowels and finals in the context. Depending on the choice of the base units, different methods of system parameter reduction by means of model tying can be applied. Some will be discussed and evaluated briefly in Section 6.

4. PITCH ESTIMATION

Several studies have indicated that for speech signals class-dependent correlation exists between the spectral envelope and pitch and that pitch can be predicted from MFCC vectors [7]. It is unclear, however, how robust the estimation process is in lower SNR scenarios. Hence, in our work we have adopted the approach of explicitly estimating pitch by the use of two different pitch estimation algorithms, a simple noise-robust pitch algorithm based on the *normalized autocorrelation* (NAC) [8, 9] and the widely used RAPT [10] algorithm as integrated in the Speech Filing System (SFS) [11].

The first algorithm is based on a frame-by-frame computation of the normalized autocorrelation $\widetilde{R}_{xx}$ of a signal window $x(n)$

$$R_{xx}(m) = \frac{1}{N} \sum_{n=0}^{N-m-1} x(n)x(n+m) \quad (1)$$

$$\widetilde{R}_{xx}(m) = \frac{R_{xx}(m)}{R_{xx}(0)} \quad (2)$$

of the low-pass filtered speech signal. The filtering is applied to obtain a smooth NAC contour. An N-best list of pitch candidates is computed by searching for the N largest positive peaks of the NAC in the relevant pitch range between approx. 70 and 400 Hz. To eliminate estimation errors such as pitch halvings and doublings and to smooth the "spiky" nature of pitch contours a Dynamic Time Warping (DTW) algorithm can be applied to the sequence of N-best pitch candidates. Furthermore, a voiced-/unvoiced decision can be made by comparing the maximum NAC coefficient to a threshold. A similar, but more elaborate algorithm, is the RAPT algorithm, which is described in detail in [10].

In order to robustly integrate pitch information into the acoustic model, a number of further measures were taken: Firstly, in order to reduce the inter-speaker variability each pitch estimate was normalized by a running average pitch, effectively removing differences in the pitch range due to speaker age or sex.

Secondly, the fact that pitch is present only in voiced frames creates discontinuities in the pitch contour. This might cause severe numerical problems in the training of the acoustic models [4]. Hence, continuation was enforced by letting the pitch variable $f_0(n)$ approach the running average pitch $\bar{f}_0(n)$ during unvoiced frames:

$$f_0(n) = \mu \cdot f_0(n-1) + (1-\mu) \cdot \bar{f}_0(n) \quad (3)$$

$$\bar{f}_0(n) = \rho \cdot \bar{f}_0(n-1) + (1-\rho) \cdot f_0(n-1) \quad (4)$$

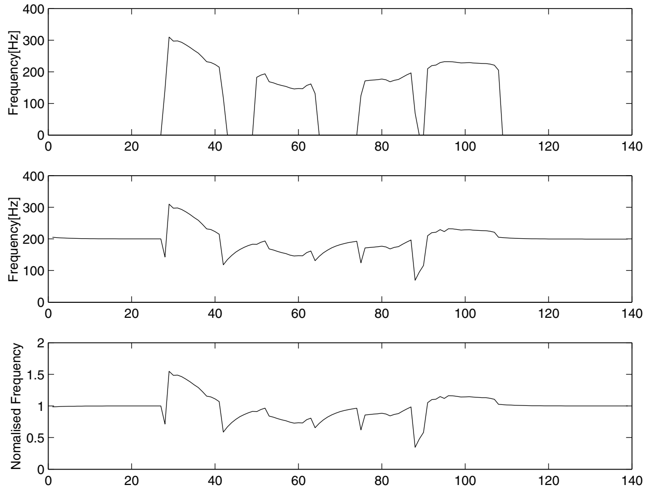

Fig. 1. RAPT pitch contour for utterance "xian4 chang3 zhi2 bo1"; upper: before postprocessing; middle: after applying a continuation algorithm; lower: normalization based on the continuated pitch.

where μ and ρ are the update parameters.

As described above, spikes occurring in the pitch contour can be eliminated with an N-best list and DTW. A computationally more efficient alternative is to use a 1-best list and subject it to a low-pass filter. In the experiments we used two different forms of the low-pass filter: a sliding average filter, replacing the center value of the window with the window mean, and a median filter, with the median value of the window instead. The window length was determined experimentally.

5. INTEGRATING PITCH AS A FEATURE IN ASR

We have followed two strategies to incorporate pitch as an additional feature: In the *one-stream* system the MFCC vector was complemented by the pitch estimate as an additional feature and the first and second derivatives of this MFCC+pitch vector were appended.

