CHAPTER 2 BACKGROUND AND RELATED WORK

This chapter presents a review of related research and relevant background for this thesis. We begin with giving a general idea about automatic tonal speech recognition (ATSR) and then discuss existing research related to this thesis. Section 2.1 provides tone recognition impact factors of ATSR. Section 2.2 discusses existing research in speech recognition systems for hardware implementation. This section also presents the concept of applying automatic speech recognition of non-tonal languages for tonal languages. In section 2.3, research in tone recognition is discussed. This section discusses conventional tone recognitions and the proposed method in term of calculation cost and recognition accuracy. Finally, we state our thesis statement. This provides an idea to improve related work performance.

2.1 Importance of Automatic Tonal Speech Recognition

Automatic speech recognition (ASR) is utilized in many applications, for instance, voice activation in intelligent home electronics, mobile phones, in-car devices, toys and robotic devices [4]-[9]. Tonal languages, for example Thai, Chinese, and Vietnamese, use tone to distinguish word meaning. Each tone provides a different grammatical meaning. Due to tone, when we apply ASR in tonal languages, recognition accuracy reduces, especially in words having the same phoneme but different tone.

In order to present an impact factor of ATSR, Figure 2.1 shows countries by population listed by the 30 most populous countries in 2013 [10].

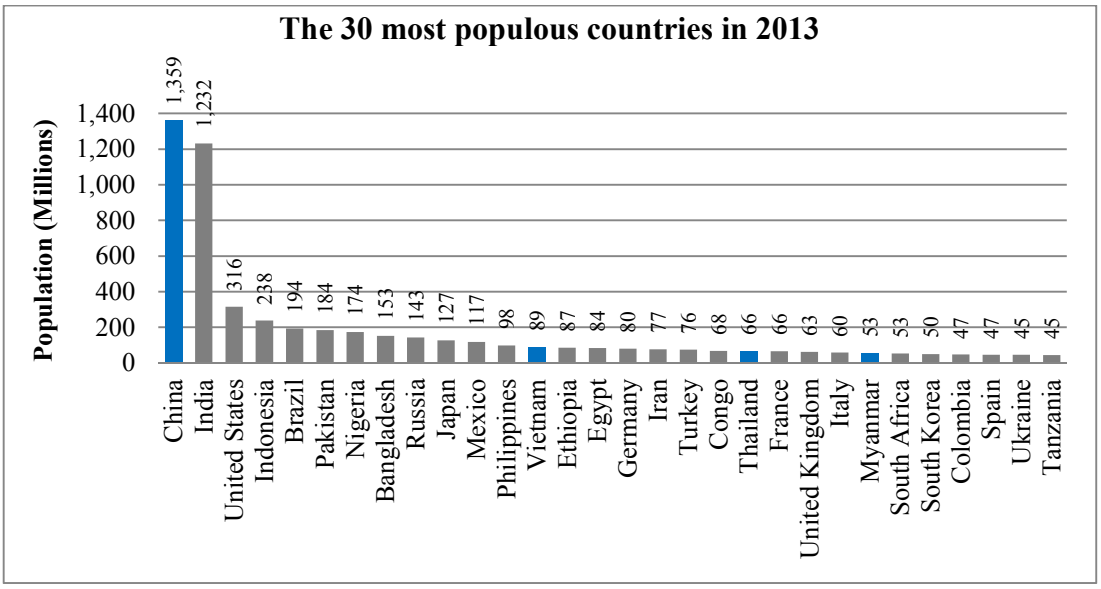


Figure 2.1 List of the 30 most populous countries in 2013.

According to Figure 2.1, there are over 1,500 million people using tonal languages, calculated by combining the number of population in China, Vietnam, Thailand, and Myanmar.

Since there is a large total population using tonal languages, speech recognition for tonal languages should be designed to improve recognition accuracy rate. Tone recognition is required to guarantee the word meaning in the automatic tonal speech

recognition (ATSR). Although tone recognition improves recognition accuracy, ATSR requires a higher calculation cost than ASR [11]-[13]. This results in higher calculation complexity. This thesis aims to reduce the complexity of tone recognition and thus ATSR can achieve real-time processing and consume low power.

2.2 Speech Recognition System for Hardware Implementation

Many researchers and developers have attempted to realize speech recognition for hardware implementation. Several mathematical methods are used to improve recognition accuracy such as hidden Markov's models (HMMs) [14]-[15] and dynamic time wrapping (DTW) [16]-[17]. While word HMM has a higher computation cost than phoneme HMM, it expresses high recognition accuracy in various environments. HMMs are therefore more effective than DTW in speaker-independent tasks [18].

HMM-based speech recognition has been considerably utilized in speech recognition technologies since HMMs provide high recognition accuracy rate even in large vocabulary continuous speech. Although HMMs provide high recognition accuracy, they take a long processing time because of the high computation costs in all reference models. Due to the high calculation cost of HMM, we discuss related works of HMM-based speech recognition suitable for hardware implementation.

There are many ways to implement HMMs on chip. Continuous Hidden Markov Model (CHMM) was implemented on MCU and Coprocessor [19]-[21] to reduce power consumption and a chip size.

Multiplier square accumulate calculation (MSAC) co-processor [22] has been implemented on application-specific integrated circuits (ASICs) to achieve real time processing and improve power efficiency. Software/hardware co-design was used for system on chip (SOC) implementation [23] to improve processing time. However, this method tested the design only on digit numbers 0-9. Speech recognition on DSP: issues on computational efficiency and performance analysis [24]: this paper proposed automatic speech recognition (ASR) algorithms on a fixed-point digital signal processor (DSP). The results showed very high recognition accuracy but the tested data included only 11 Cantonese words and 11 English digit strings. The Viterbi algorithm and HMM structure built in together method [25] was presented to calculate parallel computation of HMM with a 0.35-micron CMOS technology. This method showed high speed processing time and good word recognition performance. However, the method was only tested with small word vocabulary. Look-up technique [26] has been proposed to reduce the complexity of the HMMs circuit. However, it provides low processing time and low accuracy.

To improve the performance of HMM implementation, stored-based block parallel processing (StoredBPP) [27], shown in Figure 2.3, was proposed. StoredBPP provided less register, processing element, and processing time compared to stream-based block parallel processing (StreamBPP). Nevertheless, this method has been still achieved on hardware simulation.

