Proceedings of The Institute of Acoustics

THE ACOUSTIC FLOW OF SPEECH

R.K. Moore, M.J. Tomlinson and S.W. Beet

Royal Signals and Radar Establishment, Malvern

INTRODUCTION

In most speech processing tasks, and in automatic speech recognition in particular, it is usual to represent speech pattern information in such a way as to emphasise those aspects of the signal which are considered to be important. For this reason, a speech signal is commonly subjected to a short-term analysis in order to make its spectral content more explicit. This, in turn, leads to a timeframe based representation of speech patterns, and to the familiar 'speech spectrogram' [1].

It is also well established that the transitional components of a speech signal are particularly relevant to its identity. However, in order to make transition information available to a speech processing algorithm, it is usually necessary to 'detect' some feature in a speech pattern (such as a formant, for example), and then to track its movement over time. Such techniques can be reasonably effective, but the usefulness of the resulting transition information depends directly on the accuracy and reliability of the feature detection process. In pitch tracking, for example, it is normally necessary to make provision for the occurrance of pitch detection errors such as pitch doubling.

It would be more useful if the transition information in a speech pattern could be made explicit without reliance on the detection of particular speech pattern features. What is needed is an enhancement of the 'spectral continuity' that is visually apparent in conventional speech spectrograms. One possibility is to 'filter' the three-dimensional time-frequency-amplitude pattern using a range of spectral 'operators', each tuned to a different direction of energy movement. However, this procedure has the effect of fragmenting the information (among the different operators) and the essential continuity and coherence of the pattern components may be obscured.

This paper presents an alternative approach which considers the detailed relationship between adjacent analysis timeframes. It is suggested that, rather than isolate a particular parameter (such as a spectral peak), it might be better to track the movement of the entire pattern (spectrum) from one time instant to the next. In other words, it is desirable to track all components of an analysis timeframe simultaneously, using the instantaneous movement of the complete pattern shape to reveal the required transition information. Since the complete pattern is used, the transition information obtained would be reasonably robust.

An analogous technique is already established in the field of image processing where it is the motion of objects within a scene which is of interest [2]. In this situation, two successive images are compared and a vector-field is obtained which describes the relative movements of each part of a picture. The resulting transition information is termed the 'optic flow' [3].

THE ACOUSTIC FLOW OF SPEECH

Figure 1 illustrates the 'optic flow' process; two successive dot pattern images are shown in figure 1a and figure 1b. Figure 1c shows the 'flow' describing the movement that has taken place and which reveals the shape of the object of interest.

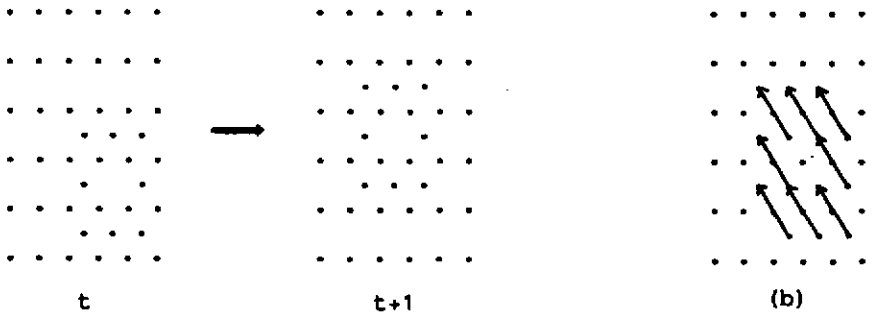

Figure 1: Demonstration of 'optic flow'

This paper, therefore, presents a new representation scheme for speech patterns which, because of the analogy with 'optic flow', is termed the 'acoustic flow' of speech.

COMPUTING THE 'FLOW'

In order to derive the 'optic flow', it is necessary to solve the 'correspondence problem' [4]. That is, each point in the image at time t has to be assigned to a corresponding point in the image at time t+1. The difficulty is that these correspondences must take into account the topology of the images; the image at time t+1 is assumed to be a well behaved distortion of the image at time t. Hence the assignment is not based on the best local correspondences, but on the overall result. This means that, unless the images are changing very slowly, an iterative 'relaxation' process has to be employed to compute the 'flow' and the optimum assignment is not guaranteed.

However, with speech data the problem can be much simpler because the patterns to be compared tend to be of lower dimensionality. Hence, to compute the 'acoustic flow' (between two spectra, for example), it is possible to use dynamic programming to determine the optimum correspondence. In the case of spectral data, the process is equivalent to 'dynamic frequency warping' since it results in a non-linear distortion of one speech spectrum into another. The 'flow' is derived from the 'frequency registration path' and, since a spectrum is involved, it can be more usefully called a 'spectral flow' pattern.

