가변프레임 길이정규화를 이용한 단어음성인식

Isolated-Word Speech Recognition using Variable-Frame Length Normalization

*신 찬 훈(Shin, C. H.)
이 회 정(Lee, H. J.)
박 병 철(Park, B. C.)

요 약

단어음성인식에서 발성속도의 차이에 따른 단어음성 길이의 비선형적 변화는 정확한 인식을 어렵게 하는 주요한 원인이 되어 왔다. DP매칭은 시간축의 비선형 신축에 의해 시간정규화를 행함으로써 인식결과에 대한 신뢰성을 상당히 높였으나 시간정규화 과정에 요구되는 과도한 계산부담이 문제로 되어 있다.

본 논문에서는 시간정규화가 필요없는 방법으로 멀티섹션벡터양자화에 새로운 길이정규화법을 적용하는 방법을 제안한다. 이 방법은 종래의 고정프레임 길이정규화에 의해 멀티섹션코드북을 작성할 때보다, 정규화길이의 설정에 훨씬 융통성을 가질 수 있으므로 분석 및 거리계산의 양 면에서 시간 단축을 가능케 하여 좀더 신속히 인식결과를 얻을 수 있는 장점이 있다.

ABSTRACT

Length normalization by variable frame size is proposed as a novel approach to length normalization to solve the problem that the length variation of spoken word results in a lowing of recognition accuracy.

*성균관대학교 공과대학 전자공학과

This method has the advantage of curtailment of recognition time in the recognition stage because it can reduce the number of frames constructing a word compared with length normalization by a fixed frame size. In this paper, variable frame length normalization is applied to multisection vector quantization and the efficiency of this method is estimated in the view of recognition time and accuracy through practical recognition experiments.

Ⅰ. 서 론

단어음성인식에서는 발성속도에 따른 시간변동의 제거를 위하여 DP매칭이 많이 이용되고 있다[1]. 그러나 이 방법은 시간축의 선형변환에 의해 시간정규화를 행하는 것에 비해 계산량이 6～8배나 증가한다[2].

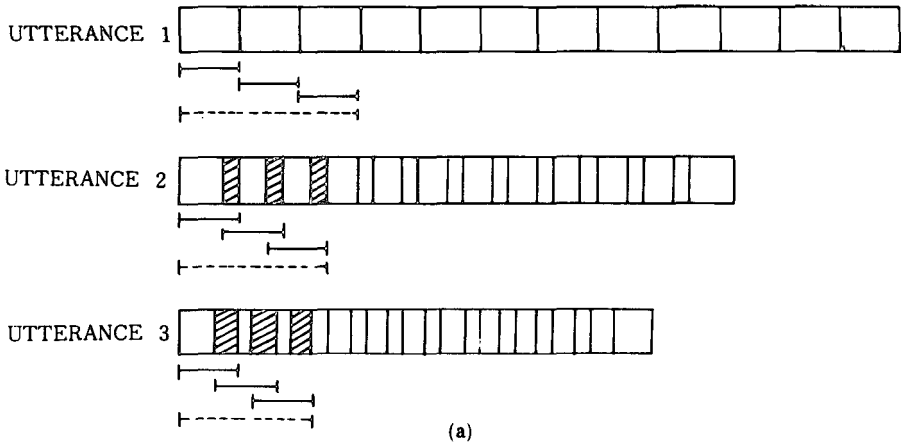
이밖에 시간정규화가 필요없이 단어별로 작성된 벡터양자화(VQ) 코드북에 의해 단어들의 음향적인 특성만을 비교하는 방법이 있으나, 이러한 코드북에는 시간적 정보가 포함되어 있지 않으므로 음향적 특성이 유사한 단어들 사이에서 부정확한 인식이 일어나기 쉽다[3]. 따라서 한 단어를 발성순서에 따라 몇 개의 섹션으로 분할하고 섹션별로 독립된 코드북을 작성함으로써 시간적 정보를 포함시키는 멀티섹션(MS)코드북이 Burton등에 의해 제안되었다[4][5]. Burton의 멀티섹션벡터양자화(MS VQ)에서는 MS코드북 작성에 이용되는 모든 음성은 발성시간에 관계없이 일정한 수의 정해진 길이를 갖는 프레임으로 정규화되어야 한다. 따라서 발성시간이 짧은 단어는 정규화 길이에 일치시키기 위해서 인접프레임들을 중첩시켜야 한다. 이것은 한 단어를 구성하는 프레임 수를 불필요하게 증가시킬 수 있으므로 분석과 거리계산의 시간을 증가시키는 요인이 될 경우가 많다.