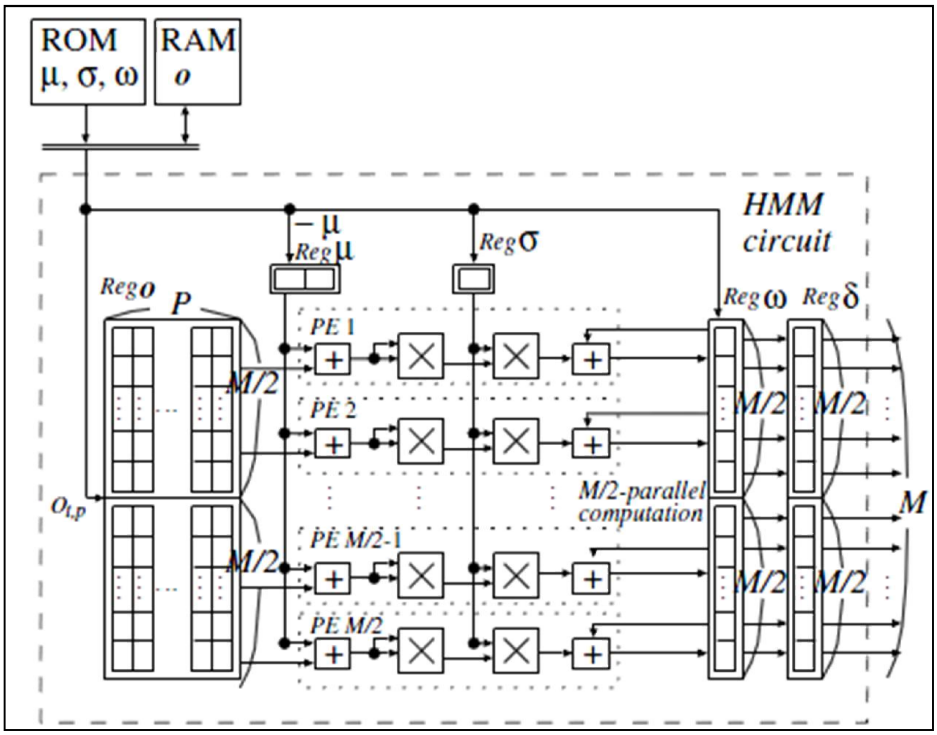
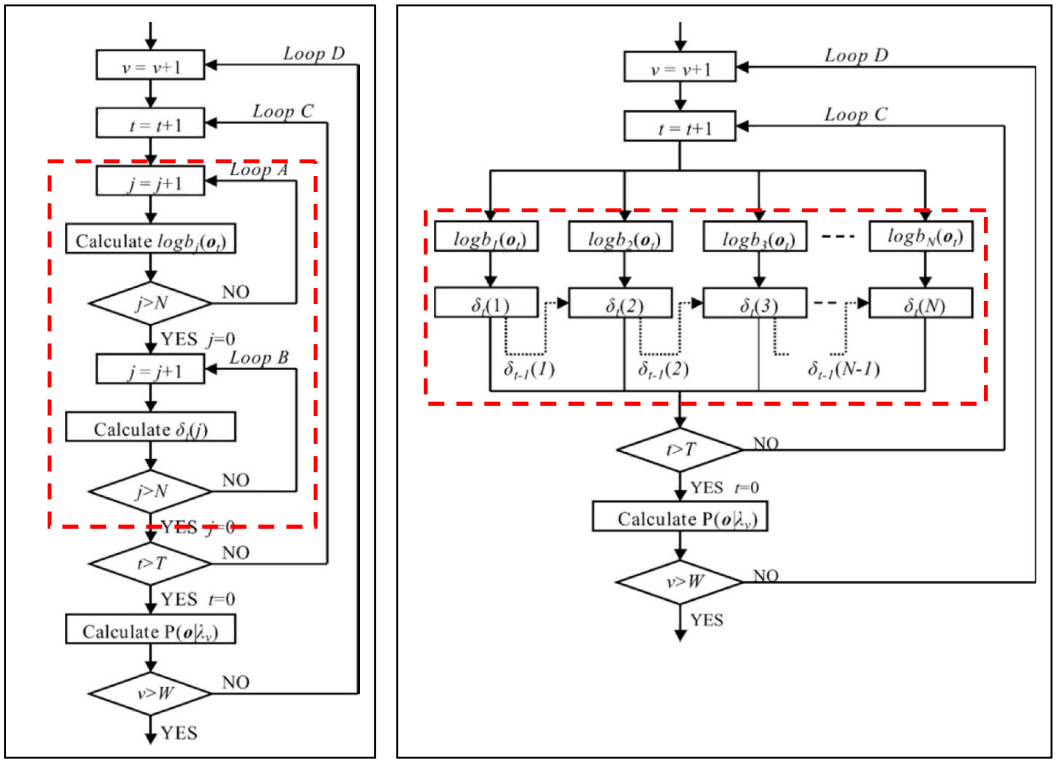


Figure 2.2 StoredBPP based VLSI architecture [27]

Yoshizawa *et al.* [28] proposed the VLSI system of scalable architecture for word HMM-based speech recognition for developing speech input interfaces to be embedded in practical application.

The architecture can extend/reduce word vocabulary during the recognition process, improve speed operation of the complete system, and also maintain high recognition accuracy. This research can be summarized as follows:

General HMM algorithm and proposed parallel HMM are described in Figure 2.3. Due to the series computation of general HMM algorithm, it has high computation cost. This paper has therefore modified the series computation of HMM to be in parallel computing in order to reduce HMM computation cost. In Figure 2.3a), loop A and loop B are converted to parallel computation in Figure 2.3b).



(a) (b)

Figure 2.3 (a) flowchart of general HMM computation (b) modified flowchart for parallel computing [28]

To achieve the VLSI system for parallel computation, this paper designed scalable architecture in a complete system as shown in Figure 2.4. The architecture included two methods: multiple process element and master-slave operation to provide high speed computation. Each method is described as follows:

Firstly, multiple process elements (PEs) are implemented inside the HMM module of complete system as shown in multiple process elements of Figure 2.4. PE1 is designed for log output probability calculation. PE2 is designed for Log-Viterbi algorithm. Then output parameters of PE2 are sent to compare with reference models.

In another process, a master-slave operation works as the bus control module of the complete system as shown in Figure 2.4 master-slave operation. This module executes speech analysis and robust processing only in the master system. Therefore, power consumption is reduced. In addition, feature vectors are transferred without handshaking between microprocessor and slave system. This results in data transfer processing time reduction.

