Speech Annotation by Multi-sensory Recording

Robert Luk
Department of Computing, Hong Kong Polytechnic University

Email: csrluk@comp.polyu.edu.hk

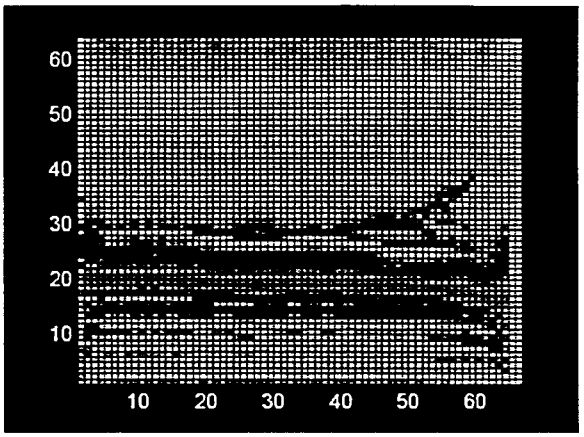
Abstract

This paper describes our effort to mark and annotate read Cantonese speech for both citation pronunciation and reading aloud sentences/phrases. Four signals are recorded simultaneously to assist marking and annotation: acoustic, laryngograph, nasal and air burst signals. A coarse match between voiced segments of speech and voiced segments of the phonetic spelling of the utterance is executed by dynamic programming as for approximate string matching. Finally, we discuss general issues in the design of our software for annotation.

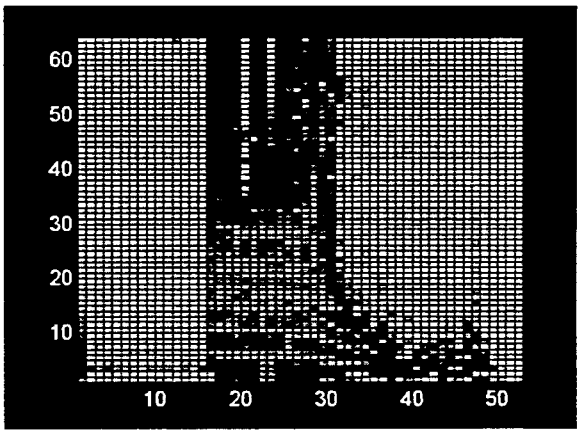
1. Introduction

This paper describes our effort to mark read speech based on multi-sensory recording. Although speech data are available for many languages (e.g. English, Putonghua, etc.), Cantonese speech data are still rare. Annotation of speech by hand is a tedious task, subject to errors and consistency problems. Therefore, we aim to annotate read Cantonese speech automatically. However, automatic annotation can be difficult even though the pronunciation of the speech sound is known because certain phonetic events are difficult to detect (e.g. plosives). We have adopted the multi-sensory technique developed for annotating English for Cantonese, after Chan and Fourcin [1].

Apart from annotation, marking speech data is also important for general speech analysis. For example, pitch synchronous Fourier transform can take the advantages of both a wide-band and a narrow-band Fourier transform where finer details of the spectrum are more apparent with pitch synchronous transform (Figure 1).



(a)



(b)

Figure 1: (a) Pitch Synchronous spectrum based on 64-point FFT with autocorrelation at double the pitch period (b) 64-point FFT with fixed window size of 200 and overlap of 50 sample points.

2. Multi-sensory recording

In this section, we describe the four signals that are simultaneously recorded. Next, we describe the physical set up for recording and the recording session.

2.1 Sensors

Four signals are received and recorded in a multi-sensory recording session and these are the acoustic signal (*Sp*), laryngograph signal (*Lx*), plosive signal (*Fx*) and nasal signal (*Nx*). *Lx* provides information about vocal fold vibrations and enables the identification of voiced/unvoiced segment as well as occurrence of each epoch (i.e. vocal fold closure). The latter is important for pitch detection as well as subsequent signal processing that are pitch synchronous. Figure 2 shows the use of *Lx* to define the voiced segment and epoch positions.

