

LPC 거리를 이용한 영상 Registration

(Image Registration Using an LPC Distance)

李 炘 武,* 李 商 郁*

(Kyung Moo Lee and Sang Uk Lee)

要 約

주어진 영상내에서 원하는 물체의 위치를 찾아내는 registration을 위하여 1차원 LPC를 이용한 새로운 알고리즘을 제안하였다. 제안된 알고리즘은 1차원 LPC를 이용하여 영상을 모델링하고, LPC계수 벡터들을 feature로 사용하며, Itakura가 제안한 LPC 거리측정법을 이용하여 feature간의 근사도를 계산함으로써 두 영상을 비교하였다. 제안된 알고리즘이 normalized 상관계수방식, template matching 방식에 비해 search 영상내에 reference물체와 비슷한 물체가 여럿있는 경우, 또 영상 intensity 변화가 있을 경우에 비교적 좋은 결과를 나타냄을 컴퓨터 시뮬레이션을 통하여 보였고, 아울러 계산량면에서도 normalized상관계수방식 보다 훨씬 적은 계산량이 필요함을 보였다

Abstract

For the registration problem in which the matching of two images is made, a new algorithm using an 1-D LPC model was proposed. The proposed algorithm employed LPC coefficients as feature vector of an image. The similarity of two images was measured using an LPC distance, proposed by Itakura, between each image's feature vector. The comparison of performance with normalized correlation method and template matching method was made by a computer simulation with several real images. The results of simulation showed that the proposed algorithm was more robust to image intensity variation and computationally efficient.

I. 서 론

영상처리와 패턴인식 분야에서 두 영상의 화소와 화소를 비교하여 다음과 같은 정보를 얻어내는 것이 필요한 경우가 많다.

(1) 각기 다른 센서로부터 얻은 두 개 또는 그 이상의 영상으로부터 동일한 물체에 대한 특징을 추출할 경우

(2) 한 물체에 대해 서로 다른 시간에서 얻은 두 영상으로부터 그 물체의 움직임을 추정할 경우

(3) 한 물체에 대해 서로 다른 위치로부터 얻은 두 영상을 이용하여 카메라로부터의 거리와 3차원적 정보를 얻을 경우

(4) 한 영상내에 주어진 어떤 물체의 위치를 찾아내는 경우

즉 registration은 이렇게 두 영상간의 상대적 이동 정보 또는 각 영상들의 기하학적 변형, 밝기의 변형 등을 알아내고 보정하는 것으로서 영상처리 분야에서 가장 기본이 되는 문제중의 하나이다.

Registration 문제중에서 가장 기본적이고 현재 활발히 연구중인 것으로는 주어진 큰 search 영상내에서 작은 reference 영상과 가장 유사한 부영상의 위

*正會員, 서울대학교 制御計測工學科
(Dept. of Control and Instrumentation Eng., Seoul Nat'l Univ.)

接受日字: 1986年 4月 3日

(* 본 연구는 한국과학재단의 차관연구비 지원으로 이루어진 "자동목표 추적장치에 관한 연구"의 일부임.)

치를 찾아내는 것이다. 이러한 연구는 1950년 경부터 문자인식에 응용되기도 하였고, 현재는 위성사진의 분석, 로보트 비전, 영상추적장치등 여러 분야에 응용되고 있다. 기존의 registration 알고리즘은 크게 다음의 세가지로 나눌 수 있다.

- ① 상관(correlation) 계수 방식 [1-3]
- ② Template matching 방식 [3-6]
- ③ Feature matching 방식 [7, 8]

일반적으로 reference 영상과 search 영상에서 취한 같은 크기의 영상, 즉 window영상을 비교하는데 있어, 상관계수방식은 대응하는 화소들의 곱의 합을 상관계수로 정의하여 이 값을 최대로 하는 search 영상내의 window 영상의 위치를 찾아나가는 방식을 말하는데, 보통이 값이 첨예한 peak치를 갖지 못하므로 reference 영상과 window 영상의 에너지 값을 이용하는 normalized 상관계수 방식이 널리 이용되고 있다. normalized 상관계수 방식은 이론적으로 가장 좋은 특성을 가지고 있으나 계산량이 엄청나게 많이 필요하다는 큰 단점이다. 따라서 두 화소들의 곱대신 절대오차의 합을 두 영상간의 오차로 정의하여 이 오차가 최소가 되는 점을 찾아가는 template matching 알고리즘이 많이 사용되고 있다. Template matching 방식은 계산에서 곱셈이 필요치 않다는 잇점은 있으나 성능면에서 normalized 상관계수 방식에 비해 정확치 못한 것이 단점이다. Feature matching 방식은 최근에 제안되고 있는 기법으로서, 영상으로부터 적절한 feature를 추출하여 두 features들의 근사성(similarity)을 계산하므로써 두 영상을 서로 비교하는 방법이다. 따라서 feature matching 방식은 normalized 상관계수 방법과 template matching 방식의 정확도와 계산량에서 절충안이라고 할 수 있다. 그러므로 feature matching 방식은 먼저 영상의 특징을 잘 나타내는 feature의 선정과 또한 두 features들간의 근사성을 잘 표현할 수 있는 적절한 metric의 선정이 가장 큰 문제라고 할 수 있다. B. Rezaie등은^[7] 화소의 값을 그 주위의 8개의 화소의 선형결합으로 나타내는 2차원 모델링을 한 후 다시 1차원 DST(Discrete Sine Transformation)를 이용하여 수평 방향으로 decorrelation시켜 계수가 하나인 interpolative 모델로 변환시킨 다음, 이 계수를 PARCOR(PARTIAL CORrelation) 방식을 이용하여 구하고, 각각의 계수를 서로 비교하는 새로운 알고리즘을 제안하였다. 그러나 이 방식은 모델링 자체가 너무 특수하고 화소수에 비해 모델링 계수가 적으므로 정확도에서 normalized 상관계수 방식에 비해 좋지 않다. 또한 두 영상간의 근사성을 계산하는데 계수 벡터들간의 차(difference)를 이용하므로써 문턱값 설정에