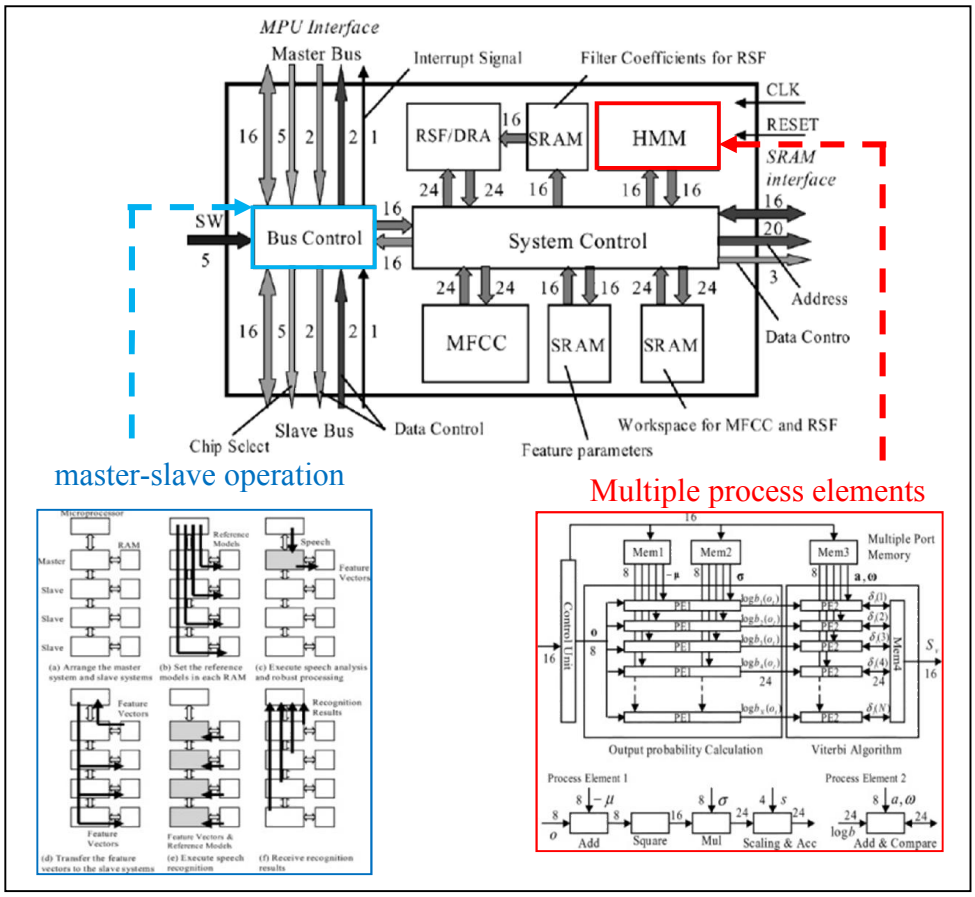


Figure 2.4 Complete recognition system [28]

P. Li *et al.* proposed a design of a low-power coprocessor for mid-size vocabulary speech recognition systems. A polynomial addition-based method is used to save hardware resources and reduce power consumption [29] without significant loss of accuracy, when they compute output probability calculation (OPC). The flow chart of the proposed OPC is shown in Figure 2.5.

In addition, they explore for the optimal number of PEs to achieve optimal tradeoff between speed processing delay, energy consumption, and hardware resources. This paper provides very low energy consumption per word recognition. However, it is limited to mid-size vocabulary (100-1000 words).

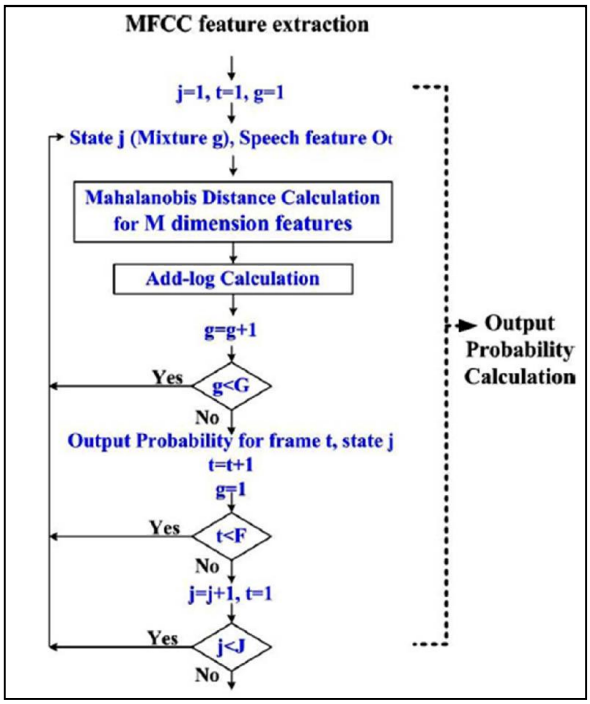


Figure 2.5 The proposed output probability calculation (OPC) flow chart [29].

He *et al.* proposes a low-power VLSI chip for 60,000 word real-time continuous speech recognition based on a context-dependent HMM [30]. They design a two-stage language model search to reduce cross-word transitions for the Viterbi search [31], beam pruning using a dynamic threshold to avoid sort processing in Figure 2.6.

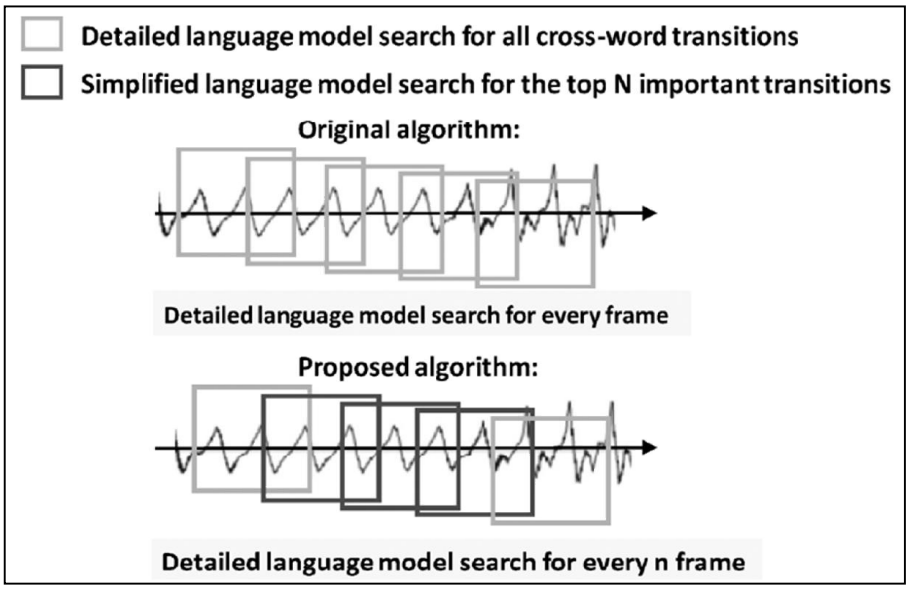


Figure 2.6 Two-stage language model search [30]

In addition, they reduce the memory bandwidth for Gaussian mixture model (GMM) computation using a variable-frame look-ahead scheme [32]. They entered part of the external DRAM data into the internal cache memory using the locality of speech recognition and also proposed specialized cache architecture shown in Figure 2.7 to improve the cache hit rate of 75%.

