SUVING: AUTOMATIC SILENCE /UNVOICED/VOICED CLASSIFICATION OF SPEECH

Mark Greenwood, Andrew Kinghorn

Department of Computer Science, University of Sheffield
Regent Court, 211Portobello St, Sheffield S14DP, UK
{u7mag, u7awrk}@dcs.shef.ac.uk

ABSTRACT

This paper is concerned with labelling sections of speech samples based on whether they are silence, voiced or unvoiced speech. The labelling is done using calculations over the speech samples; zero crossing and short-term energy functions. These functions complement each other and as such can be used more accurately together to label the parts of speech. The results of applying these functions to ten speech samples are compared to the result of the same samples having been manually labelled, to produce a percentage accuracy for each speech file. This study found that the average percentage accuracy of the algorithms implemented, over all ten-speech samples was about 65%, and concludes that the accuracy could be slightly improved through the use of a more accurate windowing function.

1. INTRODUCTION

A classification of speech into voiced or unvoiced sounds provides a useful basis for subsequent processing, for example fundamental frequency estimation, formant extraction or syllable marking. A three-way classification into silence/unvoiced/voiced (hence the title, SUVing) extends the possible range of further processing to tasks such as stop consonant identification and endpoint detection for isolated utterances. Strictly, speech sounds such as voiced fricatives (e.g. "z") can have characteristics of both voiced and unvoiced sources simultaneously which makes classification more difficult, so for the purposes of this study we will assume that a 3-way classification is sufficient for the needs of any further processing.

We approach this problem of SUVing from two different directions: zero crossings and short-term energy. These two methods compliment each other well, and prevent us from having to rely heavily on one single method to label the different parts of speech.

We do not provide an interface for viewing the results of this study instead the results of the SUVing are stored in a format that allows them to be loaded into the *slt* tool which is included as part of [2].

2. SUVING USING ZERO CROSSINGS

The notion of zero-crossings is defined to be:

> *"the number of times in a sound sample that the amplitude of the sound wave changes sign"*

For a 10ms sample of *clean* speech, the zero-crossing rate is approximately 12 for voiced speech and 50 for un-voiced speech [1]. For clean speech the zero-crossing rate should also be useful for detecting regions of silence, as the zero-crossing rate should be zero.

Unfortunately, very few sound samples are recordings of perfectly clean speech. This means that often there is some level of background noise, that interferes with the speech, meaning that silent regions actually have quite a high zero-crossings rate as the signal changes from just one side of zero amplitude to the other and back again. For this reason a tolerance threshold is included in the function that calculates zero-crossings to try and alleviate this problem. The thresholds work by removing any zero-crossings, which do not both start and end a certain amount from the zero value. In this study we have used a threshold of 0.001. This means that any zero-crossings that start and end in the range of x, where $-0.001 < x < 0.001$, are not included in the total number of zero-crossings for that window. This enables us to filter out most of the zero-crossings that occur during silent regions of the sample due only to background noise.

In this study to calculate zero-crossings we used a 10ms non-overlapping rectangular window. This does not produce such good zero-crossing results as an overlapping hamming window would, but since we are not interested in the fine details, this method works well when used to SUV a speech sample.

3. SUVING USING SHORT-TERM ENERGY

Short-term energy allows us to calculate the amount of energy in a sound at a specific instance in time, and is defined in Equation 3-1.

$$E_n = \sum_{m=n-N+1}^{n} (x(m)w(n-m))^2$$

Equation 3-1: Short-Term Energy (w is the window, n is the sample that the window is centered on, and N is the window size [1]).

Unfortunately, unlike zero-crossings there are no standard values of short-term energy for specific window sizes. Short-term energy is purely dependent upon the energy in the signal, which changes depending on how the sound was recorded. For example, if a person is recorded saying the same phrase twice, one while whispering and once while shouting, then the short-term energy values will be vastly different, although the zero-crossing values should be roughly the same. This means that you have to inspect the recorded speech files to determine at what level to make the distinction between voiced and unvoiced speech.

There is one thing that is standard though, and that is that short-term energy is higher for voiced than un-voiced speech, and should also be zero for silent regions in clean recording of clean speech.

In a similar way to zero-crossings we calculate the short-term energy using a 10ms non-overlapping rectangular window. This, again, is not as accurate as using an overlapping hamming window but it is adequate for the SUV labelling of speech.

4. SUVING USING BOTH METHODS

From the descriptions of the methods that are used to SUV label a speech signal, in this study, it should be clear that the two methods compliment each other well.