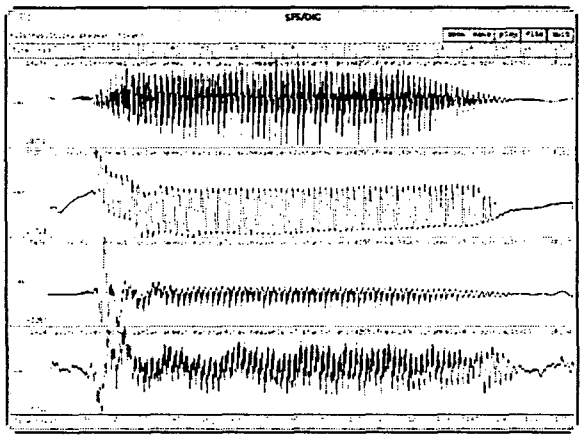


Figure 2: The multi-sensory recording of the syllable /pa/. The 4 channels from the top to bottom are: acoustic signal (*Sp*), laryngograph signal (*Lx*), turbulence signal (*Fx*) and nasal signal (*Nx*), respectively.

The *Fx* signal is picked up by a high-frequency sensitive miniature microphone placed 1 to 2 cm near the mouth. The signal is drastically attenuated so that only a sudden burst of air can provide sufficient excitation for recording. The burst of air is registered for aspiration or turbulence near the month (e.g. fricative and aspirated voice stop) which may be undetectable in the acoustic signal (*Sp*). Figure 3 shows the aspirated voice stop /p/ that is not registered in *Sp*. We anticipate that for continuous speech this type of events occur often.

A piezeo-ceramic transducer is placed near the nose bridge to detect nasal resonance. The signal from this transducer is *Nx* and it is useful to determine when nasalization occurs. This would be useful for detecting nasal consonants because it is simply an absorption of the vocalic energy, represented as a spectral zero. Figure 4 shows the recording of nasal resonance for the word /ma/.

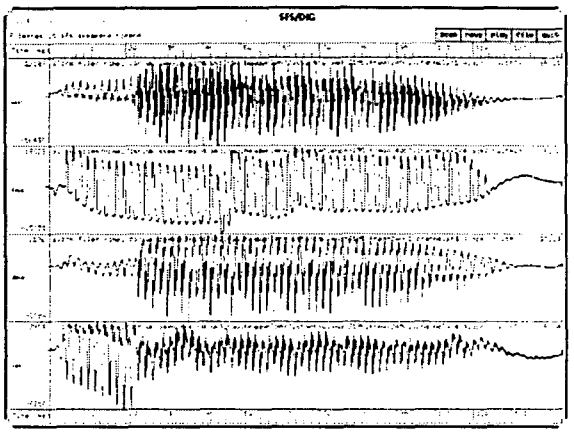


Figure 3: Multi-sensory recording of the syllable /ma/. Note that nasal resonance occur at the beginning in the bottom channel (Nx) and the amplitude is reduced in the top channel (Sp).

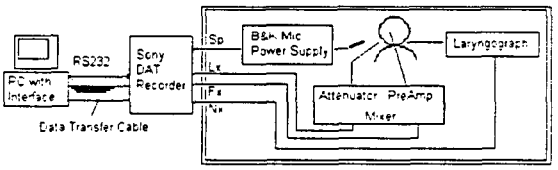


Figure 4: A schematic diagram that shows the physical set up of a multi-sensory recording sensors.

2.2. Recording Set Up

Recording is carried out in an anceohic chamber at City University of Hong Kong (Figure 5). The mixer, sensor, laryngograph and microphone power supply are placed inside the ancoehic chamber where as the recorder and PC are placed outside because of noise from cooling fans. The mixer provides amplification for the Nx signal and attenuation for the *Fx* signal. Likewise, the microphone power supply and the laryngograph provide amplification of the Sp and *Lx* signals, respectively. Four channel tape recordings are carried out first because they can serve as back up. Afterwards, the recorded data

are transferred to the PC by the computer interface under computer control via the RS232 link. It is possible to mark the beginning and ending of each utterance using the DAT recorder.

2.3 Recording Session

We have carried out recording isolated Cantonese [2] speech sounds as well as read speech of phrases and sentences. For isolated syllables, subjects are asked to pronounce all combinations of Cantonese initials, finals and tones which amounts to several thousand syllables. To save time and manual effort, the subject reads aloud a page of syllables (about 50) which are recorded on to the DAT tape before transfer to the PC. To maintain some consistency, subjects are asked to read aloud a carrier sentence by heart and pronounce only the target syllable.

