

실시간 구현을 위한 SOLA 알고리즘의 계산량 감소에 관한 연구

함명규, 정현욱, 배명진
승실대학교 정보통신공학과
156-743 서울시 동작구 상도동 1-1
mkham@korea.com

A Study on the Reduction of Maximum Complexity in SOLA Algorithm for Real Time Implementation

MyungKyu Ham, HyunUk Jung, MyungJin Bae
Dept. Information and Telecommunication Engr., Soongsil University
1-1 Sangdo-5Dong, Dongjak-Ku, Seoul 156-743, KOREA
mkham@korea.com

요 약

6.22MIPS 가까이 감소시킬 수 있었다.

음성속도변환(TSM : Time Scaling Modification) 알고리즘은 시간축에서 음성 신호의 속도를 변환할 수 있는 방식이다. 이러한 방법으로는 OLA(Overlap Add), SOLA (Synchronized Overlap Add) 알고리즘 등이 연구되어 왔다. 2 가지 방식 중에도 동기화를 시켜 overlap 을 시키는 SOLA 알고리즘이 OLA 방법에 비해 음질이 우수하다.

본 논문에서는 TMS320C5416 DSP 에 계산량이 감소된 SOLA 알고리즘을 실시간 구현하였다. 기존의 SOLA 알고리즘에서 동기화를 위해 사용하고 있는 cross-correlation 함수는 곱셈연산에서 발생하는 bit 의 dynamic range 가 커서 나눗셈 연산에서도 과도한 연산량을 필요로 한다. 따라서, 이러한 계산량의 감소를 위해 기존의 cross-correlation 함수가 대신 더하기와 빼기의 연산으로 수행되는 NAMDF 함수를 사용하여 계산량을 줄였다. 제안한 방법을 SOLA 알고리즘에 적용하여 성능 평가를 실시하였다. TMS320C5416 DSP 에 실시간으로 실험한 결과 NAMDF 함수를 사용하였을 경우 음질의 저하가 거의 없었으며, 계산량을 기존의 cross-correlation 방식에 비해

1. 서 론

음성속도변환은 시간축에서 입력 신호를 압축하거나 확장하여 신호의 재생 속도를 변환하는 것이다. 응용 분야로는 MP3 플레이어와 같은 휴대용 오디오 기기에 외국어 학습같은 부가적인 기능 구현을 위해 특정 음성 복호화기와 같이 사용되기도 한다.

음성 속도 변환 알고리즘은 시간축을 변환하는 방법으로 입력 신호를 윈도우 단위로 분할하여 이웃한 윈도우간에 Overlap-Add 연산 과정을 거쳐 입력 신호를 압축하거나 확장하는 OLA(Overlap-Add) 알고리즘과 이웃한 윈도우간의 피치 동기를 이용하여 Overlap-Add 를 함으로서 OLA 의 클릭킹과 잔향 단점을 극복하여 보다 자연스러운 출력 음성을 얻을 수 있는 SOLA(Synchronized Overlap-Add) 알고리즘과 다양한 SOLA 변형 알고리즘이 존재한다.

본 논문에서는 실시간 구현을 위해 기존의 SOLA 알고리즘에서 동기화를 위해 사용되는 cross-correlation 함수에 대해 계산량을 줄이기 위해 NAMDF

합수로 대체하여 계산량을 감소시키는 방법을 제안하고자 한다.

2. SOLA 알고리즘

SOLA 알고리즘은 시간 영역에서 템포를 변환시키는 대표적인 방법으로 이웃한 윈도우간의 피치 정보를 이용하여 Overlap-Add 연산을 수행함으로써 기존 OLA 방법의 단점을 개선한 알고리즘이다[1][2].