In general, in a multi-stream system using Gaussian mixtures the observation probability b_l of state l for observation $\mathbf{o}$ can be described as follows:

$$b_l(\mathbf{o}_n) = \prod_{s=1}^{S} \left[\sum_{m=1}^{M_{ls}} c_{lsm} \, \mathcal{N}(\mathbf{o}_{sn}; \mu_{lsm}, \Sigma_{lsm}) \right]^{\gamma_s} \quad (5)$$

with S being the number of streams, M_{ls} the number of mixtures for state l of stream s, c_{lsm} the mixture weight for mixture m, and the exponent γ_s the stream weight.

For $S = 2$ this yields the *two-stream* system in which the MFCCs and their derivatives constitute one stream, while the pitch and its derivatives constitute the second stream. One advantage of the two-stream approach is that we can assign different stream weights γ_s to emphasize a particular stream. Moreover, the separation into streams allows independent clustering of each stream's probability distribution functions (pdfs).

6. EXPERIMENTAL RESULTS

6.1. Setup

In our experimental evaluations spectral subtraction followed by the MFCC generation is performed by the Harman/Becker internal

Four Baseline Systems	#Monophones	#Triphones
Toneless Phoneme System	52	1438
Toneless IF System	93	1868
Tonal Phoneme System	152	2269
Tonal IF System	305	2226

Table 3. Number of base units in each baseline system.

		Clean Data		Noisy Data	
		Mono	Tri	Mono	Tri
Toneless Phoneme System WER (%)	digits	11.59	8.09	27.87	22.84
	cmd+cs	11.55	3.70	23.38	7.98
Toneless IF System WER (%)	digits	9.93	7.57	26.93	23.36
	cmd+cs	8.47	3.63	18.95	7.70
Tonal Phoneme System WER (%)	digits	10.81	7.95	26.12	22.05
	cmd+cs	9.49	2.44	19.82	5.87
Tonal IF System WER (%)	digits	9.98	8.60	25.34	23.39
	cmd+cs	6.29	2.51	13.55	6.64

Table 4. Word error rates measured on the four baseline systems on both clean and noisy data.

front-end while the acoustic modeling and decoding is performed with HTK [1]. Eleven MFCCs including the log-energy as well as Δ- and $\Delta\Delta$-coefficients are calculated. Each HMM state's pdf is modeled as a mixture of 8 Gaussian distributions with diagonal covariance. We have based our investigation on the Mandarin data collected within the Speecon project [2]. We refer to the headset data (channel0) as clean data and to the distant talking microphone data (channel2) as the noisy data. For each channel, 550 speakers were recorded, which results in over 100 hours of speech data. In all our experiments we use 80% of the speakers for training and the remaining 20% for testing. Tests are performed with two types of grammars: The first is an isolated word grammar which contains roughly 1000 words and comprises voice-commands and names of cities and streets. This will be referred to as "cmd+cs". The second is a digit loop grammar, which allows the recognition of digit sequences of arbitrary length.

6.2. Useful base units

Table 3 lists the number of base units (monophones) as well as the number of triphones in the four setups investigated. As we only train triphones appearing at least 100 times within the training data, the number of triphones only moderately varies between the systems, which makes system sizes and recognition performances comparable.

Table 4 shows the measured system performances of all possible combinations of models and data. In all experiments the performance on clean data is measured with models trained on clean data, while the performance of the noisy data is evaluated with different models that were trained on the noisy distant microphone data. Several important conclusions can be drawn from these results: When looking at the monophone system performance, IF models clearly outperform the phoneme models, but when turning to context-dependent models, there is no clear tendency anymore. In the case of the tonal model, the phoneme system even outperforms the IF-based one. Strikingly, the tonal models offer a vast improvement in accuracy without incorporating pitch as a dedicated additional feature. It is a 34% WER reduction on the cmd+cs test set compared to the toneless system in triphone modeling. This clearly indicates that pitch information is present in the MFCCs and confirms that the simplified view of tone purely affecting pitch without influencing other articulation parameters is only a rough approximation. The digit-loop tests hardly gain from tonal models, especially so with context-dependent models. Considering that there are many digit utterances within the speech data, even the toneless versions of the involved triphones mainly see digit data in the correct tone. Therefore, specializing the models to tone-dependent versions hardly changes the models involved in digit recognition.