어려움이 따른다. 즉 정확한 위치에서의 값과 그 주위에서의 값들이 서로 비슷한 값들을 가지게 되는데 이것은 근사성 계산에서 계수 벡터들간의 차를 이용하였기 때문이다. 한편 Ranganath등은^[8] 영상내의 각 행과 열에 대해 인접한 화소들의 상관계수를 영상의 feature로 사용하여 두 영상을 비교하였다. 이 경우도 recursive한 방법으로 계산량은 상당히 감소시켰으나 normalized 상관계수 방식에 비해 정확도면에서 좋지 않다. 일반적으로 기존의 알고리즘들은 search 영상내에 reference 영상과 에너지가 비슷하거나 크기가 비슷한 영상들이 여러 존재할 때 구별이 어렵고 또한 영상의 intensity 변화, 즉 scale 또는 bias에 대해 영향을 많이 받으므로 정확한 문턱값 설정이 어렵다. 따라서 본 논문에서는 영상을 1 또는 2차원 LPC(Linear Predictive Coding)를 이용하여 모델링 하고 LPC계수들을 feature로 사용하는 새로운 registration 알고리즘을 제안하였다. 두 LPC계수들간의 근사성 측정은 음성인식에서 많이 사용되고 있는 LPC거리측정법^[9]을 사용하였다. 따라서 본 논문에서는 LPC거리측정법이 음성뿐만 아니라 영상신호에서도 두 영상을 비교하는데 대단히 효과적이라는 것을 실제 영상을 이용한 컴퓨터 시뮬레이션을 통해 입증하였다. 본 논문에서 제안한 registration 알고리즘의 정확성은 normalized 상관계수 방식과 비교하여 검토하였으며 실제 계산량도 많이 감소함을 보였다.

II. 영상모델링

1. 영상모델링 방법

2차원 영상을 random field적인 측면에서 모델링하는데는 여러가지 방법이 있다. 먼저 연속하는 행 또는 열들을 서로 연결하여 2차원의 영상을 1차원의 영상 데이터로 변환시킨 후, 이 1차원의 영상을 모델링하는 데는 다음의 두가지 방법이 있다.^[10]

Autoregressive model

$$X(k) = \sum_{j=1}^P \alpha_j [X(k-j)] + \epsilon(k) \quad (1)$$

noncausal interpolative model

$$X(k) = \sum_{j=1}^P \alpha_j [X(k+j) + X(k-j)] + \epsilon(k) \quad (2)$$

여기서

$X(k)$ 는 k 번째 화소의 값

$\epsilon(k)$ 는 모델링 오차

α_j 는 모델링 계수

P 는 모델의 차수

를 나타낸다.

Autoregressive 모델은 all-pole 모델로서 전달함수

가 $1/(1 - \sum_{k=1}^P \alpha_k z^{-k})$ 인 IIR 필터로 생각할 수 있다. 또한 이 필터의 출력을 $X(k)$ 라고 가정하면 $X(k)$ 는 식(1)과 같이 표시되며 입력신호 $\epsilon(k)$ 는 백색잡음원이다. 또한 영상을 2 차원적으로 모델링하면 식(3)과 같이 나타낼 수 있다.

$$X(k, j) = \sum_{m,n \in R} \alpha_{m,n} X(k-m, j-n) + w(k, j) \quad (3)$$

여기서

R은 상관구역

$N(k, j)$ 는 모델링 오차

$\alpha_{m,n}$ 은 모델링 계수

를 나타낸다.

또한, 2 차원 모델의 상관구역 R에 따라 causal, semicausal, noncausal 모델 등으로 나뉘어진다.^[11]

일반적으로 1 차원 모델링하는 것보다 2 차원 모델링하는 것이 보다 정확하겠으나, 1 차원 모델은 그 해법과 응용예가 상당히 많이 알려져 있는 반면^[10-15], 2 차원 모델은 현재 제한적으로 응용되고 있다. 왜냐하면 모델의 계수를 구하는 과정에서 나타나는 block toeplitz 상호상관 행렬의 역을 구하기가 용이하지 않고 또 모델의 안정도가 보장이 안된다는 결점이 있다.^[16-17] 따라서 본 논문에서는 1 차원 모델을 사용하여 영상을 모델링하는 방법을 사용하였다.

2. 1 차원 LPC (Linear Predictive Coding) 을 사용한 영상 모델링

현재의 신호 샘플의 값을 과거 몇 개의 신호 샘플값들의 선형결합으로 예측하여 예측 오차를 최소화하는 모델을 생각하면 이것은 식(1)과 같은 형태가 되어 autoregressive 모델이 된다. 이러한 방법으로 신호를 모델링하고 예측 계수를 구하는 문제는 음성처리분야에서 많이 다루어져 왔으며 보통 이것을 LPC라고 한다. $M \times N$ 영상에 대하여 1 차원 LPC를 적용하여 모델링할 때 각 행들을 서로 연결하여 LPC를 적용하면 연결 부분의 데이터간에는 행내의 데이터들에 비해 상관도가 낮으므로 오차가 많이 발생 할 수 있다.

그러므로 행들을 연결하여 사용하지 않고 각각의 행들에 대해 1 차원 LPC를 적용하여 그 LPC 계수들을 $M \times N$ 영상의 feature로 사용한다면 오차는 적어지게 된다.

예측 오차는 식(4)와 같이 실제의 신호 샘플과 선형 예측한 신호 샘플사이의 차로 정의하여,

$$\begin{aligned} \epsilon(k) &= X(k) - \hat{X}(k) \\ &= X(k) - \sum_{j=1}^P \alpha_j X(k-j) \end{aligned} \quad (4)$$

여기서, $X(k)$ 는 실제 신호 샘플값

$\hat{X}(k)$ 는 예측한 신호 샘플값

$\epsilon(k)$ 는 예측 오차

α_j 는 예측 계수

P는 예측 차수를 나타낸다.

이때, 예측 계수는 식(5)와 같이 표시되는 임의의 구간에서의 예측 오차의 제곱의 총합을 최소화하는 것으로 결정된다.

$$\begin{aligned} E &= \sum_k \epsilon^2(k) \\ &= \sum_n \left(x(k) - \sum_{j=1}^P \alpha_j X(k-j) \right)^2 \end{aligned} \quad (5)$$

그런데, 합구간을 어떻게 설정하느냐에 따라 예측 계수를 구하는 방법은 autocorrelation 방법과 covariance 방법으로 나뉘진다.^[12] Autocorrelation 방법은 covariance 방법에 비해 정확도는 떨어지지만 안정도가 보장이 되고, Durbin 알고리즘과 같은 빠른 해법이 존재한다. 따라서 본 논문에서는 autocorrelation 방법을 이용하여 예측 계수를 구하였다.