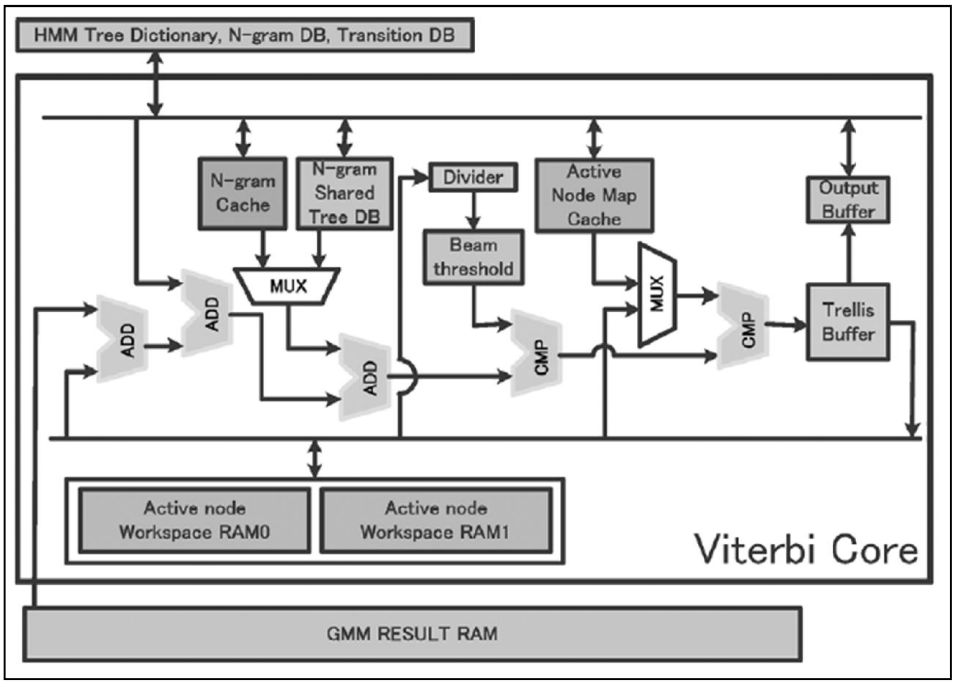


Figure 2.7 Viterbi cache architecture [30]

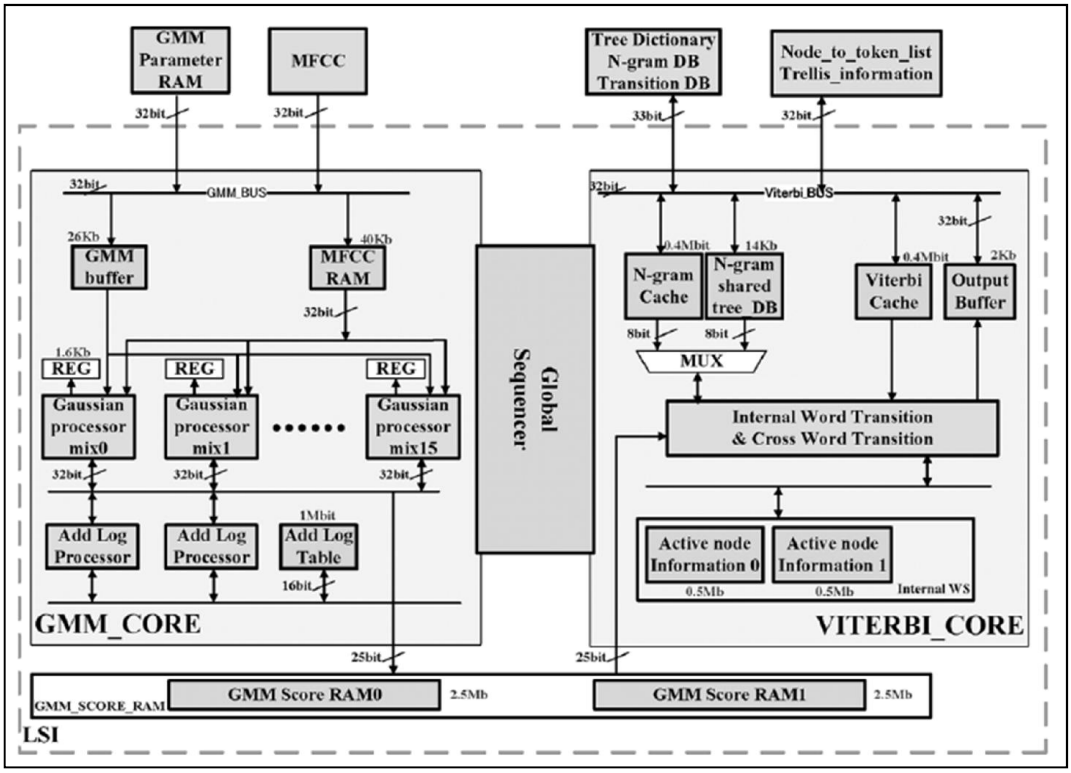


Figure 2.8 Proposed speech recognition [30]

Figure 2.8 shows the overall chip architecture. The operation flow of the highly parallel GMM calculation is shown in Figure 2.9. In Figure 2.10, data loading and GMM computation, and add-log processing are processed in parallel. Figure 2.11 illustrates the Viterbi transition flow. This transition is divided into three steps in each frame: word-internal transition, trellis-save [33], and cross-word transition.