본 논문에서는 프레임길이를 그 단어의 길이에 따라 변하게 함으로써 정규화길이를 설정하는데 융통성을 가질 수 있는 가변프레임에 의한 길이정규화법을 제안하고, 콤퓨터 시뮬레이션을 통해 이 방법의 유효성에 대한 평가를 시도한다.

Ⅱ. 가변프레임 길이정규화에 의한 MS코드북의 작성

VQ 코드북의 sequence로써 시간정보(time-sequence information)를 포함시키는 방법을 MS 코드북이라고 하는데, 이것은 인식성능을 개선하고 계산의 복잡성을 경감시키는 효율적인 방법으로 알려져 있다.

어떤 단어의 MS코드북은 그 단어를 동일길이의 섹션으로 나누고, 각 섹션마다 LBG알고리듬을 적용하여 작성한다. 이 때 각 섹션은 동일한 길이를 갖는 일정한 수의 프레임으로 구성된다. MS코드북과 표준벡터양자화 코드북 사이의 근본적인 차이는 MS코드북의 경우는 섹션별로 코드북을 작성해야 하므로 training에 사용될 모든 음성은 사전에 일정한 길이로 정규화해 두어야 한다는 점이다. Burton 등은 모든 training sequence를 동일한 수의 프레임으로 길이정규화시키는 방법으로 높은 인식률을 얻는데 성공하였다[4]. 여기서는 한 프레임의 길이가 고정되어 있으므로 편의상 고정프레임 길이정규화라고 부르기로 한다. 고정프레임 길이정규화는 가장 길게 발성된 음성에 맞추어 정규화길이가 정해지므로 그림 1 (a)와 같은 세 음성데이터가 있을 때 가장 긴 UTTERANCE 1을 기준으로 하여 고정된 길이의 프레임으로 나누어 얻은 총 프레임



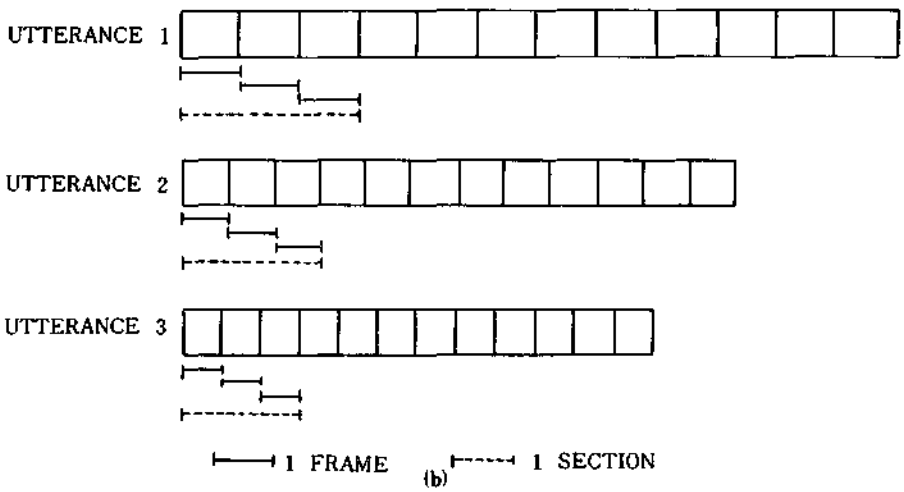


그림 1 Training sequence의 길이정규화
(a) 고정프레임 길이정규화
(b) 가변프레임 길이정규화
Length normalization for training sequences
(a) Fixed-frame length normalization
(b) Variable-frame length normalization.