For continuous read speech, subjects are given a list of sentences or phrases to read aloud. These sentences are selected from a corpus, that maximizes the coverage of Cantonese diphones based on a greedy algorithm [3]. The 104 sentences covered 348 Cantonese diphones. The corpus is a collection of news articles from the PH corpus [4].

3. Isolated Syllable Marking

Each file contains a set of syllables read aloud in a recording session. The 4 channels are sampled at 16kHz and quantized to 16 bits. The first step is to isolate the syllables from silence and label these syllables with the corresponding phonetic spelling augmented with a tone. Next, the four channel data is compressed into a marked speech data to save storage by a multiplicative factor of 4. The marked speech data uses the least significant three bits to encode where an epoch, some turbulence at the month, some nasalization or silence have occurred, according to the scheme shown in Table 1. Silence is also encoded because the recording will be carried out for an utterance instead of isolated syllables for later work.

Bit Pattern	Meaning
000	**Silence Presence**
011	**Not Silence**
1xx	**Epoch Presence**
010	**Plosive/Fricative Presence**
001	**Nasality Presence**

Table 1: Bit pattern scheme for representing the different marks of speech data. Nasal and turbulence are assumed not to simultaneously occur.

The least significant three bits instead of the most significant three bits are chosen for encoding because of compatibility reasons. The three bits can be considered as an additive noise component of magnitude at most 3 bits (i.e. 8). Usually, speech signals are much larger than 8 so that the noise due to the least 3 bits are almost negligible based on this encoding. We have found no noticeable degradation in the marked speech signal, which can be fed to other software like MATLAB as binary data.

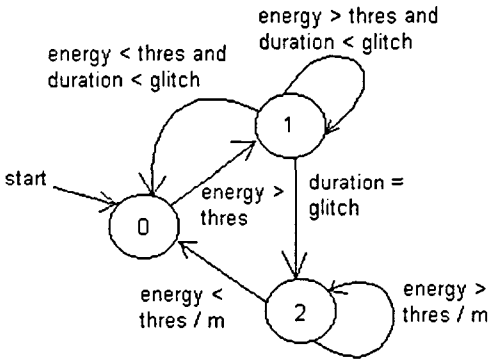


Figure 5: State transition diagram of the finite state machine for speech segmentation.

3.1 Speech Segmentation

The 4-channel recording is segmented based on the running energy of the speech signal *Sp*. A finite-state machine (FSM) keeps track of the segmentation decision (Figure 5). At state 0, the FSM considers the speech as silence. When the running energy is beyond a threshold *T*, the FSM makes a transition to state 1. The FSM remains in state 1 provided that the running energy remains beyond the threshold. Otherwise, it will make a transition back to state 0. If the FSM

remains in state 1 for a sufficiently long time that the speech signal cannot be a glitch, the FSM makes a transition to state 2. It will remain in state 2 if the running energy is beyond T divided by m. The multiplicative reduction m accounts for the steady reduction of speech energy near the end. Otherwise, the FSM makes a transition back to state 0.

3.2 Phonetic Spelling Labeling

Each segmented speech data corresponds to a syllable and the data has to be labeled with the corresponding phonetic spelling. Due to noise, sometimes glitches are mis-recognized as speech data and there are usually more segmented speech files than the amount of labels. We used a simple strategy to sort the data by size and delete the extra small files before labeling is carried out.

3.3 Epoch Detection

The detection of epoch is based on the Lx signal. The epoch is roughly located when the Lx signal is at the maximum near the largest change in the Lx signal. A simple detection strategy is to determine the first order backward difference:

$$DLx[i] = Lx[i] - Lx[i-2]$$

The detection selects those with a positive slope (i.e. $DLx[i] > 0$). A threshold T_e is set according to the following rule:

$$T_e = 0.1 \times \max_i \{DLx[i]\}$$

in order to decide those slopes which are definitely too small to consider for the identification of the epoch. Another threshold T_k is determined by the k-means algorithm which decides which of the remaining slopes are large enough and which are too small. Any remaining slopes, which are larger than T_e and which occurred consecutively, are deleted except at the last position. The remaining slopes positions are then the epoch positions (Figure 6).