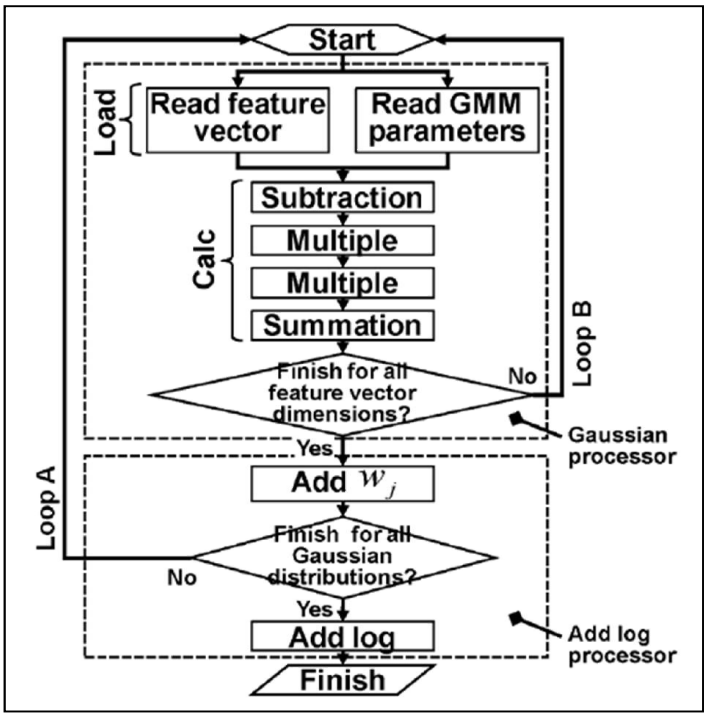


Figure 2.9 GMM computation flow

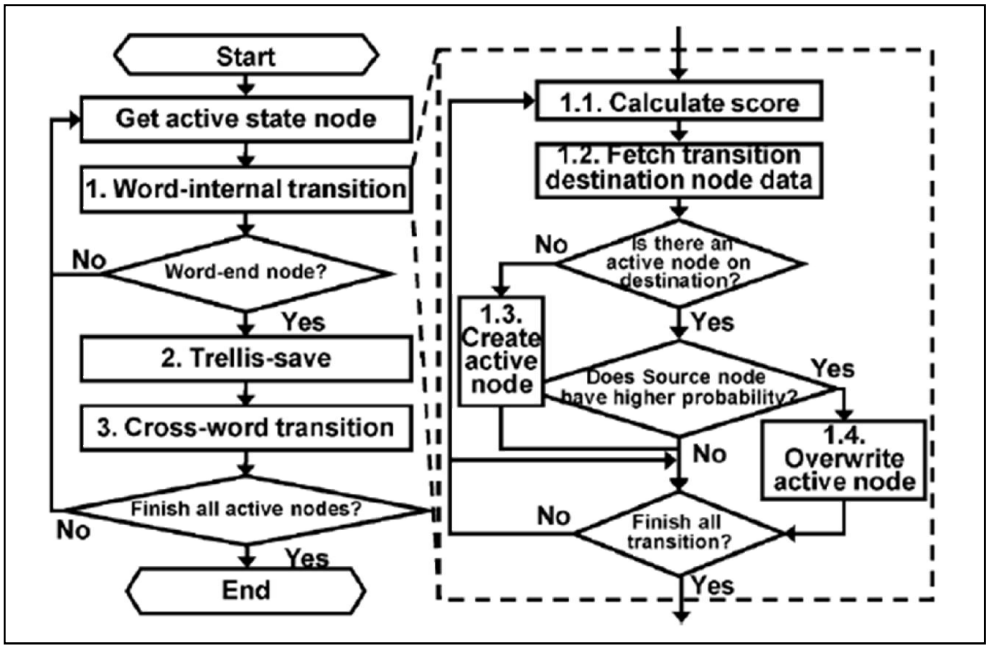


Figure 2.10 Viterbi computation flow

This results in 95% bandwidth reduction (70.86 MB/s) and 78% required frequency reduction (126.5 MHz).

While this research provides high performance, recognition accuracy is reduced when we apply this architecture in automatic tonal speech recognition (ATSR), especially in the case of words having the same phonemes but different tones.

Therefore, tone recognition system should be applied for improving recognition accuracy in ATSR.

2.3 Tone Recognition

Fundamental frequency (F_0) or pitch of voiced speech is utilized in speech analysis, synthesis and coding [34].

In non-tonal languages such as English and French, pitch variation called intonation is used to express emphasis, contrast, and emotion [35]. Intonation is used to automatically label phrasing finals in spontaneous conversation [36]. Intonation is also used to enhance intelligibility of speech for hearing-impaired speakers/listeners [37]-[38] and to detect mispronunciations in intonation of some languages spoken by non-native speakers [39]-[40].

In tonal languages, for example Thai, Chinese and Vietnamese, tone recognition is required to guarantee word meaning in automatic tonal speech recognition (ATSR). Tone is classified by fundamental frequency (F_0) contour [2]. The potential pitch frequency range is generally found between 50 and 250 Hz for men while the range usually falls between 120 and 500 Hz for women [34]. The pitch frequency ranges of elderly men and elderly women, people who are 70-80 years old, are lower than those of middle-aged people. Average F_0 of elderly men is 127.6 Hz while of elderly women is 187.7 Hz [41]. For children, average F_0 for boys is approximately 262 Hz and for girls approximately 281 Hz [42]. Tone recognition is generally developed in both isolated speech [43]-[44] and continuous speech. Continuous speech is more applicable for natural language which is typically used for communication.

In continuous speech, the concentrated F_0 is generally varied by the neighboring syllable F_0, and recognition accuracy is reduced because of incorrect F_0 features. To develop a highly efficient tone recognition, researchers should enhance F_0 so that it is the same as the ideal F_0 contours of each tonal language and/or develop high recognition accuracy tone recognition.

2.3.1 High Recognition Accuracy Tone Recognition

Firstly, we discuss high recognition accuracy tone recognition research. Many researchers have tried to improve tone recognition accuracy by mathematical techniques such as a multilayer perceptron neural network [45], hidden Markov models (HMMs) [46], and Fujisaki's model [47]-[48].

Potisuk *et al.* [2] proposed an analysis-by-synthesis algorithm for Thai tone classification in continuous speech. Fujisaki's model is used to find tone and reduce the tonal assimilation and declination effect which occur from neighboring syllables as shown in Figure 2.11. The top panel shows actual and synthesized F_0 contours based on

the extension to Fujisaki's model for continuous speech with an HLFHR tone sequence. The bottom panel shows a plot of the amplitude and temporal locations of the tone commands.

The experimental results show 89.1% recognition accuracy for speaker-dependent.

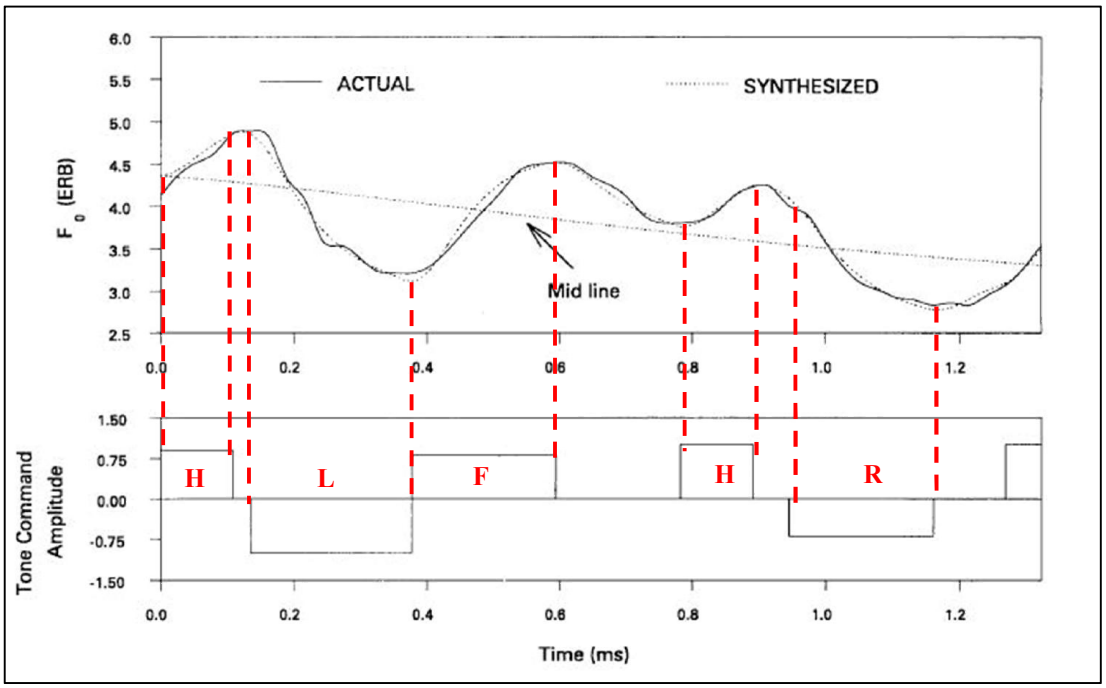


Figure 2.11 Actual and the synthesized F_0 contours (top) A plot of the amplitude and temporal locations of the tone commands (bottom) [2]

Thubthong *et al.* [49] proposed tone recognition for Thai continuous speech. The recognition reduces the assimilation and declination effect by using a half-tone model. F_0 is classified into first half and second half syllables. This model can reduce the assimilation and declination effect better than Potisuk *et al.* [2]. Figure 2.12 shows the tone features at different time points for baseline tone features, baseline tone features + tonal assimilation features, first-half tone features, and second-half tone features.