수를 정규화길이로 한다. 그림에서는 정규화길이를 12프레임으로 한 경우의 예를 설명하고 있다. 첫번 음성보다 상대적으로 짧은 두, 세번째 음성도 12프레임으로 길이를 정규화시키기 위해 이웃 프레임끼리 중첩을 시켜 주어야 한다. 빗금친 부분은 첫번 째 섹션에 대한 중첩부분만을 표시한 것이다. 이와같이 중첩하는 부분이 많아지는 것은 LPC 분석과 거리계산의 시간을 증가시키게 되므로 인식속도의 개선이란 측면에서도 바람직하지 않다. 중첩이 일어나는 것을 회피하기 위하여 본 논문에서는 음성데이터의 길이에 따라 프레임길이를 다르게 하는 방법에 대하여 고찰한다. 이것은 프레임길이가 고정되어 있지 않으므로 가변프레임 길이정규화라고 할 수 있다. 그림 1 (b)는 가변프레임에 의한 길

이정규화법을 설명하기 위한 것이다. 각 음성데이터의 길이를 정해진 프레임 수 즉 정규화길이로 나누어 한 프레임의 길이를 정하게 된다. 정규화길이는 단위 프레임이 극단적으로 길어져서 음성정보 자체를 손상시키는 일이 없도록 적절하게 선택해야 한다.

가변프레임 길이정규화를 이용하여 MS코드북을 작성하는 과정을 그림 2에서 설명하고 있다. 한 단어 W를 1회 발성한 음성을 training sequence로 이용한다. 한 프레임을 LPC분석하여 얻은 특징벡터를 $\bar{v}$ 라고 하면 1회 발성 음성은

$$W=(\bar{v}_1\ \bar{v}_2\ \cdots\ \bar{v}_K) \tag{1}$$

와 같이 $\bar{v}$의 sequence로 표시할 수 있다. 여기서 K는 단어 W의 정규화길이를 나타낸다. 그림 2에서 한 단어의 정규화길이는 12프레임(K=12)이다.

인식대상어휘가 모두 L개의 단어로 되어 있을 때, 각 단어마다 I회 발성된 음성을 이용하여 MS코드북을 구성하기 위하여 우선 이들을 J개의 섹션으로 분할한다.

$$W^l(i)=[V_1(i)\ V_2(i)\cdots V_J(i)]^t$$
$$(\ell=1,\ 2,\ \cdots,\ L) \tag{2}$$

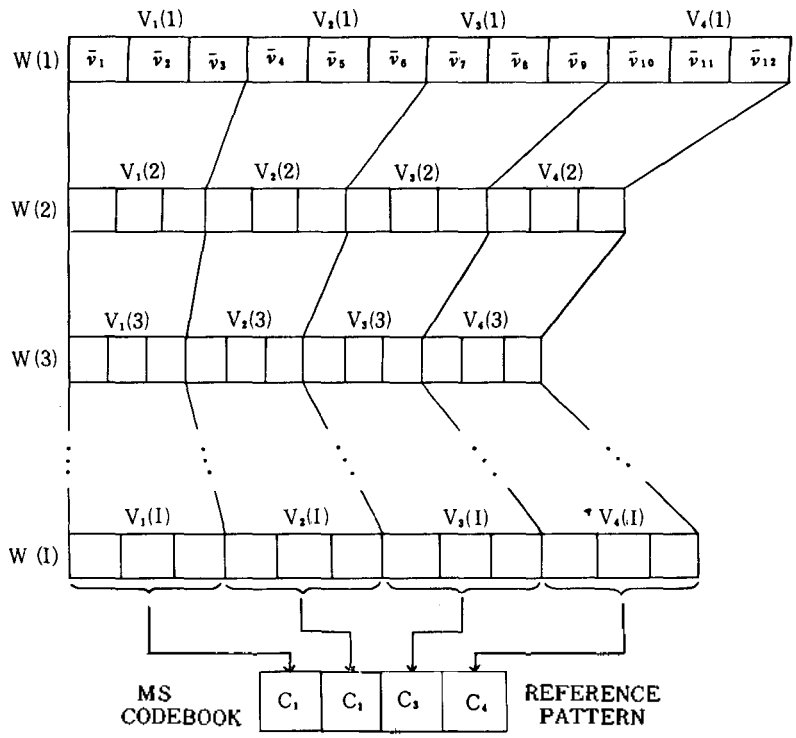


그림 2 가변프레임 길이정규화를 이용한 MS코드북의 작성
MS codebook design by using variable-frame length normalization.