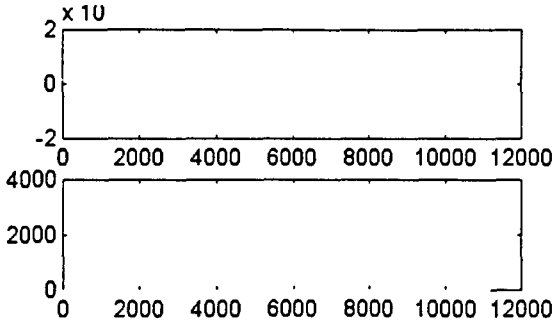


Figure 6: Pitch detection from the Lx signal shown at the top. The result is shown at the bottom where each spike represents the largest positive slop found.

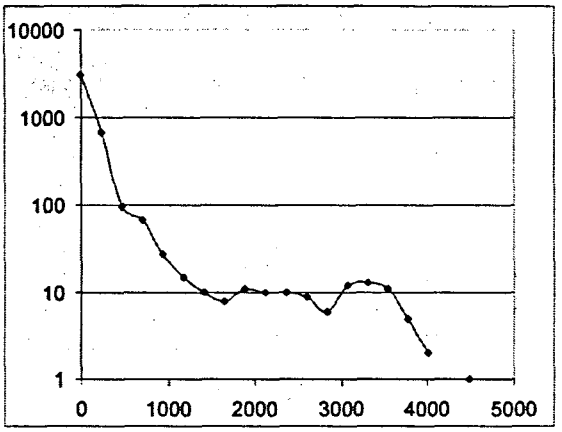


Figure 7: The frequency distribution of the amplitude of the positive backward difference of the Lx signal. The threshold was found to be 1530 by the k-means algorithm which is reasonable.

The k-means algorithm for determining T_k assumes there are two clusters: c_1 for slopes that are significantly large and c_2 for those slopes which are significantly small. Initially, the algorithm selects the two extreme slope values (i.e. maximum and minimum) as the centriod of the two respective clusters. A slope x is randomly selected and decided which cluster it belongs based on the following rule:

if $d(c_1,x) > d(c_2,x)$ **then**
$c_2 := \{x\} \cup c_2$
else
$c_1 := \{x\} \cup c_1$

where *d()* is the distance between the centroid of a cluster and the slope *x*. After each assignment, the centroid of the changed cluster is updated. Assignment of slopes to the two clusters is repeatedly carried out until no more slope values to assign (Figure 7).

3.4 Plosive/Fricative Detection

Certain plosives (e.g. /p/) and fricatives (e.g. /f/) produces turbulence near the month (Figure 8). This sudden burst of air is registered in the *Fx* signal as a sudden rise in magnitude. We follow Chan and Fourcin [1] to find the envelop of the *Fx* signal by first high-pass filtering (with sigma smoothing) the signal at 1kHz and smooth it by a median filter of length 201.

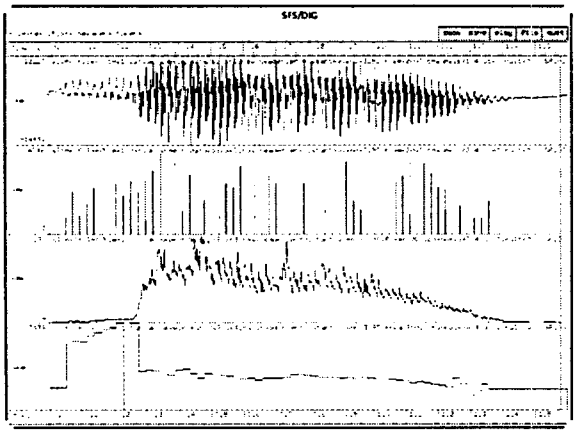


Figure 8: Post-processing of a multi-sensory recording of the speech /ma/.

3.5 Nasalization

The amount of nasality is computed based on both the *Nx* signal and the *Lx* signal as in [1]. Nasality is considered as the energy absorbed in the nasal cavity, reflected by the amount of nasal resonance picked up by the peizeo-ceramic transducer. The absolute value of *Nx* would indicate the amount of energy in the vibration but this has to be summed over one pitch period to indicate the amount of absorption for the pulse of air released in one vocal cord open-close cycle. Thus, we compute N_e as the sum of the absolute value of the *Nx* signal in one pitch period between two consecutive epoches.