The recognition results showed 94.77% and 93.82% recognition accuracy for speaker-dependent and speaker-independent, respectively.

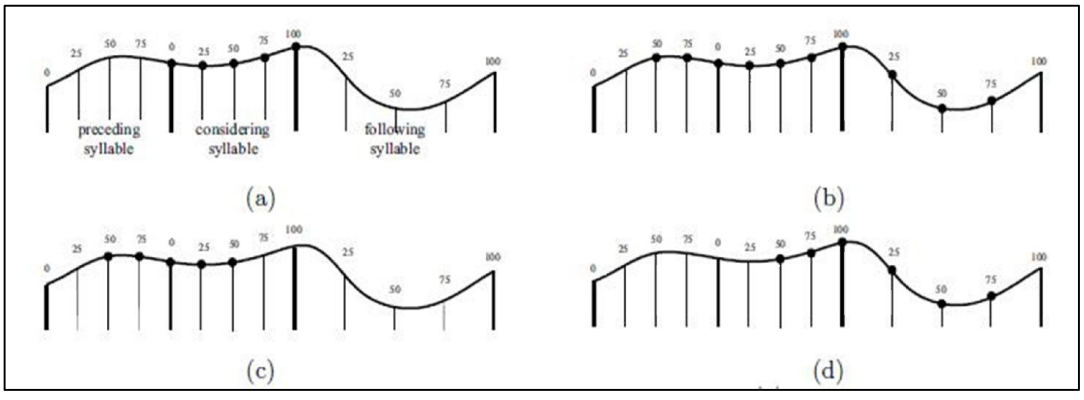


Figure 2.12 (a) baseline tone features (b) baseline tone features + tonal assimilation features (c) first-half tone features (d) second-half tone features

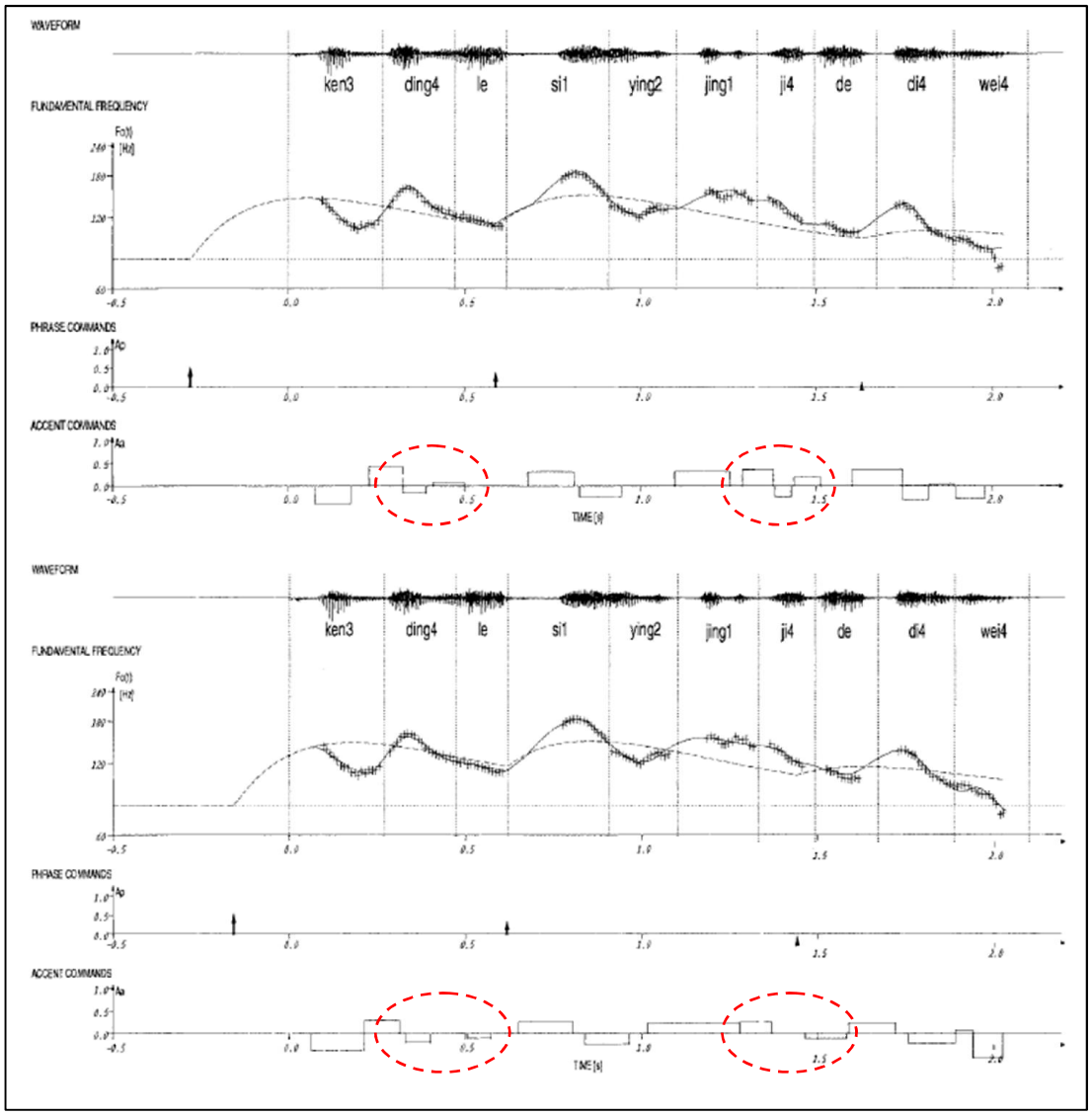


Figure 2.13 Example of the F_0 contour model parameter extraction for a spoken phrase of Standard Chinese read by Speaker B

Gu *et al.* [48] proposed automatic extraction of tone command parameters for the model of F_0 contour generation for standard Chinese. Since Chinese requires both positive and negative tone commands, they use information on syllable timing and tone labels to solve the extraction of model parameter problem. Figure 2.13 shows F_0 contours and the underlying tone command pattern for four lexical Chinese tones. Figure 2.14 indicates the example of the F_0 contour model parameter extraction for a spoken phrase of Standard Chinese. The dashed ovals show that some consecutive tone commands of the same polarity shown in manual extraction are merged as a result of automatic extraction. This results in error reduction.