Figure 2 illustrates the process for a simple rising tone: figure 2a shows a conventional time-frequency-amplitude representation of the tone and figure 2b shows the non-linear frequency alignment between two adjacent analysis timeframes. It can be seen that the resulting correspondences reveal the

spectral shift that is present. Figure 2c shows these correspondences in terms of pairwise frequency assignments, and the resulting 'acoustic flow' pattern indicates the local upward shift in frequency. Figure 2d shows the complete 'spectral flow' picture for the original spectrogram; note how the upward movement of the tone is explicitly represented in the 'flow'.

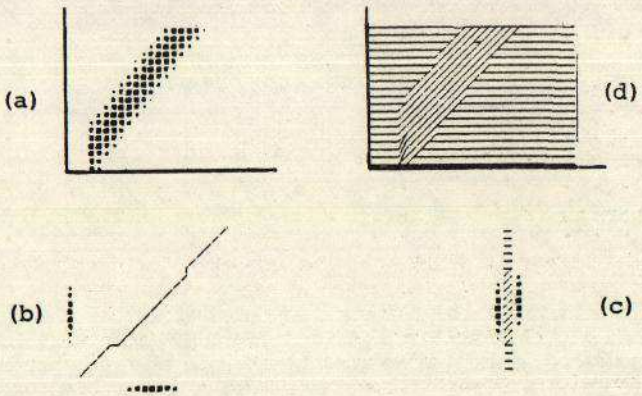

Figure 2: Demonstration of 'acoustic flow'.

THE 'ACOUSTIC FLOW' PATTERN OF SPEECH

Figures 3 and 4 show the 'spectral flow' and the more familiar time-frequency-amplitude representation for two different sentences taken from the RSRE speech data base [5] (speaker TB-male, session-308). The transition information is clearly made explicit by the 'flow' pattern.

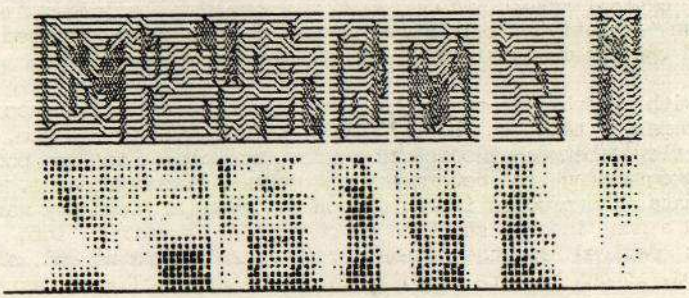

Figure 3: 'Flow' pattern for "you fought on the deck of a boat".

THE ACOUSTIC FLOW OF SPEECH

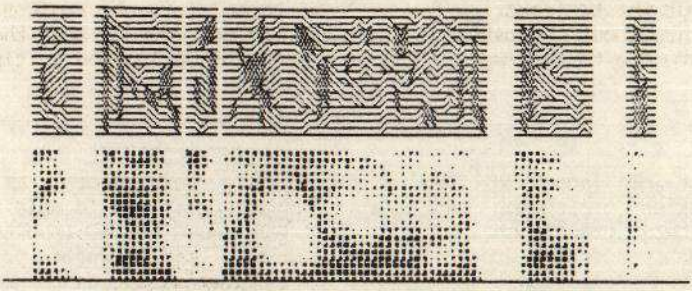

Figure 4: 'Flow' pattern for "we go to a yawl on the dock".

The examples in figures 3 and 4 are based on the output from a channel vocoder analyser [6]. However, the resolution of the vocoder is rather limited and, in particular, the transition information present in the original signal is not well preserved. It is therefore of considerable interest to determine the 'spectral flow' of data whose quality is more akin to that of a conventional speech spectrogram. Figure 5 illustrates such a higher resolution spectrogram of the word "zero" (RSRE speech data base, speaker CB-female, session-36, list-7, part-1) and its associated 'spectral flow'.