The presence of nasality (Figure 8) is determined by a threshold T_N where any N_e value larger than T_N implies there exists some significant nasalization. To decide a better threshold between significant nasalization and insignificant nasalization, a different threshold T_n is used, which is determined by the k-means algorithm.

4. Continuous Speech Annotation

Annotation for a speaker-independent continuous speech is not an easy task without training. Our main idea is to find a reliable coarse match between the available phonetic spelling of the speech and perform additional processing to locate fine details.

A reliable cue is voicing which is available from the *Lx* signal because it is decoupled from the acoustic environment, making voice identification under extreme noisy environment possible. Also, since the *Lx* signal represents the source signal without convolving with the vocal track, it is relatively easy and reliable to detect the occurrence of pitch marks and therefore voicing. For matching phonetic spelling with speech sound, usually a syllable corresponds to a voice segment because each syllable must have a peak. Thus, the voice segment can be used the basic unit for finding the annotation of the speech.

4.1 Voiced segment identification

To detect voicing, the *Lx* signal is differenced and thresholded by the k-means algorithm, as in marking speech data (Section 3.3). In addition, the voice segment must have some continuity in the vocal cord vibration which restricts the duration of the voice segment to have at least 2 cycles. Taking the range of pitch to be between 2 and 20ms [5], the duration of voiced segment must be at least 40ms long. Figure 9 shows an example of finding the voiced segments when reading aloud a sentence. The accuracy of the *Lx* signal is usually within 1 *Lx* cycle.

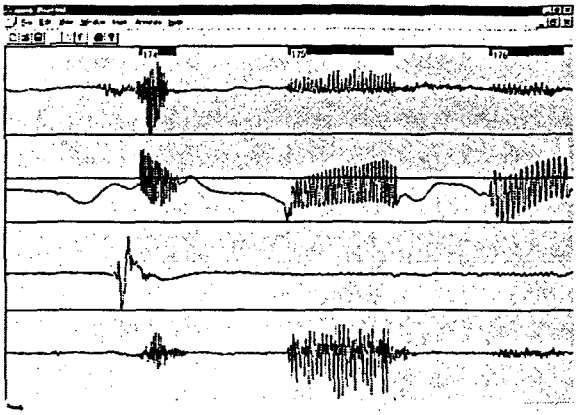


Figure 9: Detection of voiced segments using the *Lx* signal. The segments found are indicated by a color (blue) ribbon at the top of the signal.

4.2 Sentence/Phrase Boundary Detection

Sentence and phrase boundary can be manually marked by the DAT recorder or from the annotation software. The later is particularly tiresome because the amount of speech data is large, typically around 100Mbytes. Therefore, the visualization software takes time to scan and display the data.

The alternative explored in here is to automatically identify these sentence/phrase boundaries by measuring the duration between two voiced segments. If the duration is more than 900ms, a sentence/phrase boundary is found. However, the subject has to be aware of this arrangement for sentence/phrase separations.

4.3 Unvoiced Context Computation

For computational efficiency, unvoiced context computation only identifies the existence or absence of air burst, noise above 4kHz and nasal resonance. The existence and absence of these events are used for coarse matching between identified voice segment from speech and from phonetic spelling (Figure 10).

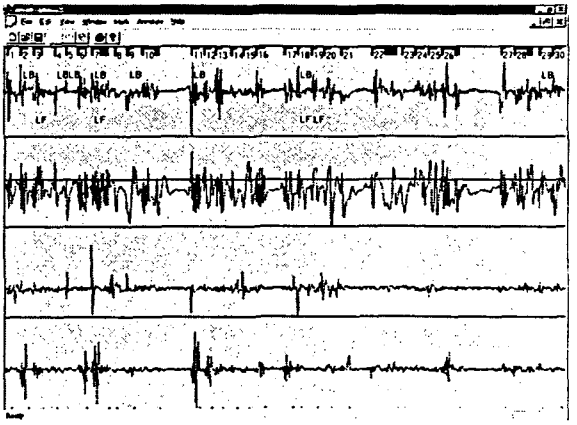


Figure 10: The detection of unvoiced contexts of an utterance of 30 syllables. Key: LB for the existence of an air burst in the left context of a voiced segment and LF for the existence of fricative noise in the left context. Since there are no right context air burst of fricative noise, they were not identified (as RB and RF respectively).