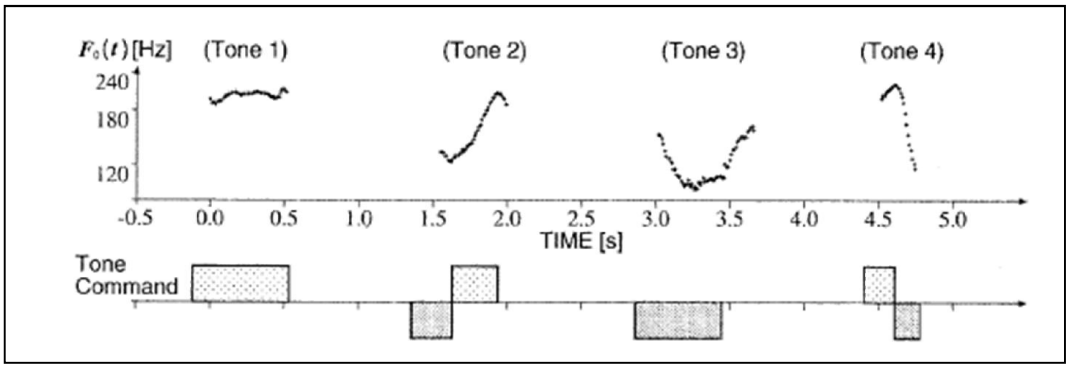


Figure 2.14 F_0 contours and underlying tone command pattern for four lexical Chinese tones of the word "yi"

2.3.2 F_0 Enhancement

There are many techniques for estimating pitch period in time domain and frequency domain. In frequency domain, the cepstrum analyzer [50] can detect both a pitch and voiced-unvoiced speech. However, harmonics tracking [51] has no pitch error compared with the cepstrum analyzer. Although harmonics tracking indicates very high performance, harmonics tracking gives high complex calculation. The average magnitude difference function (AMDF) and autocorrelation (AC) are generally used to estimate pitch period in time domain. The AMDF pitch detector [52] provides good estimation of pitch contour and the nature of AMDF operations is suitable for special purpose hardware.

In order to avoid the neighboring syllable effect in continuous speech, several researchers have enhanced F_0 contours by developing high efficient syllable segmentation and also by normalizing F_0 contour. Tone recognition of syllable segment Thai speech based on multilayer perceptron [53] not only identifies each syllable based on the relationship between the peaks and valleys of energy contours, but also normalizes F_0 contour. While this research provides an efficient recognition accuracy, it recognizes tone from the whole F_0 contour which is easily affected by adjacent syllables. AC coefficient is used to track the pitch of tone nucleus [54].

Wang *et al.* [54] proposed tone recognition of continuous Mandarin speech based on tone nucleus model and neural network. They estimate F_0 features without any negative influence from neighboring syllables by viewing only tone nucleus [55]-[56]. Due to the proposed tone modeling, it is not only provides a clear linguistic meaning for the normalization process but also shows explicit potentials for detection of intonation structure from the target points.

Tone nucleus modeling consists of three steps shown in Figure 2.15. There are four major modifications of the current algorithm compared with [55].

1. Autocorrelation coefficient computed during pitch tracking is incorporated together with delta logF0 since tone nucleus is considered to be associated with high autocorrelation coefficients.
2. The initial segmentation is changed from uniform ratio $n_1 : n_2 : n_3 = 1 : 1 : 1$ to non-uniform ratio $n_1 : n_2 : n_3 = 5 : 7 : 3$ due to imbalance of the articulate effects from previous and succeeding syllables to some extent.
3. T-test based segment merge is replaced by Viterbi decoding which allows state skip to realize segment merge.
4. If the segment result is two, one is the tone nucleus because it includes the larger number of points.

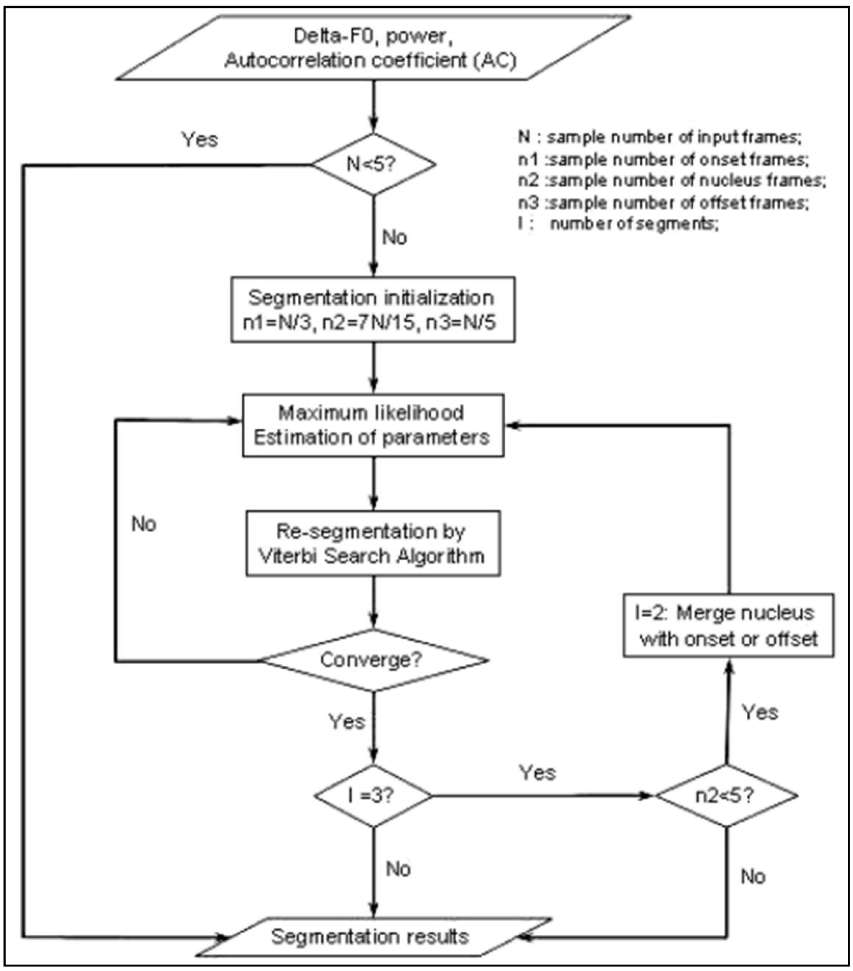


Figure 2.15 Tone nucleus extraction algorithm

Figure 2.16 shows examples of onset course, tone nuclei, and offset course, indexed by on, N, and off, respectively. The tone nucleus is shown in the dashed lines. Due to the tone modeling, this method can estimate F_0 features without any negative influence from neighboring syllables.