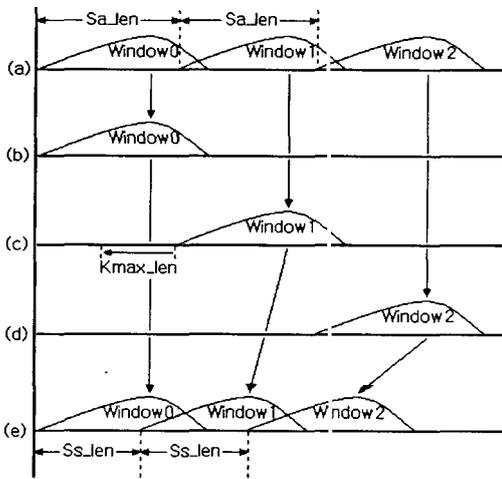


그림 2-1. SOLA-B 알고리즘을 이용한 압축모드 속도변환

- (a) 원음성 신호, (b) 윈도우 0의 신호,
- (c) 윈도우 1의 신호, (d) 윈도우 2의 신호,
- (e) 출력된 신호

그림 2-1 에서 윈도우의 길이는 원 입력 신호에 윈도우를 곱해 일정한 크기를 가지는 프레임 길이를 나타내고, Sa 는 분석 시프트(Analysis Shift)로서 입력 신호의 분석 분할 단위 Ss 는 합성 시프트(Synthesis Shift)로서 출력 신호의 합성 분할 단위, k_{max} 는 연속된 2 개의 프레임간 피치 동기를 맞추기 위한 것으로 피치 검색의 최대 이동 범위를 정의한다. 또한 속도 변화율은 $\alpha = Ss/Sa$ 의 값으로 정의되어, 만약 α 의 값이 1 보다 작으면 음성 압축 효과에 의해 음성 속도는 원음성보다 빠르게 되고,

1 보다 클 경우에는 음성 확장 효과에 의해 원음성의 속도보다 느리게 된다. 일반적으로 속도 변화율 α 는 0.5~ 2.0사이의 값으로 제한된다[3].

SOLA 는 먼저 원 신호에 윈도우를 곱해 일정한 크기를 갖는 프레임으로 잘라내고, 입력 신호 $x(n)$ 의 첫 번째 프레임을 변환되는 출력 신호 $y(n)$ 의 첫 번째 프레임에 그대로 복사하며, 다음 프레임을 얻을 때는 입력 신호의 분석 단위인 Sa 만큼 일정한 간격으로 이동시키면서 윈도우를 오버래핑하여 다음 프레임을 구하게 된다. 연속된 2 개의 프레임을 정의하게 되면, 다음은 이웃한 프레임간 피치 동기화 길이인 k 값을 구하여 프레임을 재배치하고 중복된 프레임 샘플에 가중치를 주어 더하여 최종적으로 속도 변환된 출력 신호를 얻게 된다. 연속된 2 개의 프레임간에 동기화 길이 값인 k 를 구하는 방법은 식(2.1)의 상호상관성 값을 구하여 이를 최대화하는 값으로 결정하게 되며

$$R_{xy}^m(k) = \frac{\sum_{j=0}^{Lm-1} y(mSa+k+j)x(mSa+j)}{\sqrt{\sum_{j=0}^{Lm-1} y^2(mSa+k+j) \sum_{j=0}^{Lm-1} x^2(mSa+j)}} \quad (2.1)$$

식(2.1)에서 Lm 은 프레임 $x(mSa+j)$ 와 $y(mSa+k+j)$ 의 오버래핑 길이이다. 동기화 길이 값인 k 를 찾게 되면, 최종 출력신호 $y(n)$ 은 다음식 (2.2)에 의하여 오버래핑 구간에서 $x(mSa+j)$ 와 $y(mSa+k+j)$ 에 가중치 $f(j)$ 를 더하여 프레임들을 재배치하고, 최종적으로 원 입력신호와 다른 길이의 신호 즉, 속도가 변환된 신호를 얻을 수 있다.

$$\begin{aligned} y(mSs+k+j) &= (1-f(j))y(mSs+k+j) \\ &\quad + f(j)x(mSa+j), \text{ for } 0 \leq j \leq Lm-1 \\ y(mSs+k+j) &= x(mSa+j), \text{ for } Lm \leq j \leq N-1 \end{aligned} \quad (2.2)$$

3. NAMDF 함수

AMDF(Average Magnitude Difference Function)법은 자기상관함수법에 수행하는 곱 연산을 절대값과 차분값으로 대신하기 때문에 상대적으로 빠르다는 장점이 있다. 이는 곧 실시간 처리에 적합하다[4].

$$AMDF(d) = \sum_{n=1}^N |s(n) - s(n-d)| \quad (3.1)$$

여기서 $s(n)$ 은 음성신호이고 d 는 지연인자이다. AMDF 법에서는 피치주기 배수에 최저값을 이루지만 NAMDF 를 사용시 피치주기 배수에 거의 영값을 갖게 된다.