4.3.1 Air Burst Detection

Air burst is detected in the *Fx* signal. For each voice segment, the left and right contexts for air burst detection are between 10ms and 40ms away from the voiced segment. Within these two portions of the speech data, we obtain the maximum absolute differenced *Fx* signal. If this maximum is larger than a threshold (set at 800), then air burst is detected.

4.3.2 Fricative-like Noise

For fricative-like consonants, the turbulence is registered as noise above 4kHz in the Sp signal. Since these noise can extend quite far from the voiced segment, Sp signal between 10ms and 800 ms away from the voiced segment is examined. For each context, Sp signal is high-pass filtered at a cutoff of 4kHz. The filtered signal is differenced and the largest magnitude is compared with a threshold. If the signal is larger than the threshold, than fricative noise is present.

4.4 Coarse Matching

The aim of coarse matching is to associate the voiced segments of the phonetic spelling and

those identified in the speech signal. The voiced segment identified in the speech signal may represent one or more voiced segment of the phonetic spelling because of co-articulation. For example, the greeting sentence can have the following phonetic spelling /li ho ma/. The three voiced segments of this phonetic spelling are/li/, /ho/ and /ma/. However, in continuous speech, the voiced segment identified may be co-articulated to gather giving rise to only 2 voiced segments: /li homa/ since nasal /m/ and vowels /o/ and /a/ are voiced. Here, voicing has the special meaning that the vocal fold vibrates. Therefore, some voiced consonants like fricatives are not considered as voiced because the production does not involve vocal fold vibrations.

The voiced segments in the phonetic spelling and found in speech are temporally ordered so that these segments can be considered as strings where each character is a voiced segment. Coarse matching can be considered as a string matching problem but due to co-articulation approximate string matching that caters for merging voiced segment in matching is needed (Figure 11).

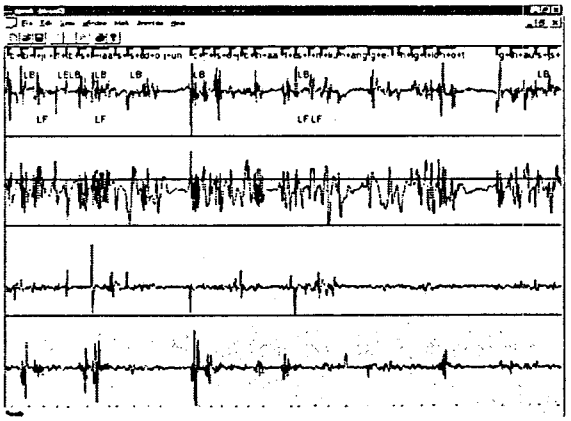


Figure 11: The labeled voiced speech segments by approximate string matching.

4.4.1. *Problem Formulation*

Let s be the sequence of voiced segments identified in the *Sp* signal. Likewise, let p be the sequence of voiced segments in the phonetic spelling of *Sp*. Let *s[i]* denote the i^{th} voiced segment and likewise for *p[i]*.

The distance *D(s,p)* between s and p is the minimal number of edit operations that transform s to p and vice versa. The minimal distance and the sequence of operations can be found by dynamic programming, using the following rule:

$$d[i,j] = min\ \{\ d[i-1,j] + insert(i-1,i,j),\ d[i,j-1] + delete(j-1,j,i),\ d[i-1,j-1] + sub(i,j)\}$$

where *d[i,j]* is the minimal edit distance from *(0,0)* to position *(i,j)*, representing the matching of voiced segments *s[0,i]* in *Sp* with those *p[0,j]* in the phonetic spelling.

4.4.2 Edit Distance

Unlike approximate string matching, the edit distance of insertion, deletion and substitutions are determined differently. For insertion, we consider the two voiced speech segments at i and i -1 are associated with a single voiced segment of phonetic spelling. Effectively, there is an error in voice segmentation where one of the segment (i or i + 1) is a spurious detection.