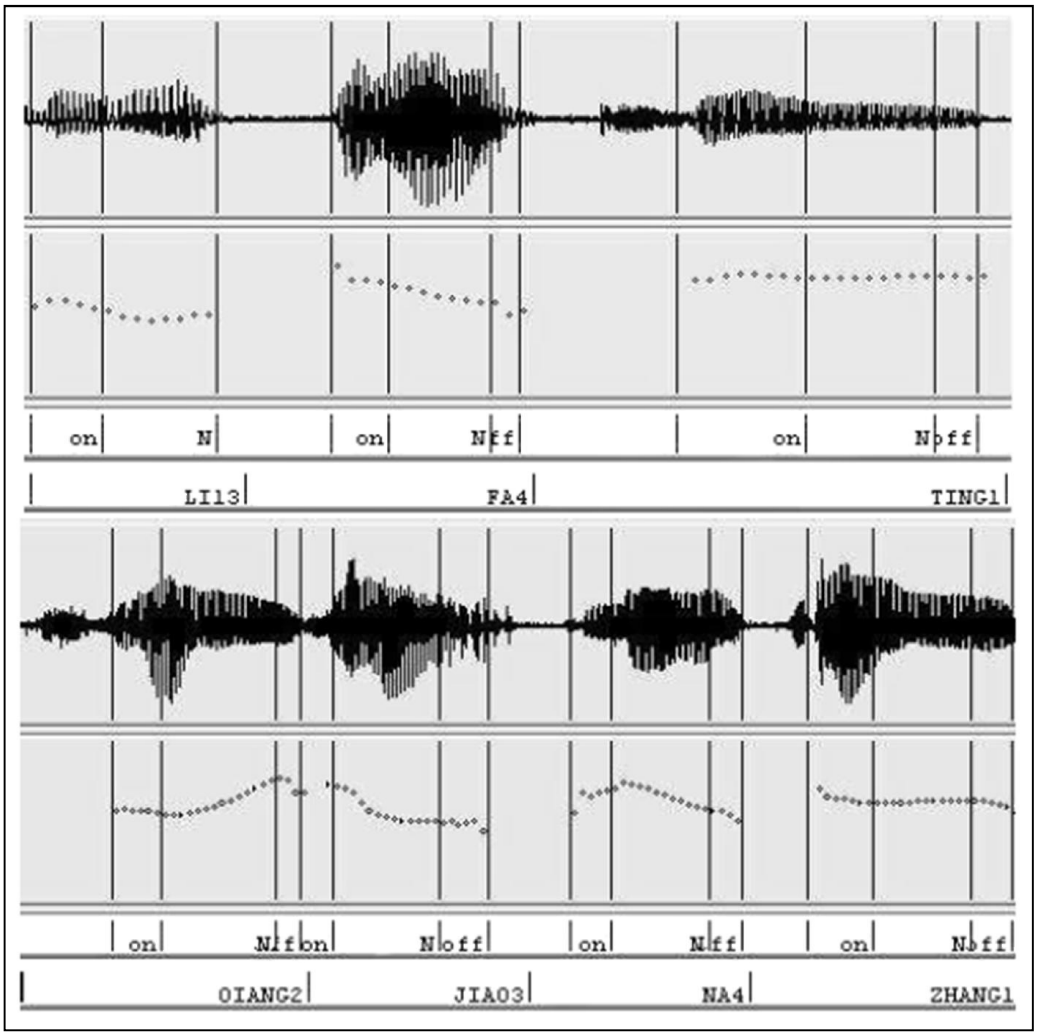


Figure 2.16 Examples of extracted onset course, tone nuclei, and offset course

The tone nucleus decreases F_0 error rate by severing the adjacent F_0 of each syllable. However, this research finds F_0 feature extraction, one of the highest calculation processes, from the whole input speech. In addition, this research has had to find the target F_0 using the tone nucleus process as a feature of the tone classifier and therefore too many processes are calculated.

2.4 Thesis Statement

There are two different kinds of speech recognition systems: network-based and non-network-based. The network-based system is suitable for continuous speech recognition and natural language speech recognition while the non-network-based speech recognition system is mainly applied to command-based speech recognition and word/phrase speech recognition. In non-network-based systems, automatic tonal speech recognition (ATSR) has to be implemented on mobile equipment. In this case, the ATSR should be real-time processing, and with low calculation complexity.

In order to develop ATSR with real-time and low calculation complexity, we first design a lower calculation cost method for tone recognition and then develop an architecture by which the above conditions can be sufficiently realized.

This thesis decreases tone recognition complexity by calculating tone from only three processes; feature extraction, F_0 estimation, and tone decision.

In the feature extraction process, the zero-crossing method is used to segment each syllable. Each word then detects only the vowel signal to find the pitch period calculated by a vowel magnitude difference function, called vowel-MDF (V_{MDF}). This feature extraction process not only reduces the number of input frame by approximately half of that used by the whole input speech but also gives no adjacent syllable influence.

Afterwards, the F_0 estimation finds F_0 from the first lower peak point. Since there is no neighboring syllable influence, the F_0 estimation provides a normalized F_0 without tone nucleus processes [54] as shown in Figure 2.17.

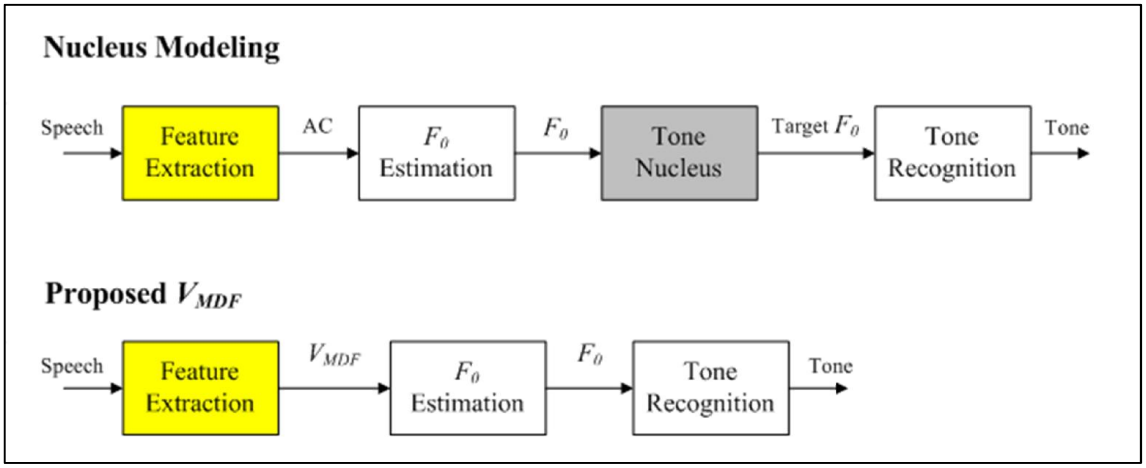


Figure 2.17 Tone nucleus modeling and proposed V_{MDF} comparison

Finally, the tone recognition classifies tones from F_0 contour characteristic via a proposed decision tree. Since there is no adjacent F_0 contour error, the tone recognition is able to classify five Thai tones without training data. Due to the reduction in the calculation costs, the reduced complexity tone recognition is able to decrease the complexity of the conventional tone recognition.

Series tone recognition operates a number of arithmetic operations in sequence. This results in a high computation cost, especially in the case of large vocabularies. This

thesis therefore develops the tone recognition in pipeline and parallel architecture to achieve a high throughput and accelerate the tone recognition.

The architecture is operated in 32 parallelisms of three process elements. Each process element (PE) is executed in pipelined processing. PE1 executes feature extraction process using a 3-stage pipeline process. PE2 executes V_{MDF} pitch period value in 3 stages and PE3 executes F_0 value in 2 stages. This results in a reduction of the processing time. Due to the reduction of the clock rate, the proposed architecture consumes less power, making it more suitable for portable electronic equipment. In addition, the architecture is able to adjust the maximum number of frames so that it can be applied to any input speech.