여기서 $V_j^{\ell}(i)$은 어떤 단어 ℓ의 한 샘플 i에 대한 j번째 세션의 벡터 sequence를 나타낸다. 따라서 한 세션이 N프레임으로 구성되어 있다면

$$V_j(i) = [\vec{v}_1(i)\ \vec{v}_2(i) \cdots \vec{v}_N(i)] \tag{3}$$

$$V_j(i) = [\vec{v}_{(j-1)N+1}(i)\ \vec{v}_{(j-1)N+2}(i) \cdots \vec{v}_{jN}(i)]$$

와 같이 각 세션을 벡터sequence로 표시할 수 있다. 그림2는 한 세션당 3프레임씩 4세션으로 되어있으므로 4개의 독립적인 VQ코드북의 조합에 의해 MS코드북이 구성된다. 세션j에 해당하는 training sequence의 집합을 Sj라고 하면

$$S_j = \{V_j(1),\ V_j(2),\ \cdots V_j(I)\}$$

$$(j = 1,\ 2,\ 3,\ 4) \tag{4}$$

이 된다. 각 세션에 대한 코드북 C_j는 S_j를 traing sequence로 하여 LBG알고리듬[6]을 이용하여 작성한다. 이 과정을 통해 작성된 코드북의 sequence 즉,

$$C = \{C_1\ C_2\ C_3\ C_4\} \tag{5}$$

는 MS코드북을 의미한다. 각 세션코드북 C_j는 2^R개의 codeword로 이루어지는데 이때 R을 codebook rate라고 한다.

MS코드북은 한 단어를 몇 개의 세션으로 나누어 코드북을 작성하기 때문에 동일 코드북에 포함된 특징벡터들 사이에는 유사성이 많으므로 codeword의 수를 많이 할 필요가 없을 뿐 아니라, 불특정화자에 대해서도 하나의 표준패턴으로 상당히 높은 인식률을 얻을 수 있다.

Ⅲ. 분석구간변화에 따른 효과 및 거리척도

가변프레임을 이용하여 특징벡터를 추출하면 발성속도차이에서 오는 입력음성의 특징변화를 고루 포함할 수 있으며 이들 특징벡터의 분포공간에 대한 *cluster*의 최적한 대표값을 선택해 놓음으로써 인식과정에서 화자의 발생속도변화에 따른 음성특징의 변화에 잘 적응할 수 있다. 이로써 입력 음성에 대한 시간정규화과정은 필요없게 된다.

본 논문에서는 거리척도로서 LPC 켑스트럼을 이용한다. 이것은 거리척도로서의 조건을 잘 만족할 뿐 아니라, 높은 인식률을 얻을 수 있는 것으로 알려져 있다.

LPC켑스트럼거리 d^2는

$$d^2 \equiv (C_0 - C_0')^2 + 2\sum_{k=1}^{p}(C_k - C_k')^2 \tag{6}$$

로 정의된다[7],[8]. 여기서 C_k, C_k'은 각각 표준패턴과 시험입력음성에 대한 LPC켑스트럼계수를 나타내며, C_0, C_0'은 두 음성패턴의 에너지가 된다. 이때 차수p는 음성의 LPC모델에 의한 차수이다.

Ⅳ. 가변프레임 길이정규화를 이용한 단어음성인식

인식하고자 하는 시험입력음성 W_x는 먼저 가변프레임 길이정규화법에 따라 정해진 수 즉 NJ개의 프레임으로 정규화된 후, 각 프레임은 J개의 세션에 할당된다. 이 결과를 벡터 sequence로 나타내면 W_x는

$$W_x = \{X_1\ X_2\ \cdots X_J\} \tag{7}$$

가 된다. X_j는 j 번째 세션을 구성하는 프레임들로부터 LPC분석을 통해 구한 특징벡터의 sequence를 나타내는 것이다. 따라서 X_j는

$$X_j = (\tilde{x}_{(j-1)N+1}\ \tilde{x}_{(j-1)N+2} \cdots \tilde{x}_{jN}) \quad (8)$$

