

신경망필터를 이용한 음질향상

Speech Enhancement using the Neural Network Filter

김종우*, 공성곤
 숭실대학교 전기공학과

Jong-Woo Kim*, Seong-Gon Kong

Department of Electrical Engineering, Soongsil University

jongwoo@ee.ssu.ac.kr, skong@ee.ssu.ac.kr

ABSTRACT

본 논문에서는 잡음환경에서의 음성신호복원(Speech Enhancement) 시스템 구현을 목적으로 한다. 이를 위한 적응필터로서 LMS(Least Mean Square)알고리즘 FIR필터를 제안한다. 또 정밀 필터로서 신경망 필터를 제안한다. 잡음환경에서의 음성신호 복원 시스템은 잡음에 의해 왜곡된 음성신호에서 잡음성분만을 제거함으로써 음성신호를 복원하는 시스템이다. 일반적으로 잡음은 시변특성과, 비선형적인 전달특성을 갖는다. 그러므로 파라미터가 고정된 필터로는 제어하기가 힘들다. 이러한 이유로 본 논문에서는 LMS알고리즘 적응필터를 적용한다. 신경망 필터는 오차 역전파 학습 알고리즘에 의해 오차를 최소화 하는 방향으로 필터의 파라미터를 수정한다. 제안한 필터로 잡음환경에서의 음성신호복원 시스템을 구성하고, 실험을 통해 필터의 성능을 확인한다.

1. 서론

산업환경의 발전과 이에 동반되어 발생하는 소음공해로 인한 피해가 증가하는 추세이다. 일반적으로 소음은 상호 음성통신의 방해뿐 아니라 정신적, 육체적인 피로를 가중시켜 심신의 능력을 저하시키는 환경오염의 한 종류로서 취급된다. 소음공해로 인한 육체적, 정신적인 피해가 증가함에 따라 소음을 제거하려는 노력이 계속되어져 왔다. 소음과 잡음환경에서는 육성을 이용한 상호 의사소통 및 통신에 많은 어려움이 있다. 현대에 이르러 유선 및 이동통신의 사용자가 증가하고 있다. 유·무선통신에 있어서 잡음환경은 정확한 음성의 전달에 직접적인 악영향을 주는 요인이므로 반드시 제거해야 하는 요소이다. 잡음환경에서의 음성신호의 복원은 음성통신에 있어서 상당히 중요한 부분을 차지한다. 전동기장치, 발전기장치를 갖춘 실내와 같이 소음이 심한 환경에서는 음성신호가 소음에 의해 오염되어 상호 음성통신 및 통신장비를 이용한 음성통신에 많은 어려움이 있다. 이러한 경우 소음제어를 '이용한 여러 가지 장비를 사용하지 않는 경우 음성통신은 불가능하게 된다. 또, 현대에 이르러 보안의 필요성에 의해 창출된 첨단과학인 인지과학 중에서 음성인식을 이용한 화자인식등의 전처리과정으로 음성신호에서 잡음을 제거하는 것은 필수 전처리과정으로 인식되어 있다.

2. 음성신호복원 시스템

잡음환경에서의 음성신호복원 시스템은 외부로부터 유입되는 잡음에 의해 오염된 음성신호에서 잡음성분만을 제거함으로써 음성신호를 복원하는 시스템이다.

2.1 음성신호복원 시스템 구조

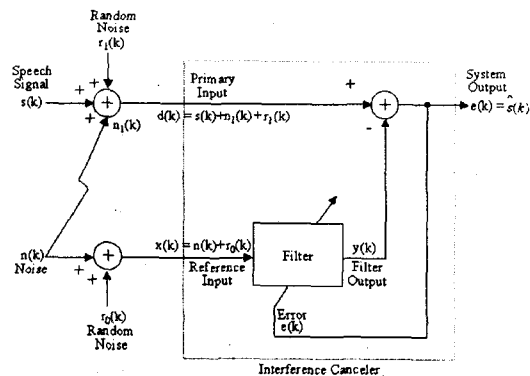


그림 1 시스템 블록다이어그램

전동기의 소음이 심한 공장내에서 전화통신을 고려한다면, 그림 1은 이러한 상황을 적용한 잡음환경에서의 음성신호복원 시스템의 구성도이다. $s(k)$ 는 복원하고자 하는 음성신호이다. $n(k)$ 는 잡음원의 잡음신호이다. $n_1(k)$ 는 시지연된 잡음신호이다. $r_0(k)$ 와 $r_1(k)$ 는 임의의 잡음신호이다.

