실시간 고차통계 정규화와 Smoothing 필터를 이용한 강인한 음성인식

정주현, 송화전, 김형순
부산대학교 전자공학과

Robust Speech Recognition Using Real-Time High Order Statistics Normalization and Smoothing Filter

Ju Hyun Jeong, Hwa Jeon Song, Hyung Soon Kim
Dept. of Electronics Engineering, Pusan National University.

{jeongju78, hwajeon, kimhs}@pusan.ac.kr

Abstract

The performance of speech recognition is degraded by the mismatch between training and test environments. Many methods have been presented to compensate for additive noise and channel effect in the cepstral domain, and Cepstral Mean Subtraction (CMS) is the representative method among them. Recently, high order cepstral moment normalization method has introduced to improve recognition accuracy. In this paper, we apply high order moment normalization method and smoothing filter for real-time processing. In experiments using Aurora2 DB, we obtained error rate reduction of 49.7% with the proposed algorithm in comparison with baseline system.

Ⅰ. 서론

주변 잡음과 채널 특성으로 인한 훈련환경과 인식환경 사이의 불일치는 음성 인식기의 성능을 저하시킨다. 이러한 불일치를 극복하기 위해 다양한 전처리 방법이 시도되고 있으며, Cepstral Mean Subtraction (CMS) 을 기반으로 하는 방법이 대표적이다. CMS 방법은 캡스트럼 영역에서 채널 특성을 포함하는 DC 성분을 제거해줌으로서 채널 특성에 덜 민감하고 주변잡음에 강인한 특성을 가지게 하며, Cepstral Variance Normalization (CVN)은 분산을 정규화 함으로서 잡음 음성의 확률분포를 원음성의 확률분포에 가깝게 해준다[1]. 최근에는 원음성과 잡음음성의 확률분포 차이를 더욱 줄여주기 위해 3차 모멘트나 그보다 높은 고차 모멘트를 정규화해주는 방법이 제안되었다[2][3]. 그러나 기존의 CMS 방법과 CVN 방법은 입력음성이 모두 들어온 뒤에 특징벡터의 평균과 분산을 구할 수 있으므로 실시간 처리가 어렵다. 실시간 처리를 위한 CMS 계열의 방법으로는 Local Cepstral Mean Subtraction (LCMS) 방법과 Sequential Cepstral Mean Subtraction (SCMS) 방법이 있다. LCMS 방법과 SCMS 방법은 입력음성 전체를 이용하는 CMS 방법, 즉, Global CMS (GCMS) 방법에 비해 성능이 많이 떨어진다. 본 논문에서는 LCMS 방법의 단점을 보완하고 성능향상을 위해 고차통계를 이용한 정규화방법을 이용하였고, 고차통계의 outlier에 민감한 특성을 감소시키기 위해 smoothing filter를 도입하였다.

본 논문의 구성은 다음과 같다. 2장에서는 CMS 기반의 정규화 방법에 대해 기술하고 또한 실시간 처리를 위한 LCMS 방법과 SCMS 방법에 대해 서술한다. 3장에서는 본 논문에서 실시간 처리를 위한 고차통계방법에 대해 기술하고 4장에서 실험 및 결과를 서술한다.

마지막으로 5장에서 결론을 맺는다.

Ⅱ. CMS 기반 정규화방법

1. GCMS

잡음이 섞이지 않은 원음성 x가 임펄스 응답 h을 가지는 채널을 거쳐서 왜곡된 음성 y을 생성시킨다면 캡스트럼 영역에서 다음과 같이 표현할 수 있다.

$$y = x + h \tag{1}$$

여기서, y는 관측 캡스트럼 벡터, h는 채널 성분의 캡스트럼 벡터, 그리고 x는 입력음성의 캡스트럼 벡터이다. 따라서 식 (1)에서 바이어스 항목으로 나타나는 h를 관측 캡스트럼 벡터로부터 제거해 줌으로써 훈련과 인식환경의 차이를 보상해 인식성능의 향상을 기대할 수 있다. 순수한 음성의 캡스트럼 벡터의 평균이 0이라고 가정하면 채널 캡스트럼의 추정치는 다음과 같이 구할 수 있다.

$$\hat{b}^{GCMS} = \frac{1}{T}\sum_{t=1}^{T} y_t \tag{2}$$

여기서, y_t는 t번째 프레임의 관측벡터이고, T는 관측 벡터의 전체 프레임 수이다. GCMS 방법에 의해 채널 왜곡이 보상된 벡터 x_{GCMS}는 다음과 같다.