III. Feature 계산 Fast 알고리즘

그림 1과 같은 $M \times M$ search 영상내에서 $N \times N$ reference 영상의 위치를 찾을 경우 일반적인 registration 알고리즘은 $(M-N+1) \times (M-N+1)$ 개 $N \times N$ window 영상을 reference 영상과 비교해야만 하며 이에 비례하는 계산량이 필요하게 된다. 그러나 본 논문에서 제안한 알고리즘의 경우에는 현재 window 영상의 feature를 그 전에 계산된 window 영상의 feature로 이용하여 recursive한 계산이 가능하므로 결과적으로 상당한 계산 이득을 얻을 수 있다. 즉, 그림 2(a)의 (i, j) window 영상, $(i+1, j)$ window 영상에서 각각의 feature를 $[b_1(i, j), b_2(i, j), \dots, b_n(i, j)]^T$ 와 $[b_1(i+1, j), b_2(i+1, j), \dots, b_n(i+1, j)]^T$ 라 하면 두 feature들 사이에는 다음과 같은 관계가 있다.

$$b_{i+1}(i+1, j) = b_{i,1}(i+1, j), \quad i=1, 2, \dots, N-1 \quad (6)$$

따라서, $(i+1, j)$ 번째 window 영상의 feature는 $b_n(i+1, j)$ 를 제외한 나머지는 모두 (i, j) 번째 window 영상의 feature로부터 제공받을 수 있다. 그리고 $(i, j+1)$ 번째 window 영상의 feature의 관계에서는 그림 2(b)에서 보듯이 종전의 (i, j) 번째 window 영상의

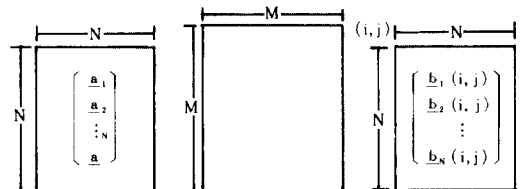


그림 1. Feature에 의한 영상의 비교
Fig. 1. Image comparison using a feature.

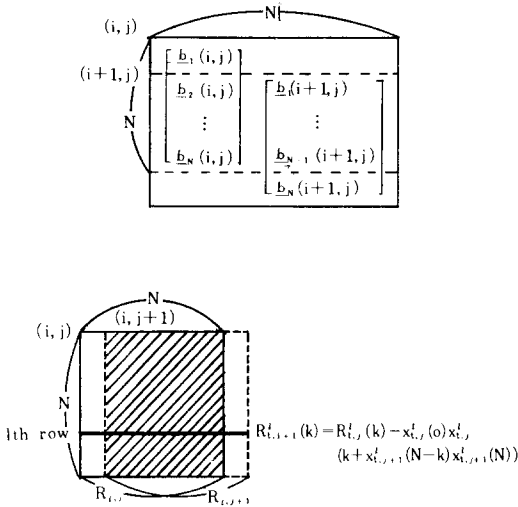


그림 2. 연속한 두 window 영상
Fig. 2. Two consecutive window images.

feature를 다시 사용할 수는 없으나 feature를 구하는데 필요한 autocorrelation 값 $R_{l,j}^l(k)$ 와 $R_{l,j+1}^l(k)$ 는 식(7)과 같은 관계를 가짐을 알 수 있다.

$$R_{l,j+1}^l(k) = R_{l,j}^l(k) - X_{l,j}^l(o) X_{l,j+1}^l(k) + X_{l,j+1}^l(N-k) X_{l,j}^l(N) \quad (7)$$

$$l = 1, 2, \dots, N$$

$$k = 0, 1, \dots, P$$

여기서, subscript (i, j) 는 window의 위치를 super-script l 은 window의 l 번째 행을 각각 나타내고 있다.

그러므로, $(i, j+1)$ 번째 window 영상의 feature $[b_1(i, j+1), b_2(i, j+1), \dots, b_N(i, j+1)]^T$ 를 구하려면 (i, j) 번째 window 영상의 autocorrelation값으로부터 식(7)을 이용하여 $(i, j+1)$ 번째 window 영상의 autocorrelation 값 $R_{l,j+1}^l(k)$ 를 쉽게 구할 수 있으므로 계산상 상당한 이득을 보게된다. 따라서, $M \times M$ search 영상내에서 $(M-N+1) \times (M-N+1)$ 개의 window 영상에 대하여 각각의 feature를 계산하는데 있어서, 위의 두식 식(6)과 식(7)을 적용시키면 그림 3과 같이 네가지 경우로 나눌 수 있으며, 각 경우에 대하여 feature를 구하는 방법을 설명하면 다음과 같다.

(1) $(0, 0)$ window 경우

식(6), 식(7)이 모두 적용안되므로 LPC계수 벡터 $b_1(0, 0)$, $i=1, 2, \dots, N$ 을 먼저 계산한다.

(2) $(0, j+1)$ window 경우, $j=0, 1, \dots, N-1$

식(7)만 적용 가능하므로 $R_{0,j}^l(k)$ 로부터 $R_{0,j+1}^l(k)$, $l=1, 2, \dots, N$, $k=0, 1, \dots, P$ 를 구한후 Durbin 알고리즘을 이용하여 $b_1(0, j+1)$, $i=1, 2, \dots, N$ 를 계산한다.

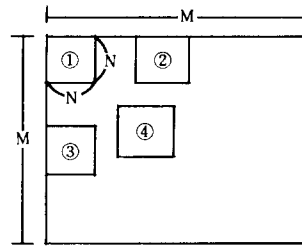


그림 3. window영상의 위치에 따른 feature 계산
Fig. 3. Feature computation depends on the position of window.

(3) $(i+1, 0)$ window 경우, $i=0, 1, \dots, N-1$

식(6)만 적용 가능하므로 $(i, 0)$ 번째 window 영상의 feature로부터 $b_1(i+1, 0)$, $i=1, 2, \dots, N-1$ 을 구한다. 그리고, $b_N(i+1, 0)$ 는 $(i+1, 0)$ 번째 window영상의 N 번째 행 데이터 $X_{i+1,0}^N(k)$, $k=0, \dots, N-1$ 에 대하여 Durbin 알고리즘을 이용하여 구한다.