음원으로부터의 잡음으로 고려하는 랜덤노이즈이다. $y(k)$ 는 필터의 출력이다. Desired Output인 Primary Input은 $s(k) + n_1(k) + r_1(k)$ 으로 설정한다. 필터의 입력인 Reference Input은 $n(k) + r_0(k)$ 으로 설정한다. 오차 $e(k)$ 는 Primary Input과 필터의 출력 $y(k)$ 와의 차로 설정한다. 결국 필터는 시지연된 잡음원신호를 추종하여 이를 상쇄시키고 랜덤노이즈를 제거하는 역할을 담당한다. $\hat{s}(k)$ 는 System Output이며 오차 $e(k)$ 는 같은 신호가 된다.

3. LMS알고리즘 FIR 적응필터

3.1 필터의 구조

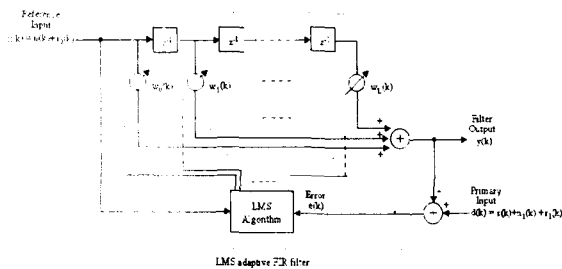


그림 2 LMS알고리즘 FIR필터 세부구조

그림 2는 LMS알고리즘을 적용한 FIR필터의 세부구조이다. 신호처리에 있어서 LMS알고리즘은 간단하고도 빠른 연산속도 때문에 많이 사용한다. LMS알고리즘은 입력에 대해서 오프라인 아닌 온라인 처리가 가능하다. 여기서 $w_0(k), w_1(k), \dots, w_L(k)$ 는 FIR필터의 각 탭 가중치이며, z^{-1} 은 이산시간에서의 한샘플 시지연을 나타낸다. FIR필터는 $n(k) + r_0(k)$ 을 입력으로 하여 필터의 출력으로 $y(k)$ 를 얻는다. $s(k) + n_1(k) + r_1(k)$ 와 $y(k)$ 의 차로 오차 $e(k)$ 를 생성한다. 오차 $e(k)$ 와 $n(k) + r_0(k)$ 을 입력으로 하여 LMS알고리즘을 동작시키며, FIR필터의 각 가중치를 업데이트 한다. 필터의 차수는 L 차가 된다.

3.2 파라미터 업데이트

입력을 행렬식으로 표현하여 이를 $X(k)$ 로 놓으면 식(3-1)과 같다.

$$X(k) = [x(k) \ x(k-1) \ \dots \ x(k-L)]^T \quad (3-1)$$

각 가중치를 행렬식으로 표현하면, 식(3-2)와 같다.

$$W(k) = [w_0(k) \ w_1(k) \ \dots \ w_L(k)]^T \quad (3-2)$$

입력이 식(3-1)과 같고 가중치가 식(3-2)와 같을 때, FIR필터의 출력 $y(k)$ 는 식(3-3)과 같다.

$$y(k) = X(k)^T W(k) \quad (3-3)$$

오차 $e(k)$ 는 desired output인 $s(k) + n_1(k) + r_1(k)$ 와 FIR필터의 출력 $y(k)$ 의 차로 설정한다. $s(k) + n_1(k) + r_1(k)$ 를 $d(k)$ 로 치환하고, 오차 $e(k)$ 를 식으로 표현하면 식(3-4)와 같다.

$$e(k) = d(k) - X(k)^T W(k) \quad (3-4)$$

식(3-5)는 LMS알고리즘을 적용하였을 때 가중치 업데이트 식이다.

$$W(k+1) = W(k) + 2\mu e(k) X(k) \quad (3-5)$$

k 번째 스텝에 대한 입력 $X(k)$ 과 오차 $e(k)$, 수렴속도 및 안정도에 관여하는 양의상수 μ , 그리고 상수 2의 곱으로 이루어진 텀이 k 번째 스텝에 대한 웨이트 $W(k)$ 를 업데이트 시켜 $k+1$ 번째 스텝에 대한 웨이트 $W(k+1)$ 를 생성한다.