로 표시된다. 이들 각 세션에 대한 특징벡터들을 표준패턴의 상대 세션코드북의 codeword들과 거리비교를 통해 전체 평균거리 D_{av}를 구한다. 어떤 단어 ℓ에 대한 표준패턴과의 전체 평균거리 D^{ℓ}_{av}은

$$D^{\ell}_{av} = \frac{1}{N}\sum_{j=1}^{J} d_j(X_j,\ C_j) \quad (9)$$

로 나타낼 수 있다. 여기서 d_j는

$$d_j = \sum_{i=(j-1)N+1}^{jN} \min_{r} d(\tilde{x}_i, c^{j}_{jr})\ (r = 0,\ 1, \cdots \cdots, 2^R) \quad (10)$$

이다. 식(10)에서 c^{j}_{jr}은 단어 ℓ의 j 번째 세션코드북의 한 codeword를 나타낸다.

이상의 과정을 모든 단어의 표준패턴에 대하여 반복하여 최종적으로 전체 평균거리가 최소인 단어 즉,

$$l^* = \arg\min_{l} D^{l}_{av} \quad (11)$$

인 단어 W^{l^*}을 W_x와 동일한 단어로 판정한다.

V. 실험 및 고찰

1. 실험조건

데이터베이스는 20명의 남성화자에 의해 발성된 10개 도시명으로 하였다. 이들 도시명은 음향적 특성이 유사하여 혼동의 우려가 큰 경우와 그렇지 않은 경우에 해당하는 것으로 적절히 선택하였다. 음성샘플의 수집은 방음장치가 되어 있지 않은 실험실에서 마이크로폰으로부터 직접 AD변환기를 거쳐 IBM PC/XT의 하드디스크에 저장하는 방법을 사용하였다. 마이크로폰을 통해 입력된 음성은 샘플링에 앞서 차단주파수가 4 KHz인 LPF에 의해 대역을 제한하였다.

표 1. 실험 조건

1. 발성 단어(10종류)
서울, 수원, 대전, 대구, 부산,
마산, 광주, 인천, 강릉, 목포
2. 발성자수 : 남성화자 20명
3. 발성횟수 : 각 단어 2회
4. 분석조건
샘플링주파수—10KHz
A/D변환정도— 8 Bit
LPF —4 KHz
window-rectangular window
프레임길이—가변
분석파라미터—LPC켑스트럼
분석차수—10차

저장된 음성은 다시 정확한 음성구간만을 추출하여 단어별로 다시 저장하였다.

본 논문에서 공통적으로 이용된 실험조건은 표 1과 같다.

2. 인식실험

인식실험은 두 단계로 이루어진다. 우선 training을 통해 인식대상인 모든 단어에 대한 표준패턴 즉 MS코드북을 작성한다. 인식단계에서는 시험음성을 시스템에 입력시켜 그 결과를 관찰한다.

그림 3은 실험에 사용된 시스템의 블럭도이다. training단계에서는 데이타베이스에 저장된 음성을 training sequence로 하여 Ⅱ장에서 설명한 방법에