(4) (i, j) window의 경우, $i \neq 0, j \neq 0$

이 경우는 식(6), 식(7)이 모두 적용 가능하다. 따라서, 먼저 식(6)을 이용하여 $(i-1, j)$ 번째 window 영상의 feature로부터 $b_1(i, j)$, $i=1, 2, \dots, N-1$ 을 구하고 식(7)을 이용하여 $(i, j-1)$ 번째 window 영상의 $R_{i,j-1}^l(k)$, $k=0, \dots, P$ 로부터 $R_{i,j}^l(k)$, $k=0, \dots, P$ 를 구한 다음 Durbin 알고리즘을 이용하여 $b_N(i, j)$ 를 구한다.

그러므로 위의 네 경우중 (1)을 제외한 세 경우 모두 미리 계산되어진 전의 window 영상의 계산값을 이용하여 recursive하게 feature를 손쉽게 얻을 수 있으므로 전체적으로 상당량의 계산을 줄이게 된다. 위의 각 경우에 대하여 계산량을 구해보면 표 1과 같다. 표 1에서 보듯이 대부분의 window 영상은 $i \neq 0, j \neq 0$ 인 경우인데, 이 경우 계산량은 window 영상의 크기 N 에 관계없이 단지 LPC계수 P 에 의해서만 결정되며, 예를 들면 $P=4$ 인 경우 24번의 곱셈만이 필요하다. 따라서 window 영상의 크기가 커질 경우 상대적으로 이 알고리즘은 더욱 효과적이다.

IV. 근사도 계산

비교하려는 두 영상에 대하여 같은 방법으로 feature를 추출하였을 경우, 적절한 metric을 정의, feature간의 거리를 계산함으로써 두 영상의 근사도 (similarity)를 판단할 수 있다. 그런데 LPC계수들간의 거리측정에 관하여서는 음성인식분야에서 많이 연구되어져 왔고 특히 그중에서 Itakura가 제시한 log

표 1. Fast 알고리즘의 계산량

Table 1. The computational requirement for the fast algorithm.

window (i, j)	계산량(곱셈)
i = 0, j = 0	$< P \cdot N^2 + \sum_{i=2}^P \{i+2 \cdot ((\frac{i}{2} + 1) - 1)\} \cdot N$
i = 0, j ≠ 0	$2PN + \sum_{i=2}^P \{i+2 \cdot ((\frac{i}{2} + 1) - 1)\} \cdot N$
i ≠ 0, j = 0	$< PN + \sum_{i=2}^P \{i+2 \cdot ((\frac{i}{2} + 1) - 1)\}$
i ≠ 0, j ≠ 0	$2P + \sum_{i=2}^P \{i+2 \cdot ((\frac{i}{2} + 1) - 1)\}$

(x) : x를 넘지 않는 최대정수
 P : LPC차수
 (N×N) : window 영상의 크기

likelihood ratio를 사용한 LPC거리측정방식이 가장 우수한 것으로 알려져 있다.^{(9),(10)} 따라서 본 논문에서는 이 log likelihood ratio를 이용한 LPC거리측정법을 사용하여 두 영상간의 근사도를 계산하였다. Log likelihood ratio를 이용하여 reference 프레임 R과 test 프레임 T와의 거리를 구하면 식(8)과 같다.

$$d(R, T) = \log \frac{\underline{a}_R V_T \underline{a}_R'}{\underline{a}_T V_R \underline{a}_T'} = \log \frac{\beta}{\alpha} \quad (8)$$

여기서,

$\underline{a}_R = (1, \alpha_{R1}, \alpha_{R2}, \dots, \alpha_{RP})$: reference 프레임의 LPC 계수 벡터

$\underline{a}_T = (1, \alpha_{T1}, \alpha_{T2}, \dots, \alpha_{TP})$: test 프레임의 LPC 계수 벡터

V_T = test 프레임의 autocorrelation 행렬이며, \underline{a}_R' 는 벡터의 transpose이다.

윗식에서 $\underline{a}_R V_T \underline{a}_R'$ 와 $\underline{a}_T V_R \underline{a}_T'$ 는 각각 입력 신호가 LPC모델 \underline{a}_R 의 필터를 통했을 때의 오차에너지, 그리고 \underline{a}_T 의 필터를 통했을 때의 오차에너지를 나타내며 \underline{a}_R 과 \underline{a}_T 가 근사하면 a(R, T)는 0에 근접하게 된다. 그러므로 모델링된 reference 영상과 window 영상의 거리 비교를 하려면 대응하는 행에 대한 LPC 계수 \underline{a}_i 와 $\underline{b}_i, (i=1, \dots, N)$ 간의 거리 $d_i (i=1, \dots, N)$ 를 구하여 이들의 합을 두 영상간의 거리로 정의, 사용하였다. 따라서 1-D LPC모델링을 이용하여 두 영상을 비교하는 알고리즘을 블럭도로 나타내면 그림 4와 같다. 표 2에는 reference 영상의 feature와 window 영상의 feature간의 거리계산에 필요한 계산량을 보 였다.

V. 컴퓨터 시뮬레이션 결과 및 검토

1. 시뮬레이션 결과 및 검토

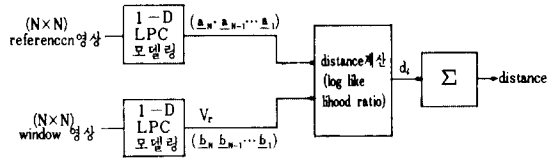


그림 4. LPC를 이용한 registration 알고리즘의 블럭도
 Fig. 4. Blockdiagram for the image registration algorithm using an LPC.

표 2. 거리측정 계산량(곱셈)

Table 2. The number of multiplication for the distance measurement.

window	(0, j) j=0, ..., M-N	(i, j), i ≠ 0
계산량	N(P+1) (P ² +1)	(P+1) (N+P ²)
search영상내의 window개수	(M-N+1)	(M-N+1) (M-N)

M×M search영상, N×N window영상, LPC 차수=P

본 논문에서는 컴퓨터 시뮬레이션을 통하여 다음과 같은 네가지 점을 검토하였다.