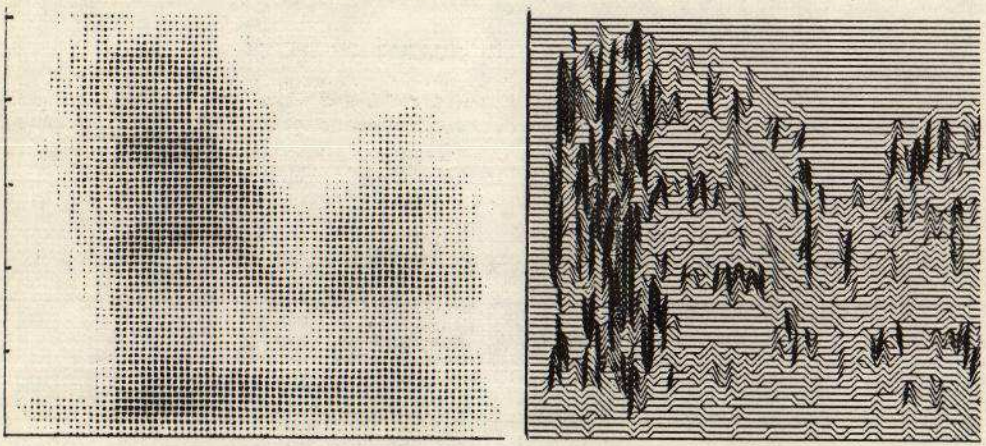

Figure 5: 'Spectral flow' pattern for the utterance "zero".

As expected, the 'spectral flow' for the higher resolution data is rather less uniform, but the trajectories for at least three formant regions are clearly visible throughout the pattern. The more violent movements at the beginning of the word are due to the rapidly changing spectrum during the initial fricative.

DISCUSSION

The concept of 'acoustic flow' is obviously very useful as an improved representation of speech patterns. In particular, it is able to effectively emphasise those transitional aspects which are known to be important cues for the speech perception process. 'Spectral flow' may therefore provide a better low level representation than previous attempts (such as 'sectorgraphs' [7], for example). Also, it would be interesting to investigate the relationship between 'acoustic flow' and other descriptively motivated techniques such as the 'speech sketch' [8].

'Spectral flow' also has properties which make it useful for a number of other tasks. All of the examples shown so far have used a wideband spectral analysis; if a narrowband analysis is performed, then the 'spectral flow' pattern tracks the pitch harmonic structure. Figure 6 illustrates the point for the word "no" carrying a fall-rise intonation contour.

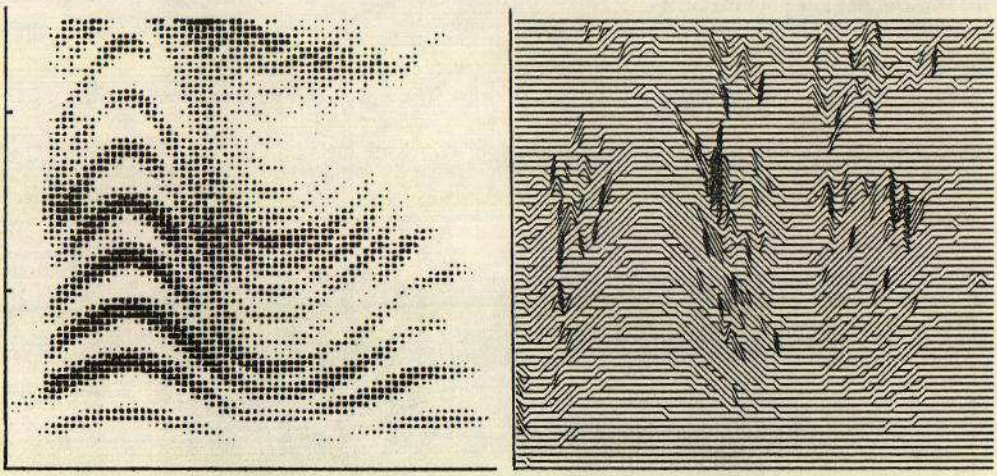

Figure 6: 'Spectral flow' for a narrowband analysis of "no".

Also, because 'spectral flow' is able to capture the transition information over an entire spectrum, it can be used to obtain an overall measure of spectral movement. For a narrowband analysis, this would yield an estimate of the pitch variations. For a wideband analysis, it would indicate regions of spectral stability, and this could have implications for automatic segmentation processes or for estimating the rate of speech production.

THE ACOUSTIC FLOW OF SPEECH

Another feature of 'spectral flow' is that it can be used to enhance the spectral comparison process (the distance measure) in automatic speech recognition. In this case two spectra would only be judged as being similar if the spectral shape and the spectral movement were in agreement. This means that two identical spectra would be judged as different if the 'flow' information did not agree.

Last, the importance of 'optic flow' is that it provides information which enables 'objects' in a visual image to be separated from each other and from the 'background'. In this situation the temporal separation caused by movement is similar to the spatial separation achieved by binocular vision. As a consquence, the acoustic analogy would imply that the signal separation that is possible with binaural processing could also be achieved using suitable differential 'flow' patterns on one acoustic channel. This means that a separate 'flow' is required for each different sound source, including foreground and background signals. (There appears to be some evidence for this type of processing in the human auditory system [9].)