		Clean Data		No. of models
		digits	cmd+cs	
Toneless IF System WER (%)	Triphone	7.57	3.63	1868
	Biphone	8.15	4.93	717
	Reduced	7.57	3.57	1660
Tonal Phoneme System WER (%)	Triphone	7.95	2.44	2269
	Reduced	8.29	2.63	1829

Table 5. Experimental results after parameter tying. "Triphone" refers to the original baseline system without parameter tying while "Reduced" refers to two different parameter tying methods based on the respective toneless and tonal triphone systems.

Because the best performance is achieved in the tonal phoneme setup and as phoneme-based modeling simplifies the modeling and recognition of English loan words, phonemes were chosen as base units for the following experiments.

Table 5 summarizes the results obtained from several approaches of model tying based on the toneless IF and tonal phoneme setups on the clean data (ref. Table 4). In the IF system, reducing the context-dependency on biphones in a way that considers context-dependency only within the syllables leads to a remarkable reduction in the number of models, but it also comes along with a considerable performance degradation. It appears to be of great importance to account for context across syllable boundaries and model tying must not be applied that aggressively. When tying all Final-Initial+Final models in which the left final context ends on the same sound and in which the right final context begins with the same sound (i.e. cluster ang-g-uai and ying-g+uo) the number of triphone models is reduced by over 10% with even a slight yet insignificant improvement in recognition accuracy. Reducing the number of context-dependent models in the phoneme-based setup is less successful. The last row of Table 5 shows the results when clustering all context-dependent versions of a phone when the context only differs with respect to the tone (i.e. clustering u2-H+y2 and u0-H+y3).

6.3. Integration of pitch as a feature

Table 6 lists the recognition performance observed with the rather cheap autocorrelation-based pitch estimator as an additional speech feature. In the one-stream setup, the pitch estimates (including Δ and $\Delta\Delta$) are simply appended to the MFCC feature vector. In the two-stream setup, the pitch estimates are modeled as a separate feature stream with each state's pitch pdf consisting of two Gaussian components and the stream weights are fixed at 0.2 and 0.8 for the pitch stream and the MFCC stream, respectively. These values were determined during preliminary experiments. It is obvious, that the pure pitch estimates without normalization and continuation as described in Section 4 significantly impair recognition accuracy. When pitch is normalized and continued, we see a remarkable improvement on the clean data, which leads to a 21% WER reduction compared to the baseline system, while we still observe a degradation on the noisy data. With the pitch estimates additionally low-pass filtered, we gain an improvement on the noisy data, too, at least for the cmd+cs test set. Looking at the two-stream models, we see much

		Clean WER (%)		Noisy WER (%)	
		digits	cmd+cs	digits	cmd+cs
one-stream	BL	7.95	2.44	21.98	5.82
	NAC0	8.60	5.46	-	-
	NAC1	6.65	1.93	23.54	6.96
	NAC2	6.69	2.19	23.88	5.13
	NAC3	8.73	2.16	24.03	8.37
two-stream	BL	7.95	2.44	21.98	5.82
	NAC1	5.85	2.04	18.89	4.57
	NAC2	5.88	1.95	18.84	4.45
	NAC3	6.03	2.06	18.60	4.56
	RAPT	6.71	1.92	18.26	4.06

Table 6. Experimental results on NAC and RAPT pitch after applying each essential processing step. BL: Tonal Phoneme-triphone baseline system. NAC0: BL+ NAC pitch. NAC1: NAC0+pitch continuation and normalization. NAC2: NAC1+smoothing with an average filter. NAC3: NAC1+smoothing with a median filter. The last row is BL+RAPT pitch with continuation and normalization.

	Clean Data WER (%)		N_1	N_2
	digits	cmd+cs		
NAC1	5.85	2.04	31k	31k
NAC1_1	6.06	2.05	31k	19k
NAC1_2	6.72	2.00	31k	660
NAC1_3	5.71	2.36	27k	31k
NAC1_4	6.76	2.58	27k	660

Table 7. Experimental results on NAC pitch using the two-stream strategy with mixture tying NAC1_1: NAC1+second stream vowel tying. NAC1_2: NAC1+second stream vowel+consonant tying. NAC1_3: NAC1+first stream tying. NAC1_4: Combination of NAC1_2 and NAC1_3. N_1 and N_2 refer to the number of distributions in the first and second stream, respectively.

more consistent and bigger improvements on both test sets. Here, the additional filtering has only a little, hardly significant influence.