- (1) 제안된 알고리즘의 LPC차수 P에 대한 영향
- (2) 비슷한 여러개의 물체가 있는 영상에 대한, 제안된 알고리즘과 normalized 상관계수방법 및 TP(template matching) 방법의 성능비교
- (3) 영상전체에 균일한 bias가 가해졌을 경우 세 알고리즘의 성능비교
- (4) 영상전체에 scale 상수가 곱해졌을 경우 세 알고리즘의 성능비교

(1)의 경우는 주어진 window 영상의 크기에 대해 계산량과 성능을 고려하여 적절한 LPC차수를 결정하는 것이 목적이고 (2)의 경우는 본 알고리즘과 normalized 상관계수방법과의 성능비교를 위한 것이다. (2)에서 본 알고리즘과 다른 feature matching 방법과의 성능비교는 고려하지 않았는데 그 이유는 첫째 normalized 상관계수방법이 계산량은 많지만 이론적으로 가장 동작을 잘하여, [7]에서도 성능비교를 normalized 상관계수방법과 하였고, 둘째 edge등을 이용한 방법은 edge를 추출하는 방법에 따라 성능이 많이 좌우되기 때문이다. 또한 TP를 첨가한것은 단순히 비교목적에 있으며 본 논문에서는 절대오차의 합을 사용하였다. 그리고 (3), (4)의 경우는 서론에서 언급한것 같이 reference 영상과 search 영상이 서로다른 센서에 의해 얻어졌거나 또는 각기 다른 시간에서 얻어진 경우, 즉 영상의 intensity 변화에 따른 성능을 시험해 보기 위한

것이다. 그리고 각 알고리즘들의 성능을 객관적으로 서로 비교하기 위하여 다음과 같이 normalize 시켰다. 즉 제안한 알고리즘의 경우

$$X(u, v) = \frac{AVG[d] - d(u, v)}{AVG[d] - MIN[d]} \quad (9)$$

이다. 또한 normalized 상관계수방법은

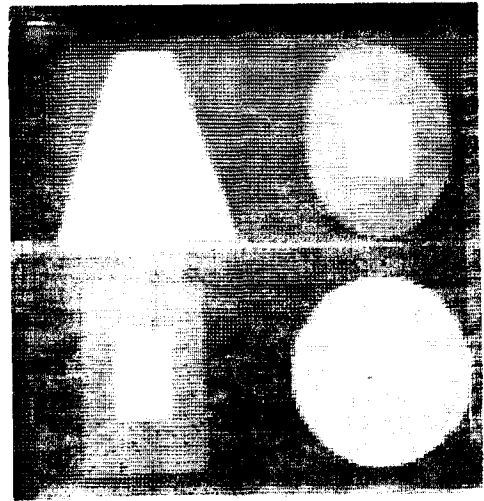
$$X(u, v) = \frac{C_N(u, v) - AVG[C_N]}{MAX[C_N] - AVG[C_N]} \quad (10)$$

으로 하였으며 TP의 경우에는

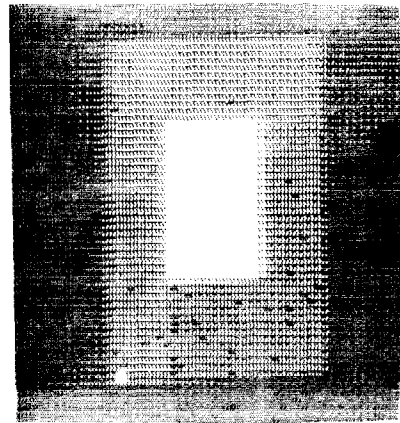
$$X(u, v) = \frac{AVG[E] - E(u, v)}{AVG[E] - MIN[E]} \quad (11)$$

로 하였다. 여기서 $AVG[\cdot]$, $MIN[\cdot]$ 및 $MAX[\cdot]$ 는 평균값, 최대값 및 최소값을 각각 의미하며 $C_N(\cdot)$ 및 $E(\cdot)$ 는 위치 (u, v) 에서 normalized 상관계수값, TP 값을 각각 나타내고 있다. 따라서 $X(u, v)$ 값은 최대값 1을 갖는 위치가 reference 영상과 가장 근사한 window 영상의 위치를 나타낸다. 또한 0값은 비교한 모든 window영상중 평균근사도를 갖는 window 영상의 위치를, 그리고 음의 값은 평균이하의 근사도를 의미한다. 그러므로 $X(u, v)$ 의 값이 정확한 reference 위치에서 유일하게 1의 값을 갖고 나머지 위치에서는 가능한 1보다 적은 값을 갖게되면 좋은 알고리즘이라고 할 수 있다. 시뮬레이션은 먼저 그림 5(a)의 테스트 영상 1과 같이 네개의 에너지가 서로 비슷하거나 크기가 비슷한 물체들로 구성된 256 gray 레벨의 32×32 search 영상과 이 영상의 (16, 0) 위치에서 얻어낸 16×16 reference영상(그림 5(b))에 대하여 제안한 알고리즘의 LPC차수 P에 따른 성능을 비교하였고 그 결과를 그림 6에 보였다. 그림 6을 살펴보면 P=2인 경우(16, 0), (16, 2)의 두 위치에서 1의 값을 가지므로 위치를 정확히 찾지 못하고 있다.

그러나 P가 4 이상인 경우는 (16, 0) 위치에서만 유일하게 1의 값을 갖고 P가 증가함에 따라 (16, 0) 위치부근 이외의 값들은 감소하고 있다. 따라서 P가 4인 경우가 16×16 window 영상에 대해 계산량과 성능면에서 적절하다고 판단하였다. 그림 7은 테스트 영상 1에 대해 제안한 알고리즘(P=4), normalized 상관계수 및 TP방법은 서로 비교한 결과이다. 그림 7을 보면 세방법모두 (16, 0)에서 정확한 reference 위치를 찾고 있다. 그러나 값들의 본분을 살펴보면 상관계수와 TP방법은 (0, 0), (0, 16) 및 (16, 16)에서 즉 reference 물체와 비슷한 물체가 있는곳에서 본알고리즘 보다 상대적으로 큰값을 가지나 제안한 알고리즘은 (16, 0)을 제외한 전 위치에서 적은값(대부분 0.3 이하)을 갖고 있음을 알 수 있다. 그림 8은 reference 영상의 각 pixel값에 bias=-20을 더한 영상을



(a)



(b)

그림 5. 시뮬레이션에 사용한 실험영상1
(a) serch(32×32)
(b) reference(16×16)

Fig. 5. Test image 1 for the simulation.
(a) search image(32×32).
(b) reference image(16×16).