CONCLUSION

This paper has introduced a new technique for processing speech patterns which makes important transition information explicit without recourse to the detection of potentially unreliable speech pattern features. The technique appears to have many uses in the general analysis of speech/acoustic signals, and is especially relevant to the area of automatic speech recognition.

Finally, all the examples presented so far in this paper show the 'flow' information separate from the normal spectrogram-like data. However, it is possible to modulate the 'flow' pattern using the spectral amplitudes thus combining the two patterns into a single representation. Figures 7 and 8 present this new time-frequency-amplitude-flow representation for the same examples shown in figures 5 and 6.

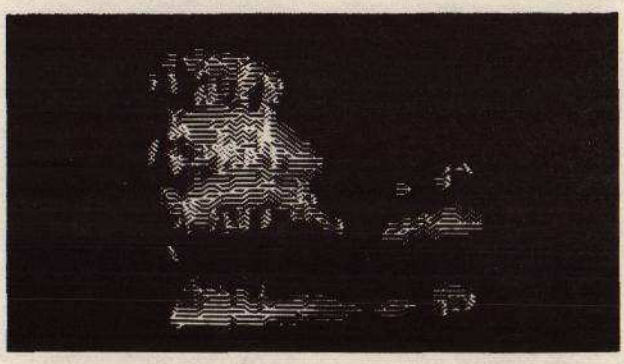

Figure 7: Greyscale 'flow' for the wideband analysis of "zero".

THE ACOUSTIC FLOW OF SPEECH

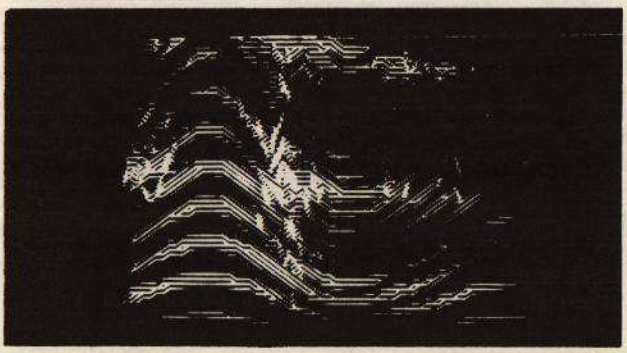

Figure 8: Greyscale 'flow' for the narrowband analysis of "no".

REFERENCES

[1] J.L. Flanagan. Speech Analysis, Synthesis and Perception. New York: Springer-Verlag, (1972).

[2] S. Ullman. 'Analysis of visual motion by biological and computer systems', IEEE Computer, August, 57-69, (1981).

[3] J.M. Prager and M.A. Arbib. 'Computing the Optic Flow: the MATCH algorithm and prediction', Computer Vision, Graphics and Image Processing, Vol.24, 271-304, (1983).

[4] R.O. Duda and P.E. Hart. Pattern Classification and Scene Analysis, Wiley, New York, (1973).

[5] M.J. Russell, R.K. Moore, M.J. Tomlinson and J.C.A. Deacon. 'RSRE speech data base recordings (1983): part II', RSRE Report No.84008, (1984).

[6] J.N. Holmes. 'The JSRU channel vocoder', IEE Proc. Communications, Radar and Signal Processing, Vol.127, Pt.F, 53-60, (1980).

[7] R. Linggard and D. Rankin. 'Sectorgraphs - A new way of looking at speech', Proc. Inst. Acoustics Autumn Conf., E1.1-E1.4, (1982).

[8] P.D. Green and A.R. Wood. 'Speech understanding by computer using descriptive techniques', Dept. Computing, North Staffs. Poly., Report No. CDTR 83-1, (1983).

[9] J.W. Hall and M. Haggard. 'Co-modulation - a principal for auditory pattern analysis in speech', 11th Int. Conf. Acoustics, Paris, (1983).

THE ACOUSTIC FLOW OF SPEECH

APPENDIX: DETAILS OF SPEECH ANALYSIS

The speech signal was high-frequency pre-emphasised and passed through a 6.4 kHz anti-aliasing filter then sampled at 19.2 kHz to an accuracy of 12 bits. The speech data was Hanning windowed to 13 ms every 6.5 ms, before a log frequency analysis by a 256 point DFT, giving a 75 Hz resolution. To remove pitch harmonics from the short-term spectra, cepstral smoothing was employed using a 2.1 ms rectangular window. Subsequent dynamic frequency warping was performed on the lower 65 channels giving a 0 to 5 kHz range. Each timeframe was subjected to an instantaneous a.g.c. action to maintain constant peak spectral amplitude.

