HIGH QUALITY SPEECH EXPANSION, COMPRESSION, AND NOISE FILTERING USING THE SOLA METHOD OF TIME SCALE MODIFICATION

James L. Wayman
Pebble Beach, CA
93953

R. Eric Reinke
Naval Underwater Systems Center
New London, CT 06320

Dennis L. Wilson
Ford Aerospace Corp.
San Jose, CA 95134

Abstract — The synchronized-overlap-add (SOLA) method of time scale modification is a computationally simple, time-domain technique that can be used for: 1) speech expansion, providing improved intelligibility for difficult speech segments; 2) speech compression/re-expansion, allowing low bit-rate transmission and storage; and 3) white-noise filtering, giving significant improvement in signal-to-noise ratio. In this paper, we discuss these three applications of SOLA and the proper system parameters required for each. The paper is accompanied by a demonstration tape.

I. INTRODUCTION

The synchronized-overlap-add (SOLA) method of time scale modification (TSM), was proposed originally by Roucos and Wilgus [1] and investigated further by Makhoul and El-Jaroudi [2] and Wayman and Wilson [3]. The latter researchers proposed a computational improvement for use in compression/re-expansion applications of SOLA and showed how the method produced an improvement in the signal-to-noise ratio of the output speech. In more recently completed work, we have investigated the application of SOLA to speech expansion without pre-compression, for the purpose of improving intelligibility, and have demonstrated the pitch and formant preserving properties of SOLA in this application, when parameters are correctly chosen. This paper will provide an overview of all of these findings and will be accompanied by a tape recording demonstrating expansion, compression, compression/re-expansion, and noise filtering.

II. DESCRIPTION OF SOLA

The SOLA method is an entirely time domain technique, requiring no frequency domain calculations or phase unwrapping. Of all time scale modification methods proposed, SOLA appears to be the simplest computationally, and therefore, most appropriate for real-time applications. The method begins by partitioning the input time domain data into overlapping frames of length N. The amount of new data per input frame is S_a, and thus, the overlap is N-S_a. Compression can be achieved by decreasing the amount of overlap to N-S_s (where $S_s < S_a$), averaging the overlapping samples, then outputting the resultant data stream. For expansion, the process is the same, but $S_a > S_s$. The ratio of S_s/S_a is the modification factor, α,. which for expansion is greater than 1 and for compression is less than 1.

Such a static method of compression (or expansion), as described above, results in very poor quality output. The SOLA method varies the degree of overlapping such that the amount of new data per output frame averages S_s, but is allowed to vary by an amount k_m at any particular overlapping, where *m* is the frame number and $-N/2 \leq k_m \leq N/2$. The value of k_m is chosen such that the cross correlation between the new frame and the composite output vector is maximized. Thus k_m is the value of k for frame *m* that maximizes

$$R_m(k) = \frac{\sum_{j=0}^{L_m-1} y(mS_s+k+j)\, x(mS_a+j)}{\left[\sum_{j=0}^{L_m-1} y^2(mS_s+k+j) \sum_{j=0}^{L_m-1} x^2(mS_a+j)\right]^{1/2}} \tag{1}$$

where L_m is the length of the overlap between the new signal samples $x(mS_a + j)$ and the composite output vector y formed by averaging the previous overlapped vectors. Once k_m is found, the vector y is updated by each new input frame, x, using the formula

$$y(mS_s+k_m+j) = (1-f(j))\, y(mS_s+k_m+j) + f(j)\, x(mS_a+j) \tag{2}$$

$$\text{for } 0 \leq j \leq L_m-1$$

and

$$y(mS_s+k_m+j) = x(mS_a+j) \tag{3}$$

$$\text{for } L_m \leq j \leq N-1$$

where f(j) is a weighting vector to be discussed. The algorithm is initialized by setting the first $y(j) = x(j)$ for $j = 1 \cdots N$.

III. SPEECH EXPANSION APPLICATIONS

We will first consider a one-way transformation of speech. That is, we wish to expand a speech segment for the purpose of listening to it in the expanded state. If the pitch and formant frequencies of the expanded speech are not preserved, the result is no better than slowing down a tape recording of the speech. The SOLA method will preserve both pitch and formant frequencies if the input frame length, N, is long enough to include two pitch periods. A similar requirement for the input frame length exists for the

23ACSSC-12/89/0714 $1.00 © 1989 MAPLE PRESS

estimation of short-time autocorrelation functions [4].

For our "telephone bandwidth" speech, sampled at 8000 samples/sec, a frame length of 128 samples was sufficient to capture two pitch periods, estimated to be 8 msec using the modified short-time autocorrelation function.

In the above section we mentioned the need for a weighting vector f(j). For expansion, we experimented with several weighting vectors. Several worked well, but the most efficient was to set f(j)=1 for all j in equation (2). This choice of weighting vector has the effect of concatenating the entire new input frame to the composite output vector after the overlap point of maximum cross correlation and discarding old composite vector samples after this point. Using such a weighting scheme, we might better call this the "synchronized-overlap-discard" method, because no averaging really takes place in (2).

