

저비용 샘플링 음향 센서네트워크에서 DTW-Cosine 알고리즘을 이용한 목표물 식별기법 (Target Classification in Sparse Sampling Acoustic Sensor Networks using DTW-Cosine Algorithm)

김 영 수 [†] 강 종 구 [†]
(Youngsoo Kim) (Jonggu Kang)

김 대 영 ^{**}
(Daeyoung Kim)

요약 센서네트워크에서 목표물 탐지하는데 있어 높은 샘플링이 수반되어야 하는 주파수 분석을 피하기 위하여, 낮은 샘플링 데이터를 이용하더라도 목표물 식별이 가능한 시계열(Time-series) 분석 기법으로서 시간 정합 유사도 측정 알고리즘을 소개하고 그 중에 가장 우수한 DTW-Cosine 알고리즘을 제안한다. 시계열 분석 기법을 이용하여 패턴을 비교하기 위해서는 지역 시간 이동 문제와 공간 신호 변이 문제를 극복해야 하는데 DTW-Cosine 알고리즘은

이를 효과적으로 극복함과 동시에 Smoothing 기법을 통하여 다른 시간 정합 유사도 측정 알고리즘들에 비해 전체적으로 최소 10.31% 이상의 우수한 성능을 보였다.

키워드 : 센서네트워크, 목표물 식별, DTW, Cosine 알고리즘

Abstract In this paper, to avoid the frequency analysis requiring a high sampling rate, time-warped similarity measure algorithms, which are able to classify objects even with a low-rate sampling rate as time-series methods, are presented and proposed the DTW-Cosine algorithm, as the best classifier among them in wireless sensor networks. Two problems, local time shifting and spatial signal variation, should be solved to apply the time-warped similarity measure algorithms to wireless sensor networks. We find that our proposed algorithm can overcome those problems very efficiently and outperforms the other algorithms by at least 10.31 % accuracy.

Key words : Sensor network, Target classification, DTW, Cosine algorithm

1. 서론

센서네트워크에 대한 최근의 급격한 기술 발전으로 인하여 목표물 식별(Target Classification)과 같은 센서네트워크 응용분야도 많은 주목을 받고 있다. 그러나 센서네트워크는 자원 제약을 받는 대부분의 센서노드들로 구성되기 때문에 저비용 샘플링 데이터를 이용한 목표물 식별 문제는 센서네트워크 응용의 주요한 이슈중의 하나이다. 특히, 목표물 식별에 주로 이용되는 음향 데이터는 매우 복잡하고 가변성이 크기 때문에 효율적인 데이터 프로세싱 기법이 절실하다.

기존의 센서네트워크에서의 목표물 식별 기법은 충분한 에너지와 컴퓨팅 성능이 보장되어야만 수행이 가능한 FFT를 통한 주파수 분석을 기반으로 연구되어져 왔다[1-3]. 주파수 분석을 위해서는 기본적으로 100 Hz 이상의 많은 샘플링이 수반되어야 의미있는 특징(Features)을 추출하기 때문에 노드의 에너지 소모가 클 뿐 아니라 네트워크 홉 수가 증가할수록 데이터 전송비용도 크게 증가할 수밖에 없다. 뿐만 아니라, 기존 연구에서는 센서노드의 이동 표적에 의한 음향학적 효과를 고려한 실질적인 연구의 수행이 부족하였다[1-4].

본 논문에서는 100 Hz 이하의 낮은 샘플링 데이터를 이용하더라도 목표물 식별을 위한 패턴분석이 가능한 시계열(Time-series) 분석 기법으로서 DTW-Cosine 알고리즘을 제안한다. 시계열 분석 기법을 적용키 위해서는 지역 시간 이동(Local time shifting) 문제와 공간 신호 변이(Spatial variation) 문제가 극복이 되어야 하

* 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원 사업(IIITA-2008-C1090-0801-0015)과 과학기술부 및 한국과학재단의 국가지정연구실 사업(R0A-2007-000-10038-0)의 지원을 받아 수행된 연구 결과임

* 이 논문은 제34회 추계학술대회에서 'DTW-Cosine 알고리즘을 이용한 저비용 샘플링 음향 센서네트워크에서의 목표물 식별기법'의 제목으로 발표된 논문을 확장한 것임

[†] 학생회원 : 한국정보통신대학교 공학부

pineland@icu.ac.kr

jjang9dr@icu.ac.kr

^{**} 정회원 : 한국정보통신대학교 공학부 교수

kimd@icu.ac.kr

논문접수 : 2007년 12월 4일

심사완료 : 2008년 2월 14일

Copyright © 2008 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 작품의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 컴퓨터의 실제 및 레터 제14권 제2호(2008.4)

는데, 이러한 문제를 해결하는 여러 알고리즘 중에 DTW-Cosine 알고리즘이 가장 뛰어난 실적을 통해 증명한다. 또한, 위에서 언급한 음향학적 변화를 고려하기 위하여 다양한 음향학적 효과(예. 도플러 효과 등)를 가미한 데이터 집합을 사용하였다.

2. 시간정합 유사도 측정 알고리즘

시간정합 유사도 측정 알고리즘(Time-warped Similarity Measure Algorithm)은 시간정합 알고리즘과 유사도 측정 알고리즘의 2-step으로 구성된다.

2.1 시간정합 알고리즘(Time-warped Algorithms)

지역 시간 이동 문제는 어떤 신호가 시간축으로 쉬프트되거나 서로 다른 길이의 신호를 비교할 때 생기는 시계열 분석에서 일반적으로 이러한 문제를 해결하는 알고리즘으로서 DTW(Dynamic Time Warping), LCS(Longest Common Subsequence) 그리고 ED(Edit Distance)이 있다.