$$x_{GCMS} = x - \hat{b}^{GCMS} \tag{3}$$

2. LCMS 및 SCMS

GCMS 방법은 모든 음성이 시스템에 전부 입력된 후에 동작하므로 실시간 처리의 어려움이 있다. GCMS 방법의 실시간 처리를 위해 제안된 방법이 LCMS 방법이며, 입력음성 전체에 대해 평균을 취하는 것이 아니라 적당한 길이의 구간에 대해 moving average를 취함으로써 채널성분을 추정한다. T_L이 moving average를 취하는 구간이라 하면, 추정된 채널 성분은 다음과 같다.

$$\hat{b}_t^{LCMS} = \frac{1}{T_L}\sum_{t'=0}^{T_L} y_{t-t'} \tag{4}$$

LCMS 방법의 변형된 형태인 SCMS 방법은 식 (5)와 같이 t번째 켑스트럼 벡터와 $t-1$번째의 바이어스 추정벡터의 가중합으로 t번째 채널성분을 추정하는 방법이다.

$$\hat{b}_t^{SCMS} = \alpha\,\hat{b}_{t-1}^{SCMS} + (1-\alpha)\,y_t, 0 < \alpha < 1 \tag{5}$$

여기서, $\hat{b}^{SCMS}_t$는 SCMS 방법에 따른 바이어스 추정벡터이고, y_t는 캡스트럼 벡터이다.

3. 고차통계 정규화

음성인식 시스템에서 주변 잡음 환경의 변화에 따라 특징벡터의 평균 이외에 다른 통계적 특성도 달라진다. 이러한 통계적 특성을 보상해주는 방법으로 CVN 방법과 CTN 방법이 있다. CVN 방법은 CMS 방법을 통해 평균을 0으로 만든 후, 2차 모멘트인 분산을 1로 정규화 한다. CVN 방법은 CMS에 비해 원음성과 잡음음성 사이의 확률밀도함수의 차이를 더 줄이는 효과가 있다. 만약 각각의 벡터차원이 서로 독립이라고 가정한다면 CVN 방법은 다음과 같다.

$$x_{CVN} = x_{CMS} / \sqrt{E[x_{CMS}^2]} \tag{6}$$

그리고 2차 통계뿐만 아니라 3차 이상의 고차 모멘트에 대해서도 정규화가 가능하며, CTN 방법은 평균과 분산뿐만 아니라 3차 모멘트인 왜도(skewness)를 정규화 하여 원음성의 특징벡터와 잡음음성의 특징벡터사이의 확률밀도함수를 더욱 비슷하게 해준다[2]. 그러나 기존 방법은 3차 방정식의 정확한 근을 찾기가 어려운 문제가 있어 단순히 근사식만을 이용하는 CTN 방법을 도입하였다[3]. 본 논문에서 사용한 CTN 방법은 CVN 과정을 거친 특징벡터를 이용해 식 (7)과 같이 정의된 비선형 변환으로 표현한다.

$$x_{CTN} = a x_{CVN}^2 + x_{CVN} + c \tag{7}$$

식 (7)에서 x_{CTN}의 평균이 0이고 1인 분산을 가지며 3차 모멘트가 0이 되도록 a와 c를 정해야 한다. x_{CVN}의 평균은 0이고 분산은 1이므로 a와 c는 다음의 관계를 가진다.

$$\begin{aligned} E[x_{CTN}] &= E[a x_{CVN}^2 + x_{CVN} + c] \\ &= a + c = 0 \end{aligned} \tag{8}$$

c와 $-a$는 같으므로 식 (7)을 다음과 같이 다시 쓸 수 있다.

$$x_{CTN} = a(x_{CVN}^2 - 1) + x_{CVN} \quad (9)$$

식 (9)를 이용해 x_{CTN}의 3차 모멘트를 정리하면 식 (10)과 같다.

$$\begin{aligned} E[x_{CTN}^3] &= E[a(x_{CVN}^2 - 1) + x_{CVN}]^3 \\ &= a^3 E[x_{CVN}^2 - 1]^3 + 3a^2 E[x_{CVN}^2 - 1]^2 x_{CVN} \\ &\quad + 3aE[x_{CVN}^2 - 1]x_{CVN}^2 + E[x_{CVN}^3] = 0 \end{aligned} \quad (10)$$

a는 실제로 매우 작은 값을 가지므로 a의 고차부분을 무시하면 a는 식 (11)과 같이 구할 수 있다.

$$a \approx \frac{-E[x_{CVN}^3]}{3E[x_{CVN}^4 - x_{CVN}^2]} \quad (11)$$

식 (11)은 간략화 된 식이기 때문에 정확한 a를 추정하기 위해서는 2번 이상의 반복과정이 필요하다.