We conducted experiments using a 4 second sample of speech by a male speaker, sampled at 8 kHz and digitized using the Ariel DSP-16 Data Acquisition System attached to an IBM PC, which ran the SOLA algorithms. Figures 1a-b show the time-domain waveform for segments of the original speech and the speech expanded 2 times. Figures 2a-b show modified short-time autocorrelation functions, which indicate pitch period, for both. Figures 3a-b show the formant frequencies calculated from a twelve pole LPC filter and overlayed on a periodogram of the speech segment for both the original and expanded speech. These figures demonstrate the preservation of both pitch period and formant frequencies by the SOLA method, when input frame size is properly chosen.

These findings apply to one-way compression of speech as well, with the exception that the weighting vectors must be differently chosen, as will be discussed below.

IV. COMPRESSION/RE-EXPANSION APPLICATIONS

When time-scale modification techniques are used in a communications system, they are used as a compression/re-expansion pair for the purpose of bandwidth reduction for transmission or storage. Consequently, quality, pitch and formants are only important in the re-expanded speech. We don't care directly about the quality of the one-way transformed speech. We are, therefore, free to to choose system parameters on the basis of computational efficiency. Although smaller frame size means that there are more frames for any sample of speech, the reduction of computations within equation (1) more than offsets this increase. We found for compression, that an input frame length of 32 samples was best. Decreasing N below this overly restricted the range of k_m, meaning that new and old signals are not allowed as great a range of overlap, causing a decrease in re-expanded speech quality.

For the compression, we again experimented with several weighting functions f(j). We found that the best results were obtained, particularly at high compression ratios, if we kept track of the number, w(j), of data points that had been averaged to compute, by (2), each value in the compressed data stream. We set the weighting, f(j), equal to 1/w(j). In this way, each new data point averaged contributed in equal proportion to the compressed value associated with it, regardless of position in the original data stream. This weighting function is the one used for pitch and formant preserving one-way compression as discussed above.

After compression for transmission or storage, we will need to re-expand the speech. This could be accomplished by reapplying equations (1) through (3), using as α the inverse of the compression ratio used to compress the speech. We have found, however, another approach that is computationally faster and improves the quality of the re-expanded speech.

We begin by transmitting (or storing) the value of the variable compression factor k_m with each S_s samples of the compressed signal. This adds about 3% to the length of the compressed data stream. These values of k_m are then used to efficiently re-expand the speech into a data stream $\hat{x}(j)$ by

$$\hat{x}(j+(m-1)S_\alpha) = y(j+mS_s+k_m) \qquad (5)$$
$$\text{for } 0 \leq j \leq N$$

In general, each $\hat{x}$ will be assigned multiple values. We simply average those values to obtain the reconstructed value of each $\hat{x}$ in the data stream.

Using this method, good quality re-expanded speech was obtained even at nominal compression ratios, α, of 1/8. Because SOLA is a completely time-domain method, it can be cascaded with other methods (VQ, LPC, FFT) to produce very low-bit rate speech transmission.