As can be seen by from this rule the RIGHT-C and LEFT-C can be null. The transcription string may also be null. For example, when two identical consonants occur in a word they are pronounced as one phoneme. The rule for omitting the second *m* in *summer* is

m (m) =

A novel facility that makes for compact specification of the rules and improves processing efficiency, provides for the LEFT-C and RIGHT-C strings to contain sets of letters. Tests are made for exclusion from the set, represented for example by ¬[x,y,z] or membership of the set, represented for example by [a,b,c,d] ; as in one of the *e* rules:-

[p,s,v] (e) [t,p] = ε

This rule would be successful for *reset* but not for *lever*. It should be noted that although this rule would apply to *compete* there are earlier rules which would also apply, thus transcribing the second syllable of it before the above rule is encountered. Therefore the ordering of the rules is of critical importance.

A unique feature of this rule schema is the ability to match the LEFT-C string with the phonemic transcription already produced. For example, when *ed* occurs at the end of a word it is pronounced /t/ when preceded by a voiceless sound. The appropriate rule is :

[/p,k,f,θ,s,ʃ/] (ed) # = t

(The # is used as a symbol for end-of-word.)

Potential linking-r is indicated by * in the **transcription** string. After processing the text, the string of phonemes is scanned for * and is replaced by /r/ when the following sound is a vowel.

Functions, shown in upper case in the example, can also be specified in the RIGHT-C and LEFT-C strings to increase the generality of the rules, e.g.

C0M	Find zero or more consonants.
C1	Find one consonant.
C1M	Find one or more consonants.
V1	Find one vowel.
V1M	Find one or more vowels.
VF1	Find a front vowel (i.e. e,i,y).

The Allophonic Transcription and Assimilation Rules

The rules for allophone production and assimilation are similar in format to the phonemic transcription rules, but contain functions that relate to features of phonemes rather than graphemes. The **object** and **transcription** strings may also contain functions. One sound may belong to several classes of features: for example *t* is voiceless, plosive, aveolar, obstruant, and consonantal. Each phoneme in the **object** is processed by all the rules. Not all phonemes have allophone or assimilation rules.

The following function mnemonics are used only in the allophone and assimilation rules.

FRIC-VOX	A voiceless fricative - f,θ,s,ʃ,h
NASAL	A nasal - n,m,ŋ
NASAL@X	A nasal at X, where X is the place of articulation
STOP	A stop - p,t,k,b,d,g
STOP+VOX	A voiced stop - b,d,g
STOP-VOX	A voiceless stop - p,t,k
STOP@X	A stop at X, where X is the place of articulation.
VOWEL	A vowel - ɑ,æ,ɛ,ə,i,ɪ,ɒ,ɔ,u,ʊ,ʌ,ɔɪ,aʊ,əʊ,eə,ʊə,ɪə,aɪ,eɪ,ɔə,ɜ

The following is a list of allophone properties or attributes used in the **transcription** string of the allophone rules.

ASPIRATED	NASALISED
DENTALISED	PALATALISED
DEVOICED	PART-ASPIRATED
GLOTTALISED	PART-VOICED
LABIALISED	UNEXPLODED
LENGTH(1 to 6)	VELARISED

Example of Allophone Rules

At the level of the phoneme-to-allophone rules, an attempt has been made to encode linguistically-motivated rules for the modification of phonemes in context. These rules are a development of the articulatory-based distinctive features of Ladefoged [1], and provide a modelling of the articulatory dimension for what is essentially a terminal-analogue formant synthesiser.

The following is an example of some of the allophone rules.

	RULE		OUTPUT
5	(VOWEL)	FRIC-VOX	= LENGTH,2
6	(VOWEL)	STOP-VOX	= LENGTH,1
7	(STOP)	|[—],STOP	= UNEXPLODED

Notes

1. The symbol ' — ' is used to represent a syllable boundary or a word boundary (see rule 7).
2. The symbol ' | ' is used to indicate that the following set is optional.
3. Square brackets (' [] ') are used to delimit a set, of which only one member may occur at the given position.

Example of Assimilation Rules

The following is an example of some of the assimilation rules. The set of rules has been partly based on the work of Wells and Colson [7].

	RULE		OUTPUT
1	(n)	[STOP@X,NASAL@X]	= NASAL@X
2	(STOP)	STOP@X,VOWEL	= STOP@X
3	(STOP)	NASAL@X	= STOP@X