현재 프레임의 피치를 측정하는 방법으로는 다음과 같이 NAMDF 를 정의하여 사용할 수 있다.

$$AMDF(d) = \frac{\sum_{n=1}^N |s(n) - s(n-d)|}{\sum_{n=1}^N |s(n)| + |s(n-d)|} \quad (3.2)$$

여기서 $s(n)$ 은 음성신호이고 N 은 NAMDF 를 구하려는 윈도우 구간이다. 지연인자 d 를 점차 증가시키면서 NAMDF 를 구해보면, 지연인자가 프레임내 음성피치에 정수배가 될 때마다 NAMDF 는 거의 영이 된다. AMDF 는 곱셈을 사용하지 않는 장점이 있다.

4. 실험 실시간 구현 및 결과

제안한 방법의 성능을 평가하기 위해 범용 16bit fixed-point DSP 를 이용하여 실시간 구현하였다. SOLA 알고리즘의 실시간 구현에 사용된 보드는 TI 사의 TMS320C5416 EVM 보드를 사용하였다[5]. 개발순서는 다음과 같이 실시하였다. 보드에 SOLA Fixed-point C 소스를 먼저 DSP 상에서 비실시간 동작되도록

구현하였다. 그리고, 비실시간 동작된 소스에 대해 루틴별로 MIPS 를 측정하고 함수들의 통합 및 불필요한 메모리할당을 제거하여 C 레벨에서 최적화를 하였다. 핸드 어셈블 작업에 들어가기 전에 5416 은 하버드 구조를 가지고 있기 때문에 효율적으로 메모리를 관리하고 수행최적화를 이루기 위해서는 변수들의 메모리 할당과 파이프라인 충돌에 주의하여야 한다. 따라서, 메모리 할당과 함수간 인자전달방법에 대해 명백한 사전정의를 하였다. 이러한 과정을 기본으로 하여 본 논문에서는 프로그램의 전 과정이 어셈블리어로 구현되었다[5].

음성 시료로는 다양한 음질 평가를 위해 남성과 여성에 대한 한국어와 영어의 2 가지 언어를 사용하였다. 연구실 환경에서 발생한 음성을 8kHz 로 표본화하고 16bit 로 양자화하여 사용하였다.

그림 4-1 은 원음성 신호를 보여주고 있다. 그림 4-2 는 TMS320C5416 DSP 를 이용하여 NAMDF 방법을 사용한 SOLA 알고리즘에 $\times 0.5$ 배로 압축된 신호를 나타낸다. 그리고, 그림 4-3 은 TMS320C5416 DSP 를 이용하여 NAMDF 방법을 사용한 SOLA 알고리즘에 $\times 2.0$ 배로 압축된 신호를 나타낸다.