Ⅲ. 실시간 고차통계 정규화 방법

본 논문에서는 고차통계 정규화 방법의 실시간 구현을 위해 기존 LCMS 방법의 아이디어를 CTN 방법에 적용한 Local CTN (LCTN)을 도입하였다. 그림 1에서 N은 정확한 3차 모멘트를 정규화하기 위한 계수 a를 구하기 위한 반복과정의 횟수를 나타내고 T는 전체 프레임의 수를 가리킨다. 실험에서는 2번의 반복과정을 거쳤다. 실시간으로 3차 모멘트를 정규화 할 경우에 outlier의 값에 민감한 특성이 있다. 이런 특성을 감소시키기 위해 LCTN 방법을 사용하여 나온 값들에 대해 smoothing filter를 적용함으로써 시간 축에 따른 bias 변화량이 크지 않도록 하였다.

Ⅳ. 실험결과

1. Aurora2 데이터베이스

제안된 방법의 평가를 위해서 Aurora2 데이터베이스 [4]가 사용되었다. Aurora2 데이터베이스는 1자리에서 7자리까지의 영어 연결숫자로 구성된 TI Digit에 다양한 잡음을 인위적으로 더한 것이다. Aurora2 데이터베이스는 훈련 데이터와 테스트 데이터로 구분되어 있으며 테스트 데이터는 채널 특성은 동일하고 서로 다른 잡음이 더해진 두 개의 subset(set A, set B)과 채널특성이 다른 subset(set C)으로 총 3개의 subset으로 구성되어 있다. 잡음환경은 8가지의 잡음종류(subway, babble, car, exhibition, restaurant, street, airport, station)와 각각 5가지 잡음 레벨(clean, 20dB, 15dB, 10dB, 5dB)로 구성되어 있다. 성능평가는 각 잡음의 종류에 대해서 20dB에서 0dB까지의 잡음 레벨에 대해 수행된다. 본 논문에서는 잡음이 섞이지 않은 깨끗한 음성에 대해서만 훈련을 하는 clean condition에 대해 실험을 수행하였다.

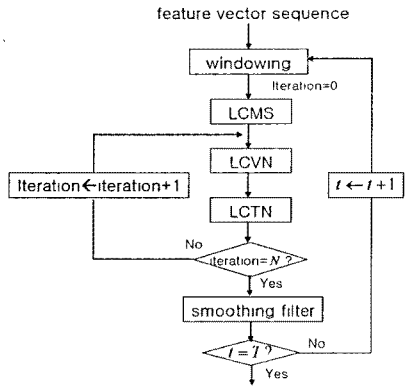

그림 1. 실시간 3차 모멘트 정규화 방법

2. 실험결과 및 검토

Aurora project의 baseline에서 사용되는 MFCC는 magnitude spectrum을 사용하여 추출되었다. 그러나, 로그 에너지와 c0, 그리고 magnitude spectrum에서 구한 MFCC 계수와 power spectrum에서 구한 MFCC 계수의 조합에 따라 인식성능은 달라진다. 그림 2의 Aurora2 데이터베이스의 baseline 시스템에 대한 실험에서 c0와 power spectrum을 이용한 MFCC 계수를 사용한 경우가 인식률이 가장 좋음을 확인했다. 이후 실험에서는 성능이 가장 좋은 c0와 power spectrum을 이용해 구한 MFCC 계수를 사용하였다.

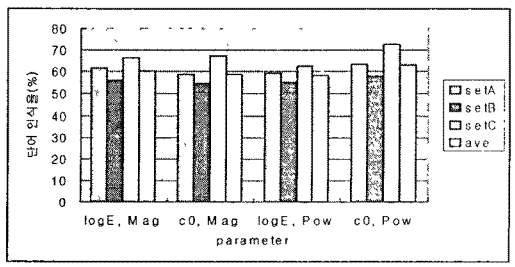

그림 2. 로그에너지와 spectrum 종류에 따른 성능 비교

표 1은 기존에 제안된 GCMS 계열의 방법을 Aurora2 데이터베이스에 적용한 실험결과를 보여주며, 정규화의 차수가 고차로 갈수록 성능이 더 좋아지는 것을 알 수 있다. 추가 실험으로서 CTN 방법에 smoothing 필터를 결합하였을 경우 CTN 방법에 비해 4.81%의 성능 향상율이 있었다. 여러 가지 smoothing 필터 중 아래와 같은 moving average를 이용한 경우가 성능이 가장 좋았다. 본 논문에서는 M의 크기는 4를 사용했다.