For voiced speech short-term energy is high and zero-crossings are low, and for un-voiced speech the opposite is true, short-term energy is low and zero-crossings are high. This can be seen clearly in Figure 4-1

In a perfect world, all speech samples would be clean and then Table 4-1 could be used to classify the speech as silence, un-voiced or voiced.

Zero-Crossings	Short-Term Energy	Label
approx. 12	High	Voiced
approx. 50	Low	Un-Voiced
0	0	Silence

Table 4-1: Perfect world labelling scheme.

Unfortunately sampled speech is never perfectly clean, usually containing some level of background noise, and so the labelling scheme in Table 4-2 is used in this study to label the speech samples.

Another problem, apart from background noise is that it is often difficult to detect silent regions of speech samples due to the fact that the short-term energy for a breath can quite easily be confused with the short-term energy of a fricative sound [3].

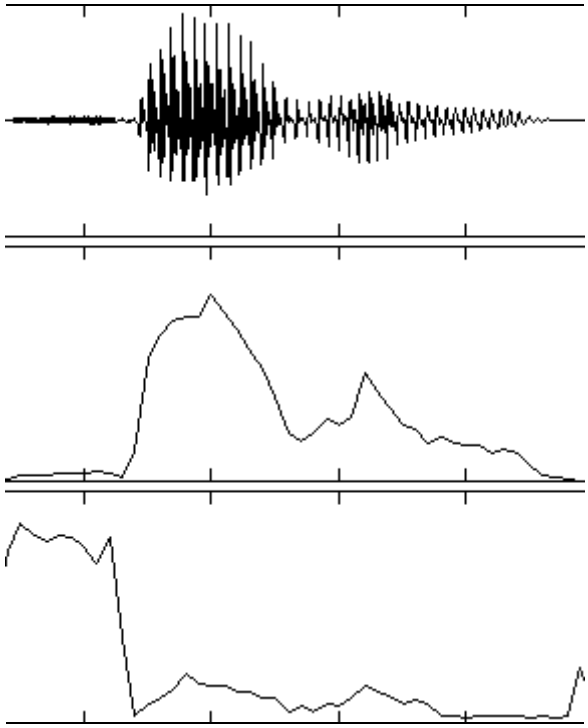

Figure 4-1: Showing the waveform, short-term energy, and zero crossings for the word "seven". Produced using the *timedom* tool which is included as part of [2].

Zero-Crossings	Short-Term Energy	Label
approx 0	approx 0	Silence
High	Low	Un-voiced
Low	High	Voiced
approx. 0	High	Voiced
High	High	Voiced
Low	Low	Voiced
approx. 0	Low	Un-voiced
Low	approx. 0	Silence
High	approx. 0	?

Table 4-2: Real world labelling scheme.

The one obvious anomaly in Table 4-2 is the last labelling scheme that labels a window as '?'. This is because it is impossible to get this arrangement of zero-crossings and short-term energy, in a speech signal. We decided that it made more sense to label this anomaly as a '?' than to try and fudge the zero-crossings and short-term energy values to make them fit the criteria of a different label. We also realised that it would be a useful debugging aid to know that a window could not be successfully labelled by the labelling function.

5. RESULTS

Two people independent of each other, manually labelled the ten sound files with silence, unvoiced or voiced. The results of the manual labelling were then

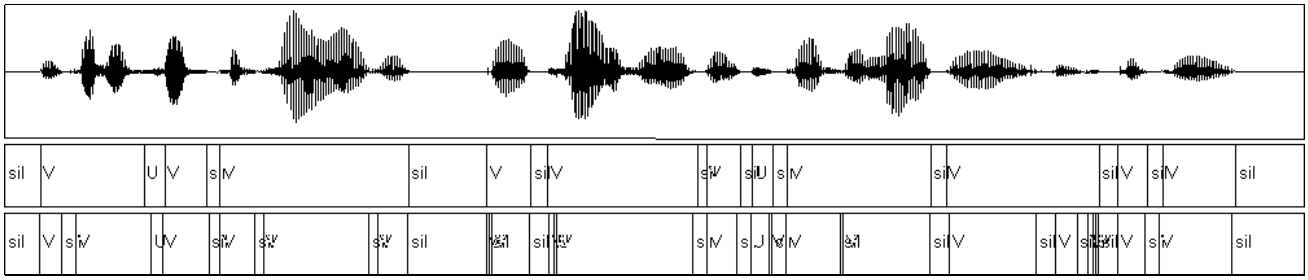

Figure 5-1: The waveform, manual transcription, and automatic transcription for the file eg1.au, where the percentage correctness is 88%. Produced using the *slt* tool which is included as part of [2].

compared and discussed so that the most accurate manual SUVing results could be obtained. The technique that was used to label the samples was to inspect the spectrogram and the speech waveform to identify silence and speech. Once areas of silence and non-silence are established, the non-silence parts of speech are labelled as voiced or unvoiced.