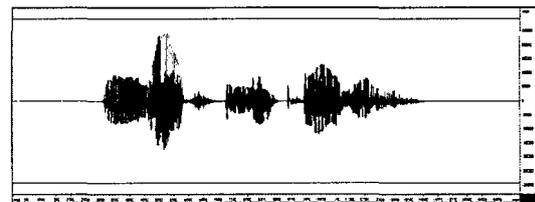


그림 4-1. 원음성 신호

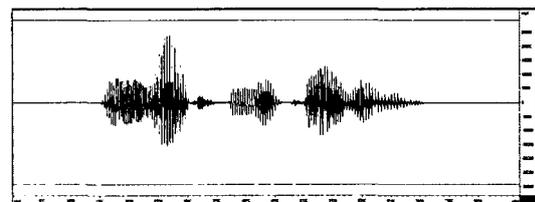


그림 4-2. SOLA 알고리즘을 이용한 $\times 0.5$ 로 압축된 신호

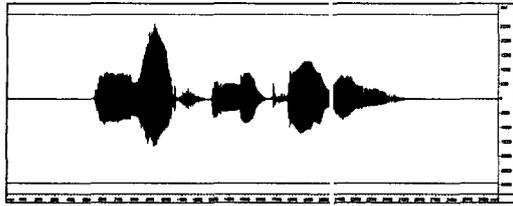


그림 4-3. SOLA 알고리즘을 이용한 x2.0 으로 압축된 신호

실시간 실험한 결과는 계산량(MIPS)과 음질로 평가하였다. 표 4-1 은 cross-correlation 과 NAMDF 방법을 사용하여 실시간 구현된 음성 신호에 대한 MOS 를 나타내고 있다. 그리고, 표 4-2 는 TMS320C5416 DSP 에서 측정된 cross-correlation 과 NAMDF 함수의 MIPS 를 나타내고 있다. 표의 결과와 같이 음질에는 차이가 없었으며, 계산량은 6.22MIPS 정도가 감소하였다.

표 4-1 기존 방법과 제안한 방법의 MOS 비교

| | Cross-correlation | NAMDF |
|------|-------------------|-------|
| 발성 1 | 3.99 | 3.97 |
| 발성 2 | 3.91 | 3.89 |
| 발성 3 | 3.95 | 3.93 |
| 발성 4 | 3.87 | 3.86 |
| 평균 | 3.93 | 3.91 |

표 4-2 기존 방법과 제안한 방법의 MIPS 비교

| Cross-correlation | NAMDF |
|-------------------|-------|
| 8.74 | 2.52 |

5. 결 론

음성속도변환은 시간축에서 입력 신호를 압축하거나 확장하여 신호의 재생 속도를 변환하는 것으로서 노래방 기기의 음악 템포변환, 외국어 학습을 위한 음성 재생 속도 변환, 그리고 데이터 압축 및 복원 등 다양한 분야에 응용되어질 수 있다. 특히 최근에는

MP3 플레이어와 같은 휴대용 오디오 기기에 외국어 학습같은 부가적인 기능 구현을 위해 특정 음성 복호화기와 같이 사용되기도 한다.

본 논문에서는 TMS320C5416 DSP 에 계산량이 감소된 SOLA 알고리즘을 실시간 구현하였다. 기존의 SOLA 알고리즘에서 동기화를 위해 사용하고 있는 cross-correlation 함수는 곱셈연산에서 발생하는 bit 의 dynamic range 가 커서 나눗셈 연산에서도 과도한 연산량을 필요로 한다. 따라서, 이러한 계산량의 감소를 위해 기존의 cross-correlation 함수가 대신 더하기와 빼기의 연산으로 수행되는 NAMDF 함수를 사용하여 계산량을 줄였다. 제안한 방법을 SOLA 알고리즘에 적용하여 성능 평가를 실시하였다. TMS320C5416 DSP 에 실시간으로 실험한 결과 NAMDF 함수를 사용하였을 경우 음질의 저하가 거의 없었으며, 계산량을 기존의 cross-correlation 방식에 비해 6.22MIPS 가까이 감소시킬 수 있었다.

6. 참 고 문 헌

- [1] E.Hardam "High quality time scale modification of speech signals using fast synchronized overlap add algorithms," IEEE. Processings of the 1990 International Conference on Acoustics, Speech, and Signal Processing, vol. 1, 409-412, Feb. 1990.
- [2] D.J.Hejna Jr., "Real-Time Time-Scale Modification of Speech via the Synchronized Overlap-Add Algorithm." Master's thesis, Department of Electrical Engineering and Computer Science, MIT, Feb. 1990.
- [3] 박규식, 오승록, 김선영, "G.729 음성 복호화기와 듀얼 SOLA 알고리즘을 통합한 최적의 음성 속도 변환 시스템." 한국음향학회지, 21(3), 293-303, April 2002.
- [4] L.R. Rabiner and R.W. Schafer, "Digital processing of Speech Signals Englewood Cliffs", New Jersey : Prentice-Hall, 1978.
- [5] 안도건, 유승균, 최용수, 이재성, 강태익, 박성현, "16 비트 고정 소수점 DSP 를 이용한 다채널 G.729A 음성 복호화기의 실시간 구현." 한국음향학회지, 19(4), 45-51, May 2000.