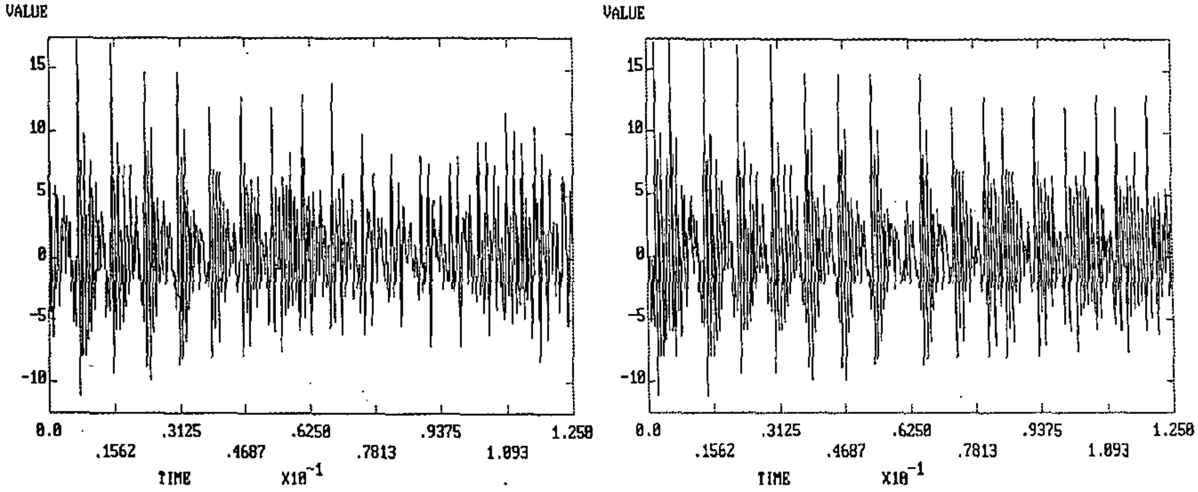
V. NOISE FILTERING

SOLA, as explained, uses a correlation method for aligning the data to be averaged during compression. As white noise is, by definition, uncorrelated, its presence in the raw speech signals is reduced when the SOLA method is applied for compression. Re-expanding the compressed signal does not recover the lost noise. Therefore, the SOLA method of compression/re-expansion has the welcome benefit of increasing the signal-to-noise ratio (snr) of the processed speech. We calculated approximately 5 dB of snr improvement after the compression/re-expansion process.

Multiple compression/re-expansion runs, varying the parameters of N and S_s between runs to allow the frames to overlap at different points each time, improved the snr over 20 dB, but with the loss of sibilant clarity, as might be expected.

VI. CONCLUSIONS

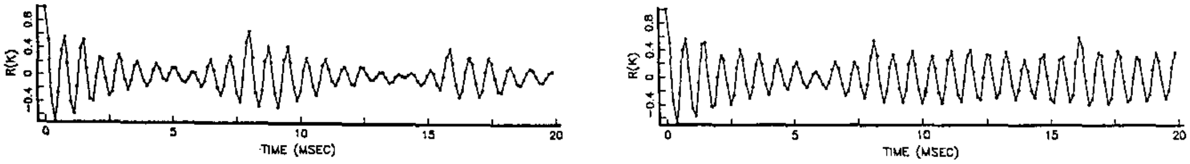
We have shown that the synchronized-overlap-add method of time scale modification can be used for one-way speech expansion (or compression), with the preservation of pitch and formant frequencies, for transmission and storage bandwidth reduction by application of compression and re-expansion, and for white-noise filtering of corrupted speech. The method is computationally simple enough to be implementable in real time.



a) Original Speech Segment

b) Expanded Speech (X2)

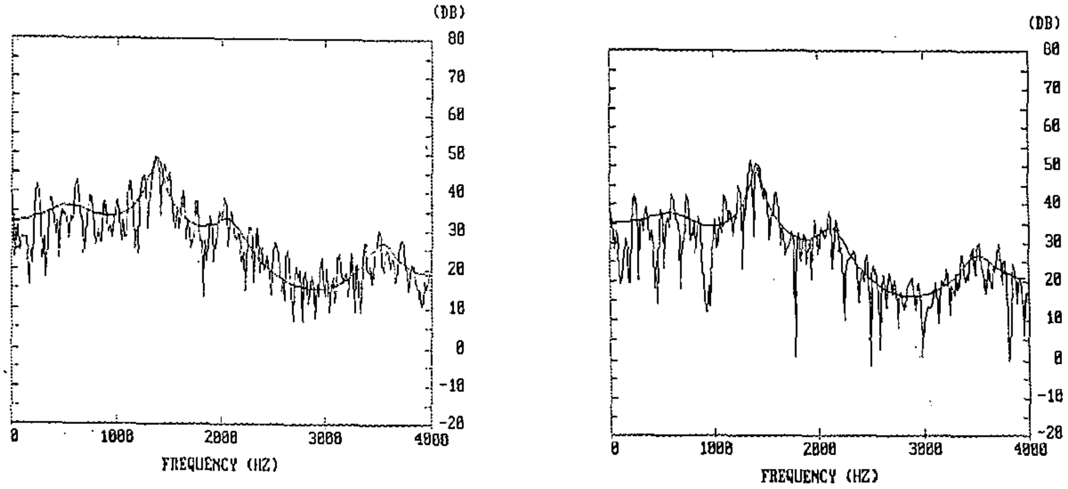
FIGURE 1: TIME DOMAIN WAVEFORM



a) Original Speech

b) Expanded Speech (X2)

FIGURE 2: MODIFIED SHORT-TIME AUTOCORRELATION FUNCTION



a) Original Speech

b) Expanded Speech (X2)

FIGURE 3: LPC SPECTRA OVERLAID ON PERIODOGRAM

REFERENCES

[1] S. Roucos and A.M. Wilgus, "High quality time scale modification for speech," in Proc. ICASSP '85, pp. 493-496, 1985

[2] J. Makhoul and A. El-Jaroudi, "Time scale modification in medium to low rate speech coding," in *Proc. ICASSP '86*, 1986, pp. 1705-1708

[3] J. Wayman and D. Wilson, "Some improvements on the synchronized-overlap-add method of time scale modification for use in real-time speech compresson and noise filtering", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-36, pp 139-140, Jan. 1988.

[4] L. R. Rabiner and R.W. Schafer, Digital Processing of Speech Signals, Prentice- Hall Inc., Englewood Cliffs, N.J., 1976