4. 신경망 필터

4.1 신경망 필터의 구조

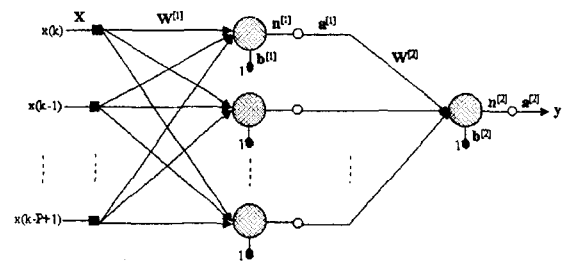


그림 3 신경망필터의 구조

그림 3은 신경망필터의 구성도이다. 필터입력인 Reference Input을 $x(k)$ 라 치환한다. 입력으로는 $x(k)$ 부터 $x(k-P+1)$ 까지로 P 개만큼의 입력을 갖는다.

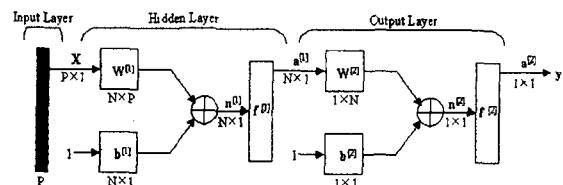


그림 4 내부연산구조

그림 4는 신경망필터 내부의 연산구조를 나타낸다. 그림 4-1과 4-2에서 X 는 입력매트릭스이다. W^{11} 는 입력층에서 은닉층으로 연결되는 가중치 매트릭스이다. W^{21} 는 은닉층에서 출력층으로 연결되는 가중치 매트릭스이다.

$\mathbf{n}^{[1]}$ 는 활성화함수를 거치기 전의 은닉층 출력 매트릭스이다. $\mathbf{n}^{[2]}$ 는 활성화함수를 거치기 전의 출력층 출력 매트릭스이다. $\mathbf{a}^{[1]}$ 는 활성화함수를 거친 은닉층의 출력이다. $\mathbf{a}^{[2]}$ 는 활성화함수를 거친 출력층의 출력이다. $\mathbf{b}^{[1]}$ 은 은닉층의 바이어스 매트릭스이다. $\mathbf{b}^{[2]}$ 은 출력층의 바이어스 매트릭스이다. \mathbf{y} 는 신경망필터의 최종출력으로 $\mathbf{a}^{[2]}$ 와 같다. 은닉층의 활성화함수 $f^{[1]}$ 과 $f^{[2]}$ 는 Hyperbolic Tangent Sigmoid 함수로 은닉층과 출력층의 활성화함수이다.

4.2 파라미터 학습

본 논문에서 사용한 신경망 필터의 구조는 P-N-1이다. 입력층과 은닉층, 그리고 출력층을 가지며 각 층을 연결하는 가중치를 업데이트 하는 알고리즘으로 오차 역전파 학습 알고리즘을 사용한다.

$$\mathbf{a}^{[1]} = f^{[1]}(\mathbf{W}^{[1]} \mathbf{X} + \mathbf{b}^{[1]}) \quad (4-1a)$$

$$\mathbf{y} = \mathbf{a}^{[2]} = f^{[2]}(\mathbf{W}^{[2]} \mathbf{a}^{[1]} + \mathbf{b}^{[2]}) \quad (4-1b)$$

식(4-1a)는 은닉층의 출력이고, 식(4-1b)는 출력층의 출력으로 최종출력을 나타낸다.

$$w_{i,j}^{[m]}(k+1) = w_{i,j}^{[m]}(k) - \alpha \frac{\partial \hat{E}}{\partial w_{i,j}^{[m]}} \quad (4-2)$$

$$b_i^{[m]}(k+1) = b_i^{[m]}(k) - \alpha \frac{\partial \hat{E}}{\partial b_i^{[m]}} \quad (4-3)$$

식(4-2)와 식(4-3)은 근사화시킨 MSE $\hat{E} = e(k)^2$ 와 Gradient Descent방법을 이용한 각층의 가중치와 바이어스의 업데이트 일반식이다. 설정한 \hat{E} 를 $w_{i,j}^{[m]}$ 로 편미분하기 위해서는 Chain-Rule을 적용해야 한다. Chain-Rule을 적용하여 \hat{E} 를 $w_{i,j}^{[m]}$ 로 편미분하면 다음식과 같다.