$$\widehat{x}_t = \frac{1}{2M+1} \sum_{m=-M}^{M} x_{t+m} \qquad (12)$$

표 1. 정규화 방법에 따른 성능 비교

전처리 방법	clean condition				
	set A	set B	set C	Ave	ERR
Baseline	61 34	55 75	66 14	60 06	0 00%
GCMS	66 36	71 43	67 20	68 55	21.26%
GCVN	75 08	75 92	76 38	75 68	39 09%
GCTN	80 71	82 32	81 32	81 48	53 62%
GCVN+smoothing filter	81 33	81 62	82 31	81.64	54 03%
GCTN+smoothing filter	83 12	83 73	83 28	83 40	58 43%

그림 3은 윈도우 사이즈에 따른 LCMS의 인식률이다. LCMS 방법의 경우에 윈도우 사이즈가 커질수록 성능이 올라가는 특성을 보여주고 있다. 윈도우 사이즈를 410ms를 사용했을 경우의 성능이 가장 우수하므로 이후 실험에서는 410ms의 윈도우를 사용했다.

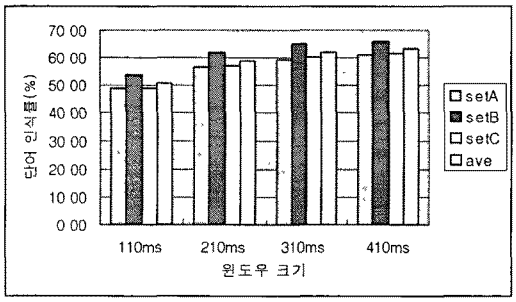

그림 3. 윈도우 사이즈에 따른 LCMS 인식률

410ms의 윈도우 사이즈에서 SCMS 방법과 LCMS 방법을 비교한 경우 SCMS 방법이 LCMS 방법에 비해 성능이 우수하다. 하지만 SCMS는 고차통계정규화 방법을 적용하기가 어렵기 때문에, LCTN의 실시간 구현은 LCMS 방법을 기반으로 결과를 얻었다. LCTN 방법을 사용하였을 때 baseline에 비해 38.06%의 오류감소율을 얻었다. 또한 outlier에 의한 왜곡을 감소시키기 위해 smoothing 필터를 사용하여 44.70%의 오류감소성능을 얻었다. 이는 전체 음성에 대한 보상을 실시하는 GCTN에 비해 성능이 떨어지지만, GCMS 방법과 GCVN 방법보다는 성능 면에서 더 우수한 결과이다.

표 2. 실시간 전처리 방법의 인식률 비교

실시간 전처리 방법	clean condition				
	set A	set B	set C	Ave	ERR
Baseline	61 34	55 75	66 14	60 06	0 00%
SCMS	61 41	66 68	62 13	63 66	9 01%
LCMS	60 77	66.14	61 23	63 01	7 38%
LCTN	73 90	76 43	75 68	75 26	38 06%
LCTN+smoothing filter	76 89	78 73	78 33	77 91	44 70%

V. 결론

본 논문에서는 CMS 방법과 고차통계방법인 CTN 방법에 대해 살펴보고 잡음보상의 실시간 처리를 위해 LCMS 방법을 CTN 방법에 적용하였다. 또한 outlier에 의한 왜곡을 감소시키기 위해 smoothing filter를 사용하였다. Aurora2 데이터베이스의 clean condition에 대해 실험을 해본 결과 실시간 처리가 가능하면서도 Aurora 2 데이터베이스의 baseline 시스템에 비해 44.70%의 성능 향상율을 보였다.

참고문헌

[1] O. Viikki and K. Laurila, "Cepstral domain segmental feature vector normalization for noise robust speech recognition," Speech Communication, Vol. 25, pp. 133-147, Aug. 1998

[2] Y. H. Suk, S. H. Choi and H. S. Lee, "Cepstrum third order normalization method for noisy speech recognition," Electronic Letters, Vol. 35, no. 7, pp 527-528, Apr. 1999.

[3] C.-W. Hsu and L.-S. Lee, "Higher order cepstral moment normalization for robust speech recognition," ICASSP, Vol. 1, pp. 197-200, May 2004

[4] H. G. Hirsch and D. Pearce, "The Aurora experimental framework for the performance evaluations of speech recognition systems under noisy conditions," ISCA ITRW ASR2000, Paris, Sep. 2000.