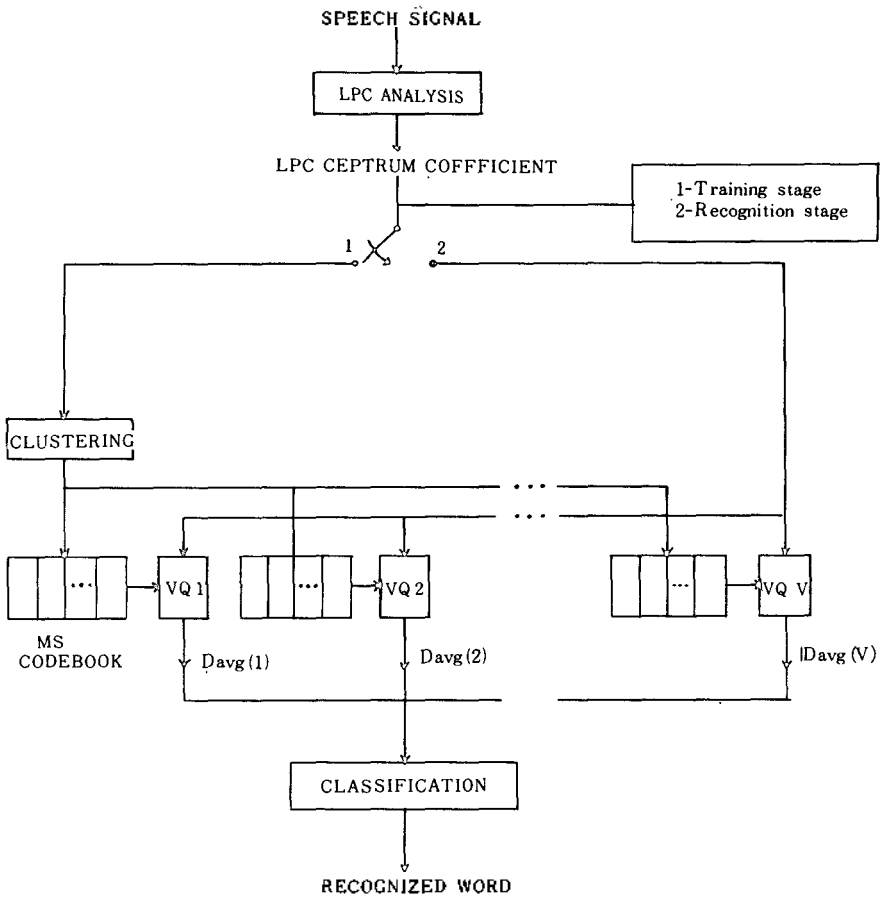


그림 3 전체 인식시스템의 블럭도
Block diagram for overall system.

따라, 각 프레임에 대한 특징벡터로서는 10차의 LPC켑스트럼계수를 이용하고 단어음성의 정규화길이는 24프레임, 세션수는 6으로 하여 모든 단어의 MS코드북을 작성한다.

인식절차는 입력된 미지의 시험음성을 모든 인식대상단어의 MS코드북과 비교하여 평균왜곡이 최소가 되는 단어를 선택함으로써 완료된다.

인식과정에서는 직접 음성을 입력시키는 대신 특징벡터의 sequence로 각 단어의 패턴을 만들어 표준패턴과 함께 메모리에 저장한 후 거리비교를 행하였다.

3. 실험결과 및 검토

실험은 특정화자와 불특정화자의 경우로 나누어 실행하였다.

특정화자의 인식실험에서는 데이터베이스 내의 모든 단어를 이용하여 MS코드북을 작성하고, 이것을 다시 시험음성으로 사용하여 인식결과를 조사하였다.

불특정화자에 대한 인식실험에서는 20명의 화자중 15명의 음성을 MS코드북 작성에 사용하고 나머지 5명에 대한 음성을 시험음성으로 이용하였다.

두 경우 모두 한 단어는 6세션 24프레임으로 길

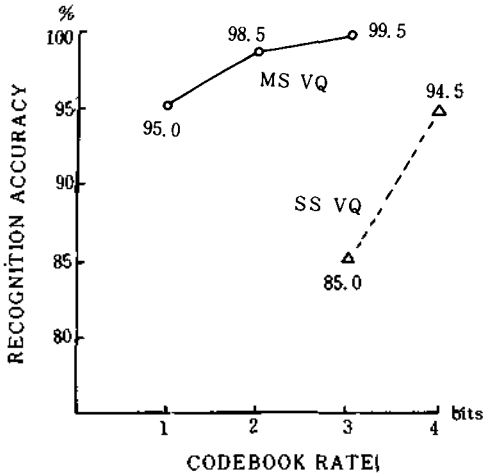


그림 4 특정화자에 대한 음성인식실험 결과
Experimental results for speaker-dependent word speech recognition.