For deletion, the voiced segments of the phonetic spelling is associated with one voiced speech segment. Effectively, this edit operation is accounting for the co-articulation of two voiced segments as in /li homa/.

Such co-articulation does not occur freely. For example, if there are plosives or fricatives in the unvoiced context between the two voiced segments (e.g. co-articulation in /li/ and /ho/), then it is very unlikely that the voiced segments are co-articulated together. In addition, if the voiced speech segment is very short, then it is also unlikely that the two voiced segments of the phonetic spelling are read with the single voiced speech segment. Thus, both unvoiced context constraints and voiced segment duration are weighting factors of the deletion operation.

For substitution at position *(i, j)* , we consider whether the voiced speech segment is the same as the voiced segment of the phonetic spelling. If they have the same unvoiced context, the substitution cost would be low. Otherwise, the cost would be based on the number of mismatches in the unvoiced context.

5. Software Design and Implementation

Inevitably, it is necessary to check and correct automatic annotation. Software tool developed for this purposes needs to visualize a large volume of data that runs into hundreds of Mbytes. This is particularly the case for multi-sensory recording.

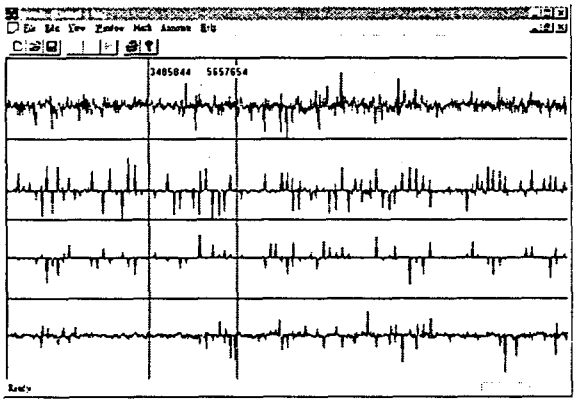


Figure 12: Visualization of the decimated version of speech file of size 110M bytes. Two markers define a region for magnification.

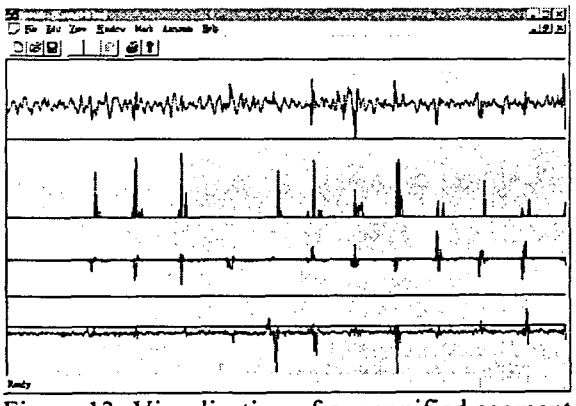


Figure 13: Visualization of a magnified segment of the speech data in Figure 12. Pitch mark identification was carried out as shown in the second channel.

For visualization, our software decimates the given speech data since the resolution of the screen is only 1024 (Figure 12). This provides a bird's eye view of the data. For speech data details, the user can zoom (Figure 13) into a region within two markers defined by clicking the mouse at the appropriate screen location. Within the magnified scale, the user can move (Figure 14) the speech data to the left or right of the current magnified region of data. The user can also save the marked region directly into a file. The name of the file can be automatically generated or found from a list of labels in a file.

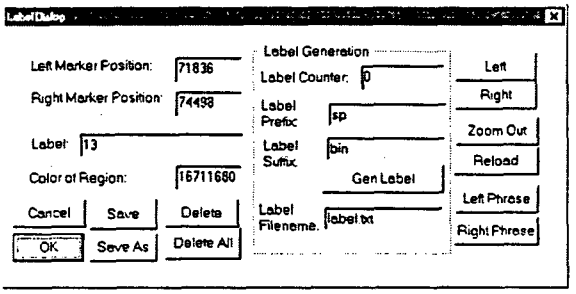


Figure 14: A dialog for moving, saving and manual labeling speech data. This dialog box is invoked when the user double clicks between two markers or within a voiced segment.