$$\frac{\partial \hat{E}}{\partial w_{i,j}^{[m]}} = \frac{\partial \hat{E}}{\partial n_i^{[m]}} \frac{\partial n_i^{[m]}}{\partial w_{i,j}^{[m]}} \quad (4-4)$$

$$\frac{\partial \hat{E}}{\partial b_i^{[m]}} = \frac{\partial \hat{E}}{\partial n_i^{[m]}} \frac{\partial n_i^{[m]}}{\partial b_i^{[m]}} \quad (4-5)$$

sensitivity $s_i^{[m]}$ 를 다음식으로 정의 한다.

$$s_i^{[m]} = \frac{\partial \hat{E}}{\partial n_i^{[m]}} \quad (4-6)$$

식(4-2)와 식(4-3)에 sensitivity를 적용시켜 매트릭스폼으로 전개하면 다음식과 같다.

$$\mathbf{W}^{[m]}(k+1) = \mathbf{W}^{[m]}(k) - \alpha \mathbf{s}^{[m]} (\mathbf{a}^{[m-1]})^T \quad (4-7)$$

$$\mathbf{b}^{[m]}(k+1) = \mathbf{b}^{[m]}(k) - \alpha \mathbf{s}^{[m]} \quad (4-8)$$

$\mathbf{W}^{[2]}$ 와 $\mathbf{b}^{[2]}$ 를 업데이트하는데 관여하는 sensitivity를 $s^{[2]}$ 라 할 때, 이와 관계된 식은 다음과 같다.

$$\mathbf{s}^{[2]} = -2e(k) \mathbf{F}^{[2]'}(\mathbf{n}^{[2]}) \quad (4-9)$$

$\mathbf{W}^{[1]}$ 와 $\mathbf{b}^{[1]}$ 을 업데이트하는데 관여하는 sensitivity를 $s^{[1]}$ 라 할 때, 이와 관계된 식은 다음과 같다.

$$\mathbf{s}^{[1]} = \mathbf{F}^{[1]'}(\mathbf{n}^{[1]}) (\mathbf{W}^{[2]})^T \mathbf{s}^{[2]} \quad (4-10)$$

본 논문에서 사용한 출력층의 활성화 함수는 Pure Linear이므로 이의 미분형 매트릭스폼은 다음과 같다.

$$\mathbf{F}^{[2]'}(\mathbf{n}^{[2]}) = [1 - (\mathbf{a}^{[2]})^2] \quad (4-11)$$

본 논문에서 사용한 은닉층의 활성화 함수는 Hyperbolic Tangent Sigmoid이므로 이의 미분형 매트릭스폼은 다음과 같다.

$$\mathbf{F}^{[1]'}(\mathbf{n}^{[1]}) = \begin{bmatrix} 1 - (a_1^{[1]})^2 & 0 & \dots & 0 \\ 0 & 1 - (a_2^{[1]})^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 - (a_N^{[1]})^2 \end{bmatrix} \quad (4-12)$$

5. 실험 및 결과

5.1 실험환경 설정

실험에 사용한 음성신호 및 잡음신호는 8[KHz] 샘플링을 거쳐 추출한다. 실험에 사용한 총 샘플의 개수는 5초동안의 샘플들로 40000개의 데이터로 구성된다. LMS알고리즘을 적용한 FIR필터의 경우 오프라인 학습이 필요 없다. 온라인으로 처리가 가능하기 때문에 학습 데이터를 설정할 필요가 없지만, 신경망 필터의 경우 학습데이터를 설정해야 한다. 학습을 거친 후 테스트 데이터의 입력으로 해당하는 필터의 성능을 알 수 있다.



그림 5 음성신호 $s(k)$

그림 5는 "안녕하세요. 지능신호처리 연구실 김

중우입니다.”라는 음성신호로 $s(k)$ 에 해당한다.

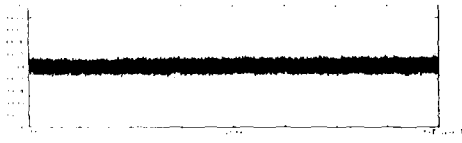


그림 6 Reference Input $n(k) + r_0(k)$

잡음원으로서 발생하는 전동기소음을 $n(k)$ 잡음원으로 설정한다. 그림 6은 Reference Input으로 잡음원과 랜덤노이즈의 합이다.