The last row of Table 6 finally states the performance when using the more elaborate and expensive RAPT pitch estimates as features in the two-stream setup. It is obvious, that there is no consistent difference on the clean data. Our cheap estimator seems to be sufficient. On the noisy data, however, the RAPT pitch-based models show a significantly better performance with around 30% WER reduction on the cmd+cs test set compared to the pitchless baseline.

Table 7 lists the system performance evaluated on different setups of parameter tying based on the unclustered two-stream setup NAC1 (ref. the 7th row of Table 6). In NAC1_1, all vowel triphones of the same center vowel and tone share the same pitch stream distribution. In NAC1_2, in addition to the vowel tying of NAC1_1, all consonant triphones of the same center consonant share the same pitch stream distribution. In NAC1_3, all vowel triphones that only differ in the tone of the center vowel share the same MFCC stream distribution. The second stream is unclustered. NAC1_4 combines the second stream clustering of NAC1_2 and the first stream clustering of NAC1_3. It is obvious that the strong tying of the second stream comes along without additional error on the cmd+cs test set and that the moderate tying of the first stream even leads to some improvement on the digit loops. The overall tendency is not that clear, but it indicates that the pitch stream can be clustered strongly, so that the increase in model size of the two-stream system compared to the pitch-less one-stream system is very moderate.

7. CONCLUSION

We presented an investigation into the crucial questions that arise in the scenario of Chinese speech recognition. Experiments were performed on clean and noisy data of the Mandarin Speecon database. Concerning the question of useful base units, it turned out that once looking at context-dependent models, phoneme models perform at least just as good as syllable-based Initial-Final models. As phoneme models simplify the modeling of English loan words, this is regarded as the prime choice for future work. For both, phoneme models and IF models, tonal base units offer more than 30% improvement in recognition accuracy over toneless base units even without having a dedicated pitch feature. The integration of pitch as additional feature stream results in more than 20% WER reduction, which is another remarkable improvement. On clean data simple autocorrelation coefficients yield similar improvements as more elaborated pitch estimators, while on the noisy data, the more expensive RAPT pitch estimator appears to be worth it. The two-stream model with a separate pitch feature stream turned out to be beneficial in terms of recognition accuracy over the one-stream approach. It also offers the opportunity of different parameter tying strategies for each stream which was shown to yield well performing systems with only a moderate increase in system parameters over the pitch-less baseline system.

8. REFERENCES

[1] S. Young et al., ”The HTK Book (for HTK version 3.2)”, Cambrigde University Engineering Department, 2002.

[2] D. Iskra, B. Grosskopf, K. Marasek, H. van den Huevel, F. Diehl, and A. Kiessling ”SPEECON - Speech Databases for Consumer Devices: Database Specification and Validation”, Proc. LREC 2002, 2002.

[3] J. Xu, M. Xu, and J. Zhang, ”A Comparison of the Acoustic Units Modeling in Continuous Mandarin Speech Recognition: Syllables, Phonemes and Initial-Final”, Proc. 6th National Conference on Man-Machine Speech Communications, pp. 267–271, 2001.

[4] C.J. Chen, R.A. Gopinath, M.D. Monkowski, M.A. Picheny, and K. Shen, ”New Methods in Continuous Mandarin Speech Recognition”, Proc. Eurospeech’97, pp. 1543–1547, 1997.

[5] Y.W. Wong and E. Chang, ”The Effect of Pitch and Lexical Tone on Different Mandarin Speech Recognition Tasks”, Proc. Eurospeech’01, pp. 2741–2744, 2001.

[6] H. Huang and F. Seide, ”Pitch Tracking and Tone Features for Mandarin Speech Recognition”, Proc. ICASSP’00, pp. 3718–3721, 2000.

[7] X. Shao and M. Milner, ”Pitch Prediction From MFCC Vectors For Speech Reconstruction”, Proc. ICASSP’04, vol. I, pp.97–100, 2004.

[8] L.R. Rabiner, ”On the Use of Autocorrelation Analysis for Pitch Detection”, IEEE Transactions on Acoustics, Speech, And Signal Processing, 25(1):24–33, 1977.

[9] X. Huang, A. Acero, and H.-W. Hon, ”Spoken Language Processing”, Prentice Hall PTR, Upper Saddle River, New Jersey, 2001.

[10] D. Talkin, ”A Robust Algorithm for Pitch Tracking (RAPT)”, in Speech Coding and Synthesis, Elsevier Science B.V., 1995.

[11] “http://www.phon.ucl.ac.uk/resource/sfs/”.