Signal processing for visualization is carried out with the data stored in the buffer and it is not directly operating on the speech data in the file. The purpose is to visualize the effect of setting parameters of certain signal processing function. Once the desired parameter values are found, signal processing is carried out for the speech data in the file. Since the buffer data is a decimated version of the data in the file, the signal processing parameters have to be scaled by the amount of decimation. For example, a 16 kHz signal may be decimated 4 times and the cutoff frequency of high-pass filtering at 4kHz has to reduced to 1kHz.

The software also enable us to visualize the marked speech data for verification and modification. Non-silence is shown as a ribbon on the top of the view window. Since nasal and air-burst do not occur simultaneously, they are shown as different color ribbons at the horizontal level in the view window. The pitch

epoches are displayed as vertical lines. Due to decimation, most pitch epoches are not displayed (Figure 15). They will appear again when a segment is magnified (Figure 16).

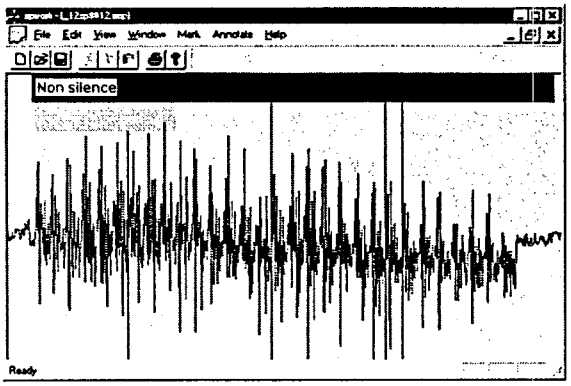


Figure 15: Visualization of the marked speech data. The dark ribbon shows the non-silence component. The light band shows the nasal segment. Due to decimation, most of the pitch marks are not displayed.

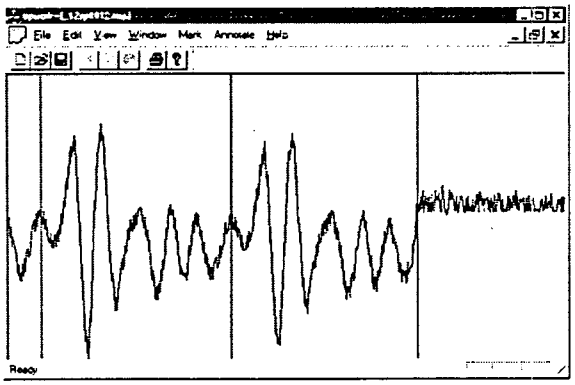


Figure 16: A magnified region of the marked speech data showing the location of the identified pitch epoches which were absent in Figure 15.

6. Discussion

We have described how 4-channels of speech data are recorded and transferred to the computer. We demonstrated that marked speech data provide important information for both annotation and speech analysis. Our marking scheme is space efficient and it is compatible with other speech processing software without regard to marking. We have also described how to post-process the 4-channels of data to obtain the marking information. Although the marking process can be completely automatic, human checking is still necessary for full correctness. Many decisions are based on setting an appropriate threshold, which can be determined by the k-means algorithm.

Acknowledgment

This research is supported by CERG PolyU 757/95H. We thank Mark Huckvale of the Department of Phonetics and Linguistics, University College of London for providing their speech filing system. We are grateful to the City University of Hong Kong for providing the aneohoic chamber and recording equipment. Finally, we thank Guo and Liu for providing the PH corpus.

References

1. Chan, D.S.F. and A.J. Fourcin (1994) "Automatic Annotation using Multi-Sensor Data", *Proceedings of ESCA Eurospeech '94*, Germany, pp. 187-190.
2. Zee, Y.Y. (1991) "Chinese (Hong Kong Cantonese)", *Journal of the International Phonetic Association*, 21(1).
3. Shih, C. and B. Ao (1994) "Duration study for the AT&T Mandarin text-to-speech system", *Conference Proceedings of the Second ESCA/IEEE Workshop on Speech Synthesis*, Mohonk, pp. 29-32.
4. Guo, J. and H.C. Lam (1992) "PH: a Chinese corpus for pinyin-hanzi transcription", *Technical Report TR93-112-0*, Institute of Systems Sciences, National University of Singapore.
5. O'Shaughnessy, D. (1987) *Speech communication: human and machine*, Reading, Mass.,Addison-Wesley.