STOP CONSONANT CLASSIFICATION USING WAVELET PACKET TRANSFORMS AND A NEURAL NETWORK

HANS L. CYCON AND W. LI

STEPHEN A. ZAHORIAN

(CA-- INSERT AFFILICATIONS-- ODU, AND SHCOOL IN ?BERLIN--ASK STEFAN)

ABSTRACT

A wavelet packet transform is described to compute N spectral/temporal features for the 6 English stop consonants /b,p,d,t,g,k/. These features were used by a Binary Pair Partitioned neural network for speaker-independent classification of the stop consonants. The wavelet packet transform is generated by a pair of quadratic mirror filters which decompose the signal into a series of subbands ("frequency slots") by repeated convolution and decimation. Choosing a complete set of subbands and dividing each subband into a number of "time slots" defines a decomposition of the time-frequency plane into N phase cells. The N mean square values (energy values) in these phase cells provide the N features for the neural network. The number of features N was varied between 18 and 200. One advantage of this type of wavelet analysis is that it is much faster than the conventional FFT based methods which is of particular interest for real time applications. In addition there is the potential to exploit non-uniform time/frequency resolution. Experimental results obtained with the stops extracted from the TIMIT data base will be presented in the paper.

INTRODUCTION

Wavelet transforms

Wavelets, or more specifically wavelet packets, show promise for signal processing, particularly for non stationary signals. Primary advantages include flexibility in adapting to the class of signals and the low computational complexity, which is especially important for real-time applications. In this paper, we describe the use of wavelet packet transformations (WPT) for feature computations for the automatic classification of the stop consonants b,p,d,t,g,k, using neural networks for pattern recognition. A wavelet packet Ψ is a square integrable function which is well localized in both time and frequency. (See [Wi94], [Co94] and [Co92] for a more detailed discussion.) A wavelet packet coefficient of a signal x is the inner product of x with a wavelet packet function generated by a mother wavelet. A wavelet packet transform is a transform of x into a collection of wavelet packet coefficients. A discrete wavelet packet transform is generated by a pair {h,g} of quadrature mirror filters (QMF`s). We consider here only finite impulse response (FIR) filters, so both h and g are represented by finite sequences. The pair {h,g} consists of a low pass filter h and a high pass filter g which partition the signal bandwidth into two equally spaced intervals. The coefficients of g and h are related by $g_k = (-1)^k h_{1-k}$ Define the associated convolution-decimation operators by

$$Hx(t) = \sum_{k=1}^{n} h_k x(2t-k) \quad \text{and} \quad Gx(t) = \sum_{k=1}^{n} g_k x(2t-k)\ ,\ t \in R.$$

These filters are related to a scaling function Φ and a mother wavelet Ψ by the equations $\Psi = G\Phi$ and $\Phi = H\Phi$. A one dimensional discrete signal $x = (x_1, x_2, ..., x_N)$, of finite length N, can be filtered by convolution of the filter coefficients with the signal values and downsampling by two:

$$Hx_i = \sum_{k=1}^{n} h_k x_{2i-k}\ ,\ i = 1,.....,N/2 \quad \text{(low pass filltering)},$$

$$Gx_i = \sum_{k=1}^{n} g_k x_{2i-k}\ ,\quad i = 1,.....,N/2 \quad \text{(high pass filtering)}.$$

A fundamental property of the wavelet transformation is that the low pass and high pass filtering, each followed by decimation, does not lose information. In fact, it can be shown that the original signal can be exactly reconstructed from Hx and Gx. If concatenated, the pair {Hx,Gx} has the same length N as the original signal x. These are referred to as the wavelet packet coefficients of the first transformation level. i.e., Hx is the first level low pass subband and Gx the first level high pass subband. Repeated application of H and G leads to a collection of subbands. Since the length of the subbands is always halved as the level increases by one the maximum number of iterations is $\log_2$ (N). This collection of subbands can be represented by a tree-structured rectangle of wavelet packet coefficients:

level 1	Gx				Hx			
level 2	GGx		HGx		GHx		HHX	
level 3	GGGx	HGGx	GHGx	HHGx	GGHx	HGHx	GHHx	HHHx
level...	...	...	...	...	...	...	...	...
level logN	...	...	...	...	...	...	...	...