사용한 결과이다. 그림 7과 비교하면 제안한 알고리즘과 normalized 상관계수방법은 별차가 없으나 TP방법은 (0, 0), (0, 16) 및 (16, 16) 부근의 값이 증가했음을 알 수 있다. 이것은 TP방법은 영상 intensity 변화에 대단히 민감하다는 사실을 입증하고 있다. 한편 그림 9는 reference pixel값에다 상수=0.5를 곱해서 얻은 영상을 사용한 결과이다. 역시 앞의 두방법은 그림 7과 비교하여 별차이가 없으나 TP는 (3, 0)부근에서 여러개의 1의값을 가짐으로서 정확한 reference 영상의 위치를 찾는데 실패하고 있다. 다음으로 실제

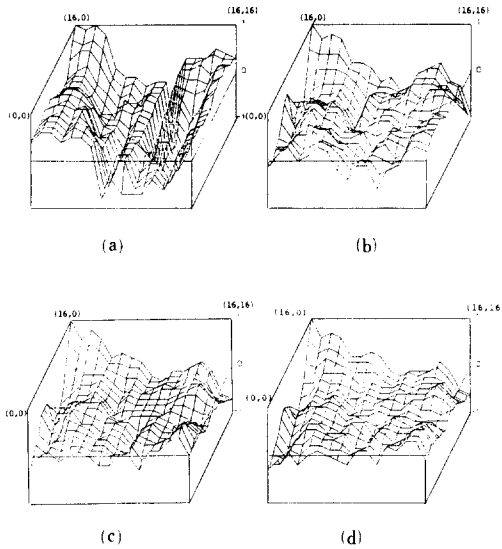


그림 6. LPC 차수 변화에 따른 시뮬레이션 결과의 3-D plot
 (a) 차수 = 2 (b) 차수 = 4 (c) 차수 = 6
 (d) 차수 = 8

Fig. 6. 3-D plot of the result of the simulation by changing the order of LPC.
 (a) order=2 (b) order=4 (c) order=6
 (d) order=8

영상에 대해 시뮬레이션을 반복 하였는데, 여기서 search 영상은 그림10(a)와 같은 128×128 영상이고 referencé 영상으로는 16×16 숫자영상으로 숫자 3 (그림10(b)), ' 숫자 8(그림10(c))를 reference 영상

1 및 2로 택하였다. 시뮬레이션 결과는 전체 window 영상에 대한 결과를 모두 도시하는 대신 search 영상 내의 숫자 1-12의 12개의 window 영상에 대한 결과만 보였다. 표3 및 표4는 각각 숫자3 및 숫자8에 대한 세알고리즘들의 결과이다. 결과를 보면 영상의 intensity 변화가 없을때는 세 알고리즘 모두 정확한 위치를 찾고 있으나 영상의 intensity 변화가 있을경우 TP는 사용하지 못함을 알 수 있다. 또한 제안된 알고리즘과 normalized 상관계수방법을 비교하면 제안된 알고리즘을 사용했을 경우 $X(u, v)$ 의 값들이 비교적 작은 값을 가짐을 알 수 있다. 표5에 본 시뮬레이션의 결과를 종합한 세 알고리즘들의 특성을 보였다.

2. 계산상의 이득

Normalized 상관계수의 경우 필요한 계산량은 $N^2 (M-N+1)^2$ 이다. 따라서 search영상의 크기가 128×128 일때 $N \times N$ window 영상의 크기 N을 변화하면서 $P = 4$ 일때의 제안한 알고리즘과 normalized 상관계수방법과의 계산량을 그림11에 도시하였다. $N=16$ 일때 본 알고리즘의 계산량이 normalized 상관계수방법에 비해 약2/3 정도이나 N이 증가함에 따라 계산량의 감소가 증가 하고있다. 이것은 제안한 알고리즘의 계산량이 N에 별로 관계가 없고 차수 P에 영향을 받기 때문이다.

VI. 결 론

본 논문에서는 새로운 영상 registration 알고리즘을 제시하였다. 제시한 알고리즘은 영상을 1 차원

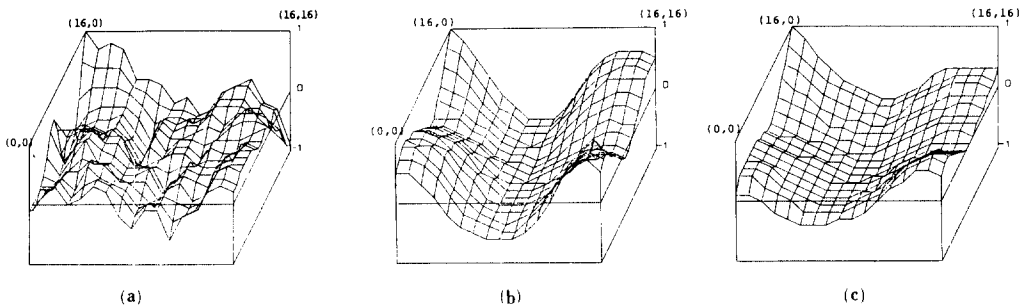
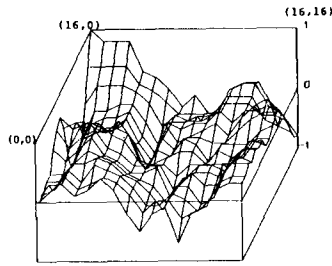
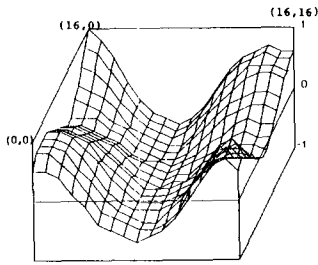


그림 7. 실험 영상 1을 이용한 시뮬레이션 결과의 3-D 프로트
 (a) 제안한 알고리즘 (차수=4) (b) normalized 상관계수 알고리즘
 (c) template matching 알고리즘

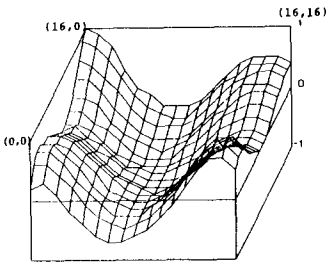
Fig. 7. 3-D plot of the results of the simulation on the test image 1.
 (a) proposed algorithm(order=4). (b) normalized correlation algorithm,
 (c) template matching algorithm.