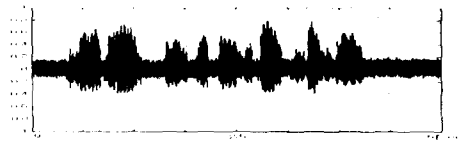


그림 7 Primary Input $s(k) + n_1(k) + r_1(k)$

그림 7은 Primary Input이다.

5.2 LMS알고리즘 FIR필터 성능확인

실험에서는 $n_1(k)$ 의 시지연을 10샘플로 설정하였으며, 학습상수는 0.05이다.

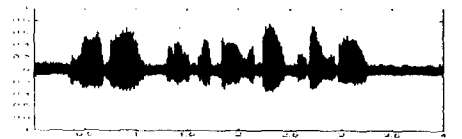


그림 8-a 9차필터, SNR:5.80[dB]

그림 8-a는 필터차수를 9차로 설정 한 경우의 결과로 SNR은 5.80[dB]이다.

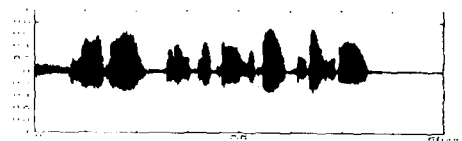


그림 8-b 10차필터, SNR:11.85[dB]

그림 8-b는 필터차수를 10차로 설정 한 경우의 결과로 SNR은 11.85[dB]이다. 9차보다 많은 성능향상을 알 수 있다. 20차로 한 경우는 10차로 설정한 경우보다 SNR이 떨어짐을 확인하였다. 결과로부터 적절한 필터차수에서 더 좋은 성능을 보임을 알 수 있다.

5.3 신경망 필터의 성능확인

입력층 14, 은닉층 5, 출력층 1의 구조의 신

경망필터로 실험 하였으며, 학습상수는 0.1이고 $n_1(k)$ 의 시지연을 10샘플로 설정한다.

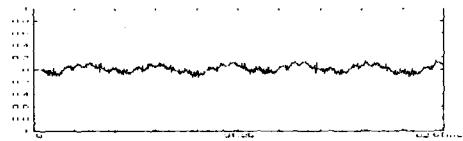


그림 9 학습데이터

그림 9는 62.5[ms]동안의 학습데이터로 500샘플에 해당한다.

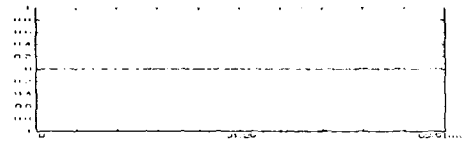


그림 10 학습을 마친 학습데이터

그림 10은 오차를 최소화하는 방향으로 학습을 마친 학습데이터이다.



그림 11 신경망필터에 의해 복원된 음성신호

그림 11은 학습데이터에 의해 오차를 최소화하는 방향으로 튜닝된 웨이트와 바이어스를 가진 신경망필터에 전체 테스트데이터를 입력하여 얻은 결과로 SNR은 약 26.36[dB]로 좋은 성능을 보인다.

6. 결론

본 논문에서는 잡음에 의해 오염된 음성신호를 복원하여 음질을 향상시키기 위한 시스템 구축을 목적으로 하며, 이는 interference canceling 모델을 기본으로 한다. 시스템의 필터부분에는 LMS알고리즘 FIR 적응필터와 신경망필터를 제안한다. 실험결과에서 알 수 있듯이 제안한 두가지의 필터를 시스템에 적용시 좋은 성능을 보임을 확인 할 수 있다.

참고문헌

- [1] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, : Prentice-Hall, 1985.
- [2] S. Haykin, *Adaptive Filter Theory*, : Upper Saddle River, New Jersey, Prentice-Hall, 1996.
- [3] Hagan, Demuth, Beale, *Neural Network Design*, : PWS Publishing Company, 1996.
- [4] W. G. Knecht and M. E. Schenkel, "Neural Network Filters for Speech Enhancement," *IEEE Trans. on Speech And Audio Processing*, Vol. 3, No. 6, 433, Nov. 1995.