Each level has N coefficients and contains all the information of the signal x, in the sense that x can be exactly reconstructed from each level. Note also that each of the subbands spans the entire time interval of the signal x. Since with each increasing iteration, frequency ranges are halved and the number of subbands is doubled, the lower levels have more detailed frequency resolution but less time resolution and vice versa. If concatenated ,any non-overlapping complete set of subbands, including those obtained by mixing levels, completely represents the signal x. Each of these corresponds to one possible wavelet packet transformation (WPT) generated by a corresponding filter bank. Therefore we have huge variety of different WPT`s and the freedom to choose one having the appropriate frequency and time resolution for the particular class of signals to be analyzed. Note that the usual fast wavelet transform (FWT), which is generated by iteratively subdividing the low pass subbands and selecting the high pass subband, is just one special case of a wavelet packet transformation

EXTRACTING FEATURES

By subdividing frequency subbands in time, “time slots” can be created within the “frequency slots” In a time-frequency representation (i.e. the time-frequency plane) this corresponds to a “tiling” of the time-frequency plane into a complete set of disjoint “phase” cells. Figure 1 shows a tiling of the time-frequency plane into 18 phase cells using 6 frequency subbands over the range of 0 - 8kHz. Figure 2 shows a time-frequency “phase plot” obtained with a WPT for the stop /d/.

(CA insert figures here whereever there is space. Note that figure 1 is at the end of the file. Figure 2 (which should be about the same size as figure 1) will be pasted in. Note that the next section should not be in large bold letters.)

DEPENDING ON THE CLASS OF SIGNALS TO BE CLASSIFIED WE CAN CHOOSE A TILING WITH TIME AND FREQUENCY RESOLUTION DEPENDENT ON POSITION IN THE TIME-FREQUENCY PLANE, AND THE TYPE OF INFORMATION LOCATED IN EACH REGION. FOR THE CASE OF STOP CONSONANTS, HIGH TIME RESOLUTION IS DESIRED AT THE BEGINNING OF THE BURST. HIGH FREQUENCY RESOLUTION IS DESIRED FOR LOW FREQUENCIES, PARTICULARLY IN THE TRANSITION REGION TO THE FOLLOWING VOWEL. USING THE TILING IN THE TIME-FREQUENCY PLANE AS DESCRIBED ABOVE, THE MEAN SQUARE VALUES (I.E. THE ENERGY) OF THE COEFFICIENTS IN EACH OF THE PHASE CELLS CHARACTERIZE THE SIGNAL IN A TIME-FREQUENCY SPACE AND ARE USED AS FEATURES TO CLASSIFY THE SIGNAL.

THE 18 PHASE CELL TILING OF THE TIME-FREQUENCY PLANE IN FIGURE 1 THUS CORRESPONDS TO 18 FEATURES. NOTE FOR EACH SUBBAND THE NUMBER OF FEATURES IS EQUAL TO THE NUMBER OF TIME SLOTS IN THAT SUBBAND. MORE TIME RESOLUTION CAN BE OBTAINED BY INCREASING THE NUMBER OF TIME SLOTS IN EACH SUBBAND. SIMILARLY, MORE REQUENCY RESOLUTION CAN BE OBTAINED USING MORE SUBBANDS. EXPERIMENTS REPORTED IN THIS PAPER USED EITHER 18 FEATURES (AS DEPICTED IN FIGURE 1), 36 FEATURES (DOUBLING THE TIME SLOTS OF FIGURE 1), OR 83 FEATURES (MORE TIME AND FREQUENCY SLOTS).