(a)



(b)

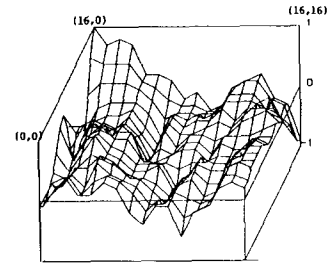


(c)

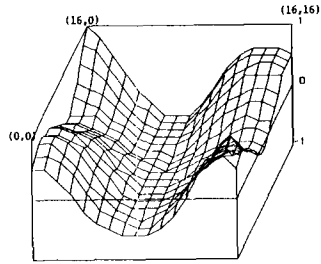
그림 8. 영상 intensity 변화에 따른 시뮬레이션 결과 (bias = -20, scale 상수 = 1)
 (a) 제안한 알고리즘 (차수 = 4)
 (b) normalized 상관계수 알고리즘
 (c) template matching 알고리즘

Fig. 8. Results of the simulation by changing the intensity of image.
 (bias = -20, scale constant = 1).
 (a) proposed algorithm (order = 4).
 (b) normalized correlation algorithm.
 (c) template matching algorithm.

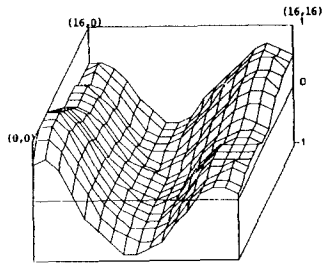
autoregressive 모델로 가정하여 1차원 LPC 계수를 그 영상의 feature로 사용하였다. 그리고 두 feature 벡터들간의 근사도 측정은 음성인식 분야에서 그 효용성이 입증된 LPC 거리척도를 사용하였으며 실제 영상을 이용한 컴퓨터 시뮬레이션을 통해 알고리즘의 타당성을 입증하였다. 컴퓨터 시뮬레이션을 통해 LPC 차수는 16×16 window 크기에서는 P = 4가 적당하였으



(a)



(b)



(c)

그림 9. 영상 intensity 변화에 따른 시뮬레이션 결과 (bias = 0, scale 상수 = 0.5)
 (a) 제안한 알고리즘 (차수 = 4)
 (b) normalized 상관계수 알고리즘
 (c) template matching 알고리즘

Fig. 9. Results of the simulation by changing the intensity of image.
 (bias = 0, scale constant = 0.5).
 (a) proposed algorithm (order = 4).
 (b) normalized correlation algorithm.
 (c) template matching algorithm.

며 실제 normalized 상관계수 방식과 비교하여 성능면에서도 뒤지지 않았다. 또한 본 논문에서는 본 알고리즘에서 feature를 recursive하게 계산할 수 있는 fast 알고리즘을 제시하여 normalized 상관계수 방식과 비교하여 월등히 적은 계산량이 필요함을 보였다.

Window 영상의 크기가 커질수록 계산량의 이득이 증가하는 것이 본 알고리즘의 특징이라 할 수 있다.

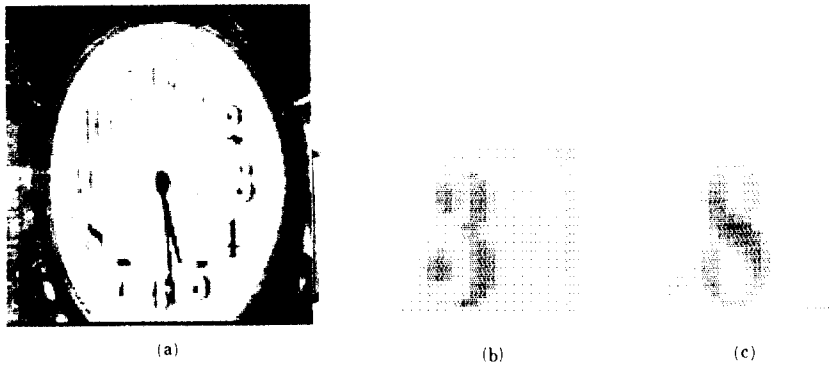


그림 10. 시뮬레이션에 사용한 실험 영상 2

(a) search (128×128) (b) reference (16×16) (c) reference (16×16)

Fig. 10. Test image 2 for the simulation.

(a) search image (128×128), (b) reference image 1 (16×16), (c) reference image 2 (16×16).

표 3. Test 영상 II의 reference 영상 1에 의한 세 알고리즘의 비교

Table 3. Comparison of the performance of the three algorithms on test image II using a reference image 1.

Simulation		window영상의 위치의 (숫자)											
		1	2	<u>3</u>	4	5	6	7	8	9	10	11	12
a)	• 제안한 알고리즘	-0.4	-0.3	<u>1.0</u>	0.0	0.1	0.1	-0.2	0.5	-0.1	-0.3	0.3	0.4
	• normalized 상관계수 방식	0.3	0.5	<u>1.0</u>	0.4	0.5	0.6	0.4	0.6	0.4	0.1	0.3	0.4
	• template matching 방식	0.0	0.4	<u>1.0</u>	0.5	0.5	0.5	0.2	0.3	0.0	0.1	0.1	0.0
b)	• 제안한 알고리즘	-0.3	-0.5	<u>1.0</u>	0.0	0.1	0.5	-0.2	0.5	-0.2	-0.5	0.2	0.4
	• normalized 상관계수 방식	0.3	0.4	<u>1.0</u>	0.3	0.4	0.6	0.3	0.6	0.3	0.0	0.3	0.4
	• template matching 방식	0.2	0.5	<u>1.0</u>	0.6	0.6	0.7	0.4	0.5	0.2	0.2	0.2	0.2
c)	• 제안한 알고리즘	-0.4	-0.4	<u>1.0</u>	0.0	0.1	0.1	-0.2	0.5	-0.1	-0.3	0.3	0.4
	• normalized 상관계수 방식	0.3	0.4	<u>1.0</u>	0.4	0.5	0.6	0.4	0.6	0.4	0.1	0.3	0.4
	• templized matching 방식	0.3	0.4	<u>0.5</u>	0.2	0.4	0.3	0.6	<u>1.0</u>	0.3	0.5	0.6	<u>1.0</u>

a) bias = 0, scale 상수 = 1 (b) bias = -20, scale 상수 = 1 (c) bias = 0, scale 상수 = 0.5

표 4. Test 영상 II의 reference 영상 2에 의한 세 알고리즘의 비교

Table 4. Comparison of the performance of the three algorithms on test image II using a reference image 2.