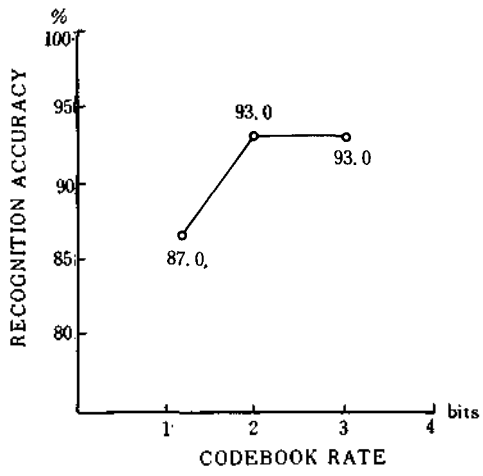


그림 5 불특정화자에 대한 음성인식실험결과
Experimental results for speaker-independent word speech recognition.

이를 정규화하였다.

특정화자 음성인식 실험은 MS코드북의 codebook rate가 1, 2, 그리고 3비트인 세가지 경우에 대해서 이루어졌다. 이 결과 얻어진 인식률은 그림 4와 같다. rate 3인 MS코드북의 인식률은 동일조건에서의 단일섹션(SS)코드북에 비해 14.5%가 높았다. 이때 계산량은 둘다 동일하다. 이것은 rate 4인 SS코드북보다도 5%가 높은 것이지만, 계산량은 그것의 $\frac{1}{2}$에 불과하다.

불특정화자에 대한 인식률은 그림 5에 나타난 것처럼 다소 저하되었다. 일반적으로 codebook rate의 증가는 인식률을 향상시키지만, 이 경우 93%가상은 향상되지 않았다. 이와같은 저조한 인식률은 training과정에서 이용된 음성데이터가 너무 적어서 불특정 다수의 다양한 음성특성을 충분히 나타낼수 있는 보편성 있는 표준패턴이 만들어지지 않은 것이 큰 요인이라고 생각된다.

정규화길이가 24프레임이고 섹션수가 6인 가변프레임방법과 정규화길이가 36프레임이고 섹션수를 6으로 한 고정프레임방법에 대해 각각 코드북레이드를 2, 3으로 한 MS코드북을 이용하여 인식한 결과를 표 2에서 비교하였다. 고정프레임의 경우 한 프레임의 길이를 20msec로 하였다.

특정화자의 경우 가변프레임방법에서는 코드북레이트가 2일 때 98.5%의 높은 인식률을 나타냈으나, 고정프레임방법의 경우 이보다 3.5% 낮은 인식률을 보였다.

불특정화자의 경우는 코드북레이트가 3일 때 고정

표 2. 고정프레임 길이정규화와 가변프레임 길이정규화의 성능 비교(인식률 %)

방식 / 인식대상	코드북레이트	고정프레임방법 (20msec프레임)	가변프레임방법
특정화자	2	95.0	98.5
불특정화자	3	93.5	93.0

프레임방법이 가변프레임방법보다 인식률이 0.5 % 더 높게 나타났다. 위 결과에서 알 수 있듯이 정규화 길이가 고정프레임방법보다 12프레임 더 짧은 가변프레임방법이 더 적은 계산량을 갖고도 특정화자의 경우 인식률이 더 높고 불특정화자의 경우에도 거의 비슷한 결과를 얻을 수 있었는데 이미 언급한 바와 같이 가변프레임을 이용하여 특징벡터를 추출하기 때문에 발성속도차이에서 오는 음성특징변화를 고정프레임을 이용하는 경우보다 더 다양하게 포함할 수 있어 더 짧은 정규화길이로도 더 좋은 인식성능을 나타내는 것으로 생각된다.

4. 계산부담의 경감효과

MS VQ에 의한 음성인식에서 시험음성과 표준패턴사이의 거리계산량은 섹션코드북당 codeword 수 Nsc와 정규화길이 L_N의 곱인 $N_{sc}L_N$이 된다. 본 논문에서 사용한 음성데이터의 길이는 0.36sec에서 0.72sec사이에 존재하고 평균길이는 0.516sec였다. 한 프레임을 20msec로 하는 고정프레임 길이정규화법에서는 샘플링주파수가 10KHz이므로 7200/200즉 36프레임이 정규화길이가 된다. 그러나 가변프레임을 사용하면 평균길이에 맞추어 길이정규화가 이루어지므로 5160/200 즉 25.8 프레임이 된다. 따라서 고정프레임 길이정규화에 비해 약 10프레임정도 길이를 줄일 수 있다. 이것은 표준패턴 하나에 10Nsc 만큼의 거리계산을 줄일 수 있음을 뜻한다. 거리계산은 표준패턴이 증가하면 따라서 증가하므로 대어휘의 인식을 위해 가변프레임을 사용하면 더욱 효과적이다.