BINARY PAIRED PARTITIONED NEURAL NETWORK CLASSIFIERS

Several feature sets, each calculated using variations of the phase space decomposition described above, were used by a Binary-Paired Partitioned neural network (BPP) classifier. The BPP classifier (Zahorian, et al., 1993) uses a single "small" network for each pair of categories which must be discriminated, and then combines these decisions to make final decisions. In previous work, this approach has yielded higher classification rates than for a single large network. For the case of 6 stops, a total of 15 2-way networks are required. For all experiments reported in this paper, a two layer feedforward network with 10 hidden nodes (sigmoidal nonlinearities) and one output node was used. Each network was trained for a number of iterations, between 50 000 and 300 000 using an initial learning rate of .45 which was reduced by a factor of .97 every 5000 iterations. Several experimental "runs" were made with this classifier. Recognition rates were computed as the percentage of tokens correctly classified, for both training and test data. Both training and test results are given in the following tables. Test results are a more meaningful measure of the ability of the classifier to generalize to new data. However, training results give a measure of the upper limit of performance, if an infinite size set of training data were available.

EXPERIMENTS AND RESULTS

Isolated stops

The data for the first set of experiments consisted of 70 tokens for each of the six stops. All tokens were produced in isolation, followed by an /iy/, by 7 male speakers. Each speaker produced each token 10 times. 35 tokens of each stop were used for training, and the remaining 35 were used for testing. Each stop was sampled at 11.025 kHz and stored in a file containing a 75 ms segment beginning 5 ms before the (automatically detected) burst. We used a bandwidth of 0 -8 kHz and a time window of 1024 sample values for the actual WPT.

Experiment 1

The objective of this series of classification experiments was to find the best filter pair out of 4. “Best“ is defined as giving the highest automatic recognition results which means it is best adapted to the class of signals investigated. All the filters used here are from a software package provided with [Wi94]. We compared results using orthogonal quadrature filters like Beylkin with 18 coefficients (B18), Coifmann with 30 coefficients (C30), Daubechies with 20 coefficients (D20) and Vaidyanathan with 24 coefficients (V24). For each class of filters, we chose only those with the highest number of coefficients since they are best suited for audio signals. All results, summarized in Table 4, were obtained using 36 features , and 100 000 training iterations for the neural networks. The best filter pair (V24) resulted in 89.5% recognition rate on the test data, considerably better than for the other filter pairs. This is a filter pair designed by Vaidyanathan and Huong especially for audio applications . Therefore this filter pair was used in all of the other experiments reported in this paper. Note, however, that the very high results on training data (typically 100%) indicate that there was really not enough training data.

TABLE 4: recognition rates for training and test data for different filter pairs

Filter	Train	Test
V24	100%	89.5%

D20	100%	84.3%
B18	99.5%	84.3%
C30	100%	84.3%

Experiment 2

This series of measurements were conducted to optimize the number of features and the number of iterations for the training data. Not all of the possible parameter combinations were tested due to the limited time available. The absolute best results were obtained with 83 features (as listed in table 3) and 200000 iterations for network training. Note that further increasing the number of iterations did not improve the recognition rate due to over training effects. A sample test with 36 features and 300 000 iterations decreased to a recognition rate of 70.9% on test data.

TABLE 5: recognition rates for different # features and # iterations

# IT	50.000		100.000		200.000	
	Train	Test	Train	Test	Train	Test
# Feat						
18	100%	83.3%	100%	88.6%	--	--
36	91%	74%	100%	89.5%	--	--
83	--	--	100%	90.0%	100%	93.3%

The run time of the WPT`s are considerably shorter than that of a comparable DCT method (Zahorian et al. 1993) For the 83 feature WPT classification the run time was 1:36 hours on a 66 Mhz Pentium, whereas a 90 feature DCT classification required 3:40 hours on the same machine. Subtracting approximately 1 hour for training the neural network, the WPT method is approximately 4 times faster than the DCT method.

Stops from Continuous Speech

The second group of experiments was done on a much more varied set of data, taken from the DARPA TIMIT acoustic-phonetic continuous speech database with 420 talkers from various dialect regions and 4200 sentences [DP90]. A total of 9512 training samples were used and 1019 tokens were used for testing. Since the stop consonants were selected from continuous speech samples, it was not always possible to detect a well defined on-set of the signal. Therefore the signals "main information content" could be shifted arbitrarily, which makes it much more difficult to extract a robust set of time-frequency features. For all experiments reported in this section, the data was synchronized to begin relative to the labeled start of the stop segment.