Voiced speech can be distinguished from unvoiced speech as it has a much greater amplitude displacement, when the speech is viewed as a waveform (this can also be seen in Figure 4-1). Another way of telling voiced from unvoiced speech is through examining the spectrogram. In a spectrogram areas of voiced speech have obvious structure (actually the formants), whereas unvoiced speech lacks any real speech.

As has already been stated, the automatic translations are then produced using a comparison of zero-crossings and short-term energy for 10ms non-overlapping rectangular windows of the speech signal.

These two sets of transcriptions are difficult to compare by eye, as can be seen in Figure 5-1, so a method of comparison is needed. The method employed in this study is to calculate the percentage of windows in the two transcriptions that match. The results of this comparison, for all ten speech samples, can be seen in Table 5-1.

Speech File	Percentage Correctness
eg1.au	88
eg2.au	65
eg3.au	66
eg4.au	66
eg5.au	84
eg6.au	73
eg7.au	47
eg8.au	46
eg9.au	50
eg10.au	66

Table 5-1: Comparison of automatic and manual transcriptions of the ten speech samples.

As can be clearly seen from Table 5-1, the percentage correctness ranges from a maximum of 88% down to 46%, which gives an average percentage correctness of 65%.

6. DISCUSSION

To label the sample a zero-crossing function and short-term energy function were applied to the sample. These functions are complementary, for example zero-crossings are high when the speech is unvoiced, but short-term energy is low at this point, the vice versa is true for voiced speech, and both are approximately zero for silence. The functions use the frequency of the sample and a window size of 10ms to split the sample into sections and produce the results. Cut off values are used to identify if a particular window of a sample is of the type 'sil', 'U' or 'V' the label given is simple if both the functions produce values which suggest the same label.

The problem is that there are cases where the functions produce differing values these problems are partially caused by background noise. This causes the cut off for silence to be raised, as it may not be quite zero due to noise being interpreted as speech by the functions, under clean speech both zero-crossings and short-term energy should be zero for silent regions.

The way different people talk, such as volume and speed also causes problems identifying endpoints of words and voiced/unvoiced speech. As the samples are at different volumes the cut off values would need to be changed for each sample making accuracy hard from sample to sample. It would also be a very time consuming activity having to tweak the cut off values for each sample, which would also defeat the object of this study.

We decided that if the results of the functions didn't match then if the short-term energy implies voiced speech and the zero crossings implies silence, then the result should be voiced speech. This is because zero-crossings have a low value for silence and voiced speech, therefore there is more chance of an error between these values, but the short-term energy is only ever high when voiced speech occurs.

In a similar way if zero-crossings imply an unvoiced speech sound and short-term energy implies silence then the speech is labelled as unvoiced, because short-term energy should be low for unvoiced speech.

In retrospect, these assumptions seem to have been valid assumptions to make, as the SUVing produced under these assumptions seem on the whole to be correct.

7. CONCLUSIONS

The parts of speech labelling produced using the algorithms outlined in this study are reasonably accurate for well recorded, fairly clean speech but are not nearly as accurate for quiet recordings of speech.

The accuracy, of the algorithms outlined in this study, could be improved in two ways. Firstly more time could be spent on tweaking the cut-off values used by the algorithms to label the different parts of speech. The problem with this, however, is that if the values are fine tuned for one speech sample it is unlikely that they will be as accurate on other speech samples.

The other possible way of increasing the accuracy of the algorithms would be to use an over-lapping hamming window, when calculating the zero-crossings and short-term energy. This would, however, mean that many more calculations were necessary for each speech file, which would drastically increase the time taken to label an entire speech file. If, however, the speed of SUVing is not an issue, then this method of improving the algorithms is preferred to fine tuning the cut-off values.

The Matlab code of the algorithms outlined in this paper, and the manual and automatic transcriptions of the speech samples, for use with the *slt* tool (provided as part of [2]), can be found on the WWW at:
http://www.dcs.shef.ac.uk/~u7mag/com325/

REFERENCES

[1] M. Cooke. *COM325: Computer Speech & Hearing* (Lecture Notes). Presented at the University of Sheffield, 1999

[2] M. Cooke, G. Brown, S. Wrigley, and D. Ellis. *Matlab Auditory Demos, version 2.0*. Available Dec 1999 from:
www.dcs.shef.ac.uk/~martin/MAD/docs/mad.htm

[3] B. Gold, and N. Morgan. *Speech and Audio Signal Processing*. John Wiley & Sons, 2000