Ⅵ. 결 론

MS VQ에서의 새로운 길이정규화법으로 가변프레임에 의한 길이정규화를 제안하고, 컴퓨터 시뮬레이션을 통해 그 효율성을 평가하였다.

특정화자의 음성인식에 있어서는 인식률과 처리속도면에서 모두 만족한 결과가 얻어졌다. 그러나 불특정화자의 경우에서는 만족한 인식률을 얻어지지 않았다. 이것은 불특정화자의 음성을 인식하기 위해서는 모든 화자의 음성을 충실하게 표현할 수 있는 보편성있는 표준패턴이 만들어져야 하는데 training에 이용된 음성데이터의 양이 너무 적어, 모든 음성특성을 충분히 포함하는 코드북이 만들어지지 않았기 때문이다. 그러나 이 경우도 표준패턴은 한 단어에 하나만 준비하면 되므로 multi-template 방식에 비해 상당한 시간단축이 가능하다.

이 방법이 불특정화자에 대해서도 좀더 실용성있게 되기 위해서는 충분한 음성데이터에 의해 training을 행하는 문제와 아울러 가능한 한 적은 데이터를 이용하여 모든 화자의 음성특성을 잘 포함할 수 있도록 효율적으로 MS코드북을 작성하는 방법에 대한 연구가 좀더 진행되어야 할 것이다.

또한 본 논문에서 취급한 모든 단어는 두 음절로 되어 있지만, 음절이 다른 단어들이 섞여 있는 경우 적절한 인식률을 유지하기 위한 한 분석구간의 길이는 제한적일 수 밖에 없으므로 최적한 정규화길이를 구하는 것이 요구된다. 이를 위해서는 분석구간의 길이변화에 따른 주파수분해능의 영향에 관해 더 많은 기초연구가 필요할 것이다.

참 고 문 헌

1. H. Sakoe and S. Chiba," Dynamic programming algorithm optimization for spoken word recognition," IEEE Trans. Acous, Speech and Sig. Process., ASSP-26, No.1, pp.43-49, Feb. 1978.
2. D. O'Shaughnessy, "Speaker recognition," IEEE ASSP Magazine, pp.4-17, Oct. 1985.
3. J.E. Shore and D.K. Burton," Discrete utterance speech recognition without time alignment," IEEE Trans. Inform. Theory, IT-29, No.4, pp. 473-491, July 1983.

4. D.K. Burton, J.E. Shore, and J.T. Buck, "Isolated-word speech recognition using multisection vector quantization codebooks, "IEEE Trans. Acous., Speech, and Sig. Process., ASSP-33, No.4, pp. 837-849, Aug. 1985.
5. D.K. Burton and J.E. Shore, "Speaker-dependent isolated word recognition using speaker-independent vector quantization codebooks augmented with speaker-specific data," IEEE Trans. Acous., Speech, and Sig. Process., ASSP-33, No.2, pp. 440-443, Apr. 1985.
6. Y. Linde, A. Buzo, and R.M. Gray, "An algorithm for vector quantizer design," IEEE Trans. Commun., COM-28, No.1, pp.84-95, Jan. 1980.
7. A.H. Gray, JR. and J.D. Markel, "Distance measures for speech processing," IEEE Trans. Acous., Speech, and Sig. Process., ASSP-24, No.5, pp. 380-391, Oct. 1976.
8. R.M. Gray, A. Buzo, A.H. Gray, JR., and Y. Matsuyama, "Distortion measures for speech processing, "IEEE Trans. Acous., Speech, and Sig. Process., ASSP-28, No.4, pp.367-376, Aug. 1980.