Experiment 3

The speech frame consisted (with one exception) of 1024 sample values (i.e. a 64ms signal interval). All results are based on the V24 filters with 200,000 neural network training iterations.

TABLE 6: recognition rates for different # features

#Features	Train	Test	Remarks
18	53.7%	52.35	# time slots s. table 1
36	65.7%	61.1%	# time slots s. table 2
83	74.9%	69.9%	# time slots s. table 3
99	78.9%	71.4%	
162	82.5%	71.1%	double feature length =2048

The relatively poor performance may be related to the difficulty of determining a well defined on-set. In addition, the stops from continuous speech simply exhibit a great deal more acoustic variability than due stops spoken in clearly spoken isolated words.

CONCLUSIONS

In this study we introduced a new method for a automatic speaker independent classification of stop consonants using a wavelet packet transformation. Our experiments indicate that for isolated stops the six stop consonants can be automatically classified with over 93% accuracy based on extracting 83 features out of a 75 ms signal interval beginning with the release of the burst. For stops extracted from continuous speech samples we achieve an accuracy only about 71% using 99 features out of a 83 ms signal interval. These relatively poor results are still comparable with a method using formant trajectories [NZ91]. However, they are not yet as good as results based on DCT time/frequency features. Nevertheless, we believe that with further optimization, the WPT methods can be greatly improved.

The primary advantage of the method introduced here is flexibility in that time and frequency resolution can be chosen arbitrarily to adapt to the class of signals to be classified. The low complexity compared to conventional DCT methods also makes it promising for real-time applications.

ACKNOWLEDGMENTS

One of us (H.L.C.) would like to thank C. A. Norton and K. Khareti for writing parts of the code used in this study.

REFERENCES

[Co92] R. R. Coifmann, Y. Meyer, S. Quake, M. L. Wickerhauser, "Signal Processing and Compression with Wave Packets," *Proceedings of the International Conference on Wavelets and Applications*, 1992, Editions Frontieres.

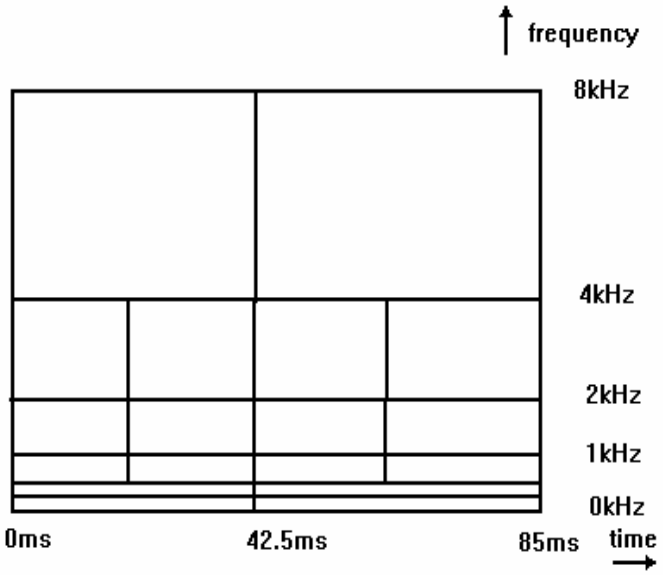
[Co94] R. R. Coifmann, M. L. Wickerhauser, "Best-Adapted Wave Packet Bases," Preprint 1994.

[DP90] DARPA TIMIT Acoustic Phonetic continuous speech corpus NIST # PB91-100354), October 1990.

[NZ91] Z.B, Nossair, S. A. Zahorian, "Dynamic spectral shape features as acoustic correlates for initial stop consonants," *J.Acoust.Soc.Am*.89 (1991)

[Wi94] M. L. Wickerhauser, "Adapted Wavelet Analysis from Theory to Software," AK Peters, LTD. Wellesey 1994.

[Z93] S. A. Zahorian, Z. B. Nossair, C. A. Norton, III, "A Partitioned Neural Network Approach For Vowel Classification Using Smoothed Time/Frequency Features," *EUROSPEECH-93*, pp. II: 1225-1228



Tiling of the time-frequency plane into
18 phase cells using 6 subbands

Figure 1