Simulation		window영상의 위치의 (숫자)											
		1	2	3	4	5	6	7	8	9	10	11	12
a)	• 제안한 알고리즘	0.1	-0.8	0.4	0.4	0.2	-0.3	-0.1	<u>1.0</u>	-0.1	-0.6	0.0	0.2
	• normalized 상관계수 방식	0.5	0.4	0.7	0.6	0.6	0.5	0.3	<u>1.0</u>	0.6	0.4	0.5	0.4
	• template matching 방식	0.3	0.3	0.3	0.2	0.3	0.3	0.3	<u>1.0</u>	0.4	0.3	0.3	0.3
	• 제안한 알고리즘	0.2	-0.1	0.3	0.3	0.2	-0.6	-0.1	<u>1.0</u>	-0.2	-0.8	-0.3	0.1
b)	• normalized 상관계수 방식	0.4	0.3	0.7	0.5	0.5	0.4	0.2	<u>1.0</u>	0.6	0.3	0.4	0.4
	• template matching 방식	0.6	0.5	0.5	0.4	0.5	0.4	0.5	<u>1.0</u>	0.7	0.5	0.6	0.6
c)	• 제안한 알고리즘	0.2	-0.8	0.4	0.4	0.2	-0.3	-0.1	<u>1.0</u>	-0.1	-0.5	-0.1	0.2
	• normalized 상관계수 방식	0.5	0.4	0.7	0.6	0.6	0.5	0.3	<u>1.0</u>	0.6	0.4	0.5	0.5
	• template matching 방식	0.9	0.2	-0.1	0.2	0.2	0.2	0.5	<u>0.7</u>	0.3	0.4	0.7	<u>1.0</u>

a) bias = 0, scale 상수 = 1 (b) bias = -20, scale 상수 = 1 (c) bias = 0, scale 상수 = 0.2

표 5. Registration 알고리즘의 성능비교
Table 5. Performance comparison of the registration algorithms.

성능 알고리즘	크기나 에너지가 비슷한 물체가 있는 경우	dc bias가 해진 경우	scale 상수가 곱해진 경우
제한한 알고리즘	○	○	○
normalized 상관계수 방식	△	○	○
template matching 방식	○, △	△	×

○ : 거의 영향을 받지 않음(양호)
△ : 영향을 받음(보통)
× : 심하게 영향을 받음(불량)

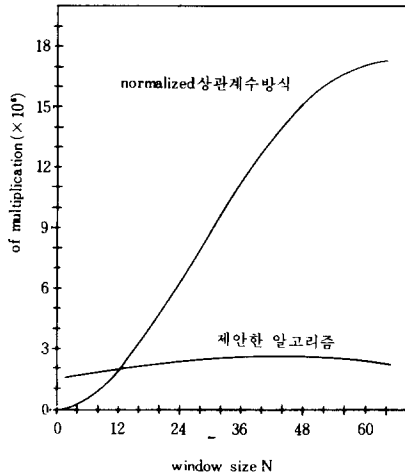


그림 11. 제한한 알고리즘과 normalized 상관계수 알고리즘의 계산량 비교 (차수=4)
Fig. 11. Comparison of the required computation between the proposed algorithm (order=4) and the normalized correlation algorithm.

그러나 앞으로 두 feature 벡터간의 근사도 측정에서 더욱더 계산량을 줄이는 문제가 또 일차적으로 영상의 해상도를 줄이거나, subtemplate를 이용하여 불필요한 window 영상에 대한 계산을 줄이는 two-stage 방식의 적용 문제들이 연구되어야 하겠다.

参 考 文 献

[1] W.K. Pratt, "Correlation technique of image registration", *IEEE Trans. AES*, vol. AES-10, no. 3, pp. 353-358, May 1974.

[2] D.I. Barnea & H.F. Silverman, "A class of algorithms for fast digital image reistration", *IEEE Trans. Computer*, vol. C-21, no. 2, pp. 179-186, Feb. 1972.

[3] A. Goshtasby, S.H. Gage & J.F. Bartholic, "A two-stage cross correlation approach to template matching", *IEEE Trans. PAMI*, vol. PAMI-6, no. 3, pp. 374-378, May 1984.

[4] G.J. VanderBurg & A. Rosenfeld, "Two-stage template matching", *IEEE Trans. Computer*, vol. C-26, no. 4, pp. 384-393, April 1977.

[5] A. Rosenfeld & G.J. VanderBurg, "Coarse-fine template matching", *IEEE Trans. Syst., Man, Cybernet.*, vol. SMS-7, no. 1, pp. 104-107, Feb. 1977.

[6] R.N. Nagel & A. Rosenfeld, "Ordered search technique in template matching", *Proc. IEEE*, vol. 60, no. 2, pp. 242-244, Feb. 1972.

[7] B. Rezaie & M.D. Srinath, "Algorithms for fast image registration", *IEEE Trans. AES*, vol. AES-20, no. 6, pp. 716-727, Nov. 1984.

[8] H.S. Ranganath & S.G. Shiva, "Correlation of adjacent pixels for multiple image registration", *IEEE Trans. Computer*, vol. C-34, no. 7, pp. 674-677, July 1985.

[9] F. Itakura, "Minimum prediction residual applied to speech recognition," *IEEE Trans. ASSP*, vol. ASSP-23, no. 1, pp. 67-72, Feb. 1975.

[10] Petros A. Maragos and et. al., "Two-dimensional linear prediction and its application to adaptive predictive coding of images", *IEEE Trans. ASSP*, vol. ASSP-32, no. 6, pp. 1213-1229, Dec. 1984.

[11] A.K. Jain, "Advances in mathematical models for image processing", *Proc. IEEE*, vol. 69, no. 5, pp. 502-528, May 1981.

[12] J.D. Markel & A.H. Gray, Jr., *Linear prediction of speech*, Springer-Verlag, New York, 1976.

[13] L.R. Rabiner & R.W. Schafer, *Digital processing of speech signals*, Prentice-Hall, Englewood Cliffs, 1978.

[14] A.N. Netravali & J.O. Limb, "Picture coding: A review", *Proc. IEEE*, vol. 68, no. 3, pp. 366-406, Mar. 1980.

- [15] A.K. Jain, "Image data compression : A review", *Proc. IEEE*, vol. 69, no. 3, pp. 349-389, March 1981.
- [16] D.E. Dudgeon & R.M. Mersereau, *Multi-dimensional digital signal Processing*, Prentice-Hall, Englewood Cliffs, 1983.
- [17] J.H. Justice, "A Levinson-type algorithm for two-dimensional Wiener filtering using bivariate Szego Polynomials", *Proc. IEEE*, vol. 65, no. 6, pp. 882-886, June 1977.
- [18] L.R. Rabiner & M.R. Sambur, "Application of an LPC distance measure to the voiced-unvoiced-silence detection problem", *IEEE Trans. ASSP*, vol. ASSP- 25, no. 4, pp. 338-343, Aug. 1977.
-