DTW 알고리즘은 두 패턴간의 가능한 시간 정렬을 나타내는 매트릭스를 통해 최적의 패스를 찾는다[5]. 최적의 매칭은 동적 알고리즘을 통해 효과적으로 계산되어질 수 있다. 즉, 두개의 시간 순서열, $X=(x_1, x_2, \dots, x_m)$ 과 $Y=(y_1, y_2, \dots, y_n)$ 이 주어졌을 때, 식 (1)과 같은 반복함수를 이용하여 최적의 경로로 찾는다.

$$D(i, j) = d(i, j) + \min \begin{cases} D(i, j-1) \\ D(i-1, j) \\ D(i-1, j-1) \end{cases} \quad (1)$$

여기서 $1 \leq i \leq m$, $1 \leq j \leq n$, $d(i, j)$ 는 x_i 와 y_j 사이의 거리를 나타낸다. $D(1, 1)$ 은 $d(1, 1)$ 로 초기화되며 정합(Alignment)은 $D(m, n)$ 을 값으로 갖는 두 순서열 사이의 최소거리를 결과값으로 갖는다. 시계열 분석에서 시간 스케일링 문제를 해결하기 위해 DTW는 시간축으로 정렬하고 잘 정렬된 시계열 매칭 배열을 용이하게 만들어 낸다. 시계열 데이터간의 DTW 거리는 그것들의 대응 구성요소(Data)의 거리 합이다.

ED, 혹은 Levenshtein Distance,는 한 스트링이 다른 스트링으로 변형되기 위해 필요한 최소의 작동구성요소 수이다[6]. 여기서 작동이라 함은 삽입, 삭제 또는 대체를 의미한다. 예를 들면, "kitten"과 "sitting" 간의 ED는 3개의 구성요소가 바뀌기 때문에 3이다.

$$d(x_i, y_i) = \begin{cases} 0 & \text{if } x_i = y_i \\ 1 & \text{if } x_i \text{ or } y_i \text{ is a gap} \\ 1 & \text{otherwise} \end{cases} \quad (2)$$

식 (2)에서 우리는 정렬에 있어서 갭(gap)이 있는지에 따른 두 번째 케이스에 주목해야한다. 스트링을 시계열 순서열로 일반화 시키기 위해서 복잡한 것은 구성요소 x_i 와 y_i 가 심볼이 아니고 실제값인 데에 있다. 대부

분의 애플리케이션의 경우, 엄격한 동등은 적절하지 않을 수 있다. 예를 들면, $x_i = 1$ 과 $y_i = 2$ 는 순서쌍 $x_i = 1$ 과 $y_i = 1000$ 과 비교할 때 더 유사하다고 간주되어질 수 있기 때문이다. 실제값을 고려하기 위한 한 방법은 어떤 임계치(δ)이내의 차이만을 갖는 두 값을 동등한 것으로 여기는 것이다.

LCS는 실제적으로 두 스트링에 있어서 길이나 중간에 끼어든 비매칭 구성요소에 상관없이 공통된 구성요소의 개수를 나타낸다[7]. 예를 들면, 스트링 "abcdefg"와 "axbydezzz" 간에는 LCS로서 "abde"를 갖는다. LCS 공식은 다음과 같이 정의된다. 두 개의 시간 순서열 $X=(x_1, x_2, \dots, x_m)$ 과 $Y=(y_1, y_2, \dots, y_n)$ 이 주어졌을 때, LCS를 갖는 하부 순서열 Z 를 찾는 것이다. LCS의 거리합수는 다음 식 (3)과 같이 정의된다.

$$d(x_i, y_i) = \begin{cases} 0 & \text{if } i=0 \text{ or } j=0 \\ d(i-1, j-1) + 1 & \text{if } i, j \geq 0 \text{ and } x_i = y_i \\ \max[d(i, j-1), d(i-1, j)] & \text{otherwise} \end{cases} \quad (3)$$

센서네트워크에서 두 순서열간의 유사도 측정을 위해 LCS를 사용하는 것은 노이즈에 의한 영향을 피할 수 있다는 장점이 있다. 그러나, ED와 마찬가지로 LCS를 수치값으로 구성되는 시계열 데이터에 적용하기 위해서는 적절한 임계치가 설정되어야 한다는 어려움이 있다[8].

2.2 유사도 측정 알고리즘(Similarity Measure Algorithms)

공간 신호 변이 문제는 센서노드 위치에 의해 발생하는 것으로 어떤 목표물로부터 오는 모든 신호는 공간적인 거리에 상관없이 동일한 물체로 인식되어야 하는 것을 의미하며 유사도 측정 알고리즘에 의해 해결될 수 있다. 유사도는 유클리디안(Euclidean), 피어슨(Pearson) 또는 코사인(Cosine) 상관 알고리즘에 의해 계산되는 상관도(degree of correlation)에 의해 측정될 수 있다.

첫째, 유클리디안 유사도는 모든 구성요소들에 대한 두 지점간 직선 연결거리의 합으로서 계산되어진다. 그래서, 작은 거리일수록 더 높은 유사도를 의미한다. x_i 를 한 벡터, y_j 를 또다른 벡터로 가정했을 때, 유클리디안 유사도는 식 (4)와 같이 계산된다.

$$C_{euclidean}(x, y) = \sqrt{\sum_{i=1}^N (x_i - y_i)^2} \quad (4)$$

두 번째, 피어슨 유사도는 매개 상관도로서 간격이나 또는 비율 스케일상에 있는 두 순서열간의 선형적 관계 정도를 반영한다. 이 유사도의 값의 범위는 -1에서 +1까지이며 -1은 완전히 부정적 선형관계를 의미하고 반대로 +1은 완전한 긍정적 선형관계를 의미하며 0은 두 순서열 사이에 아무런 선형적 관계가 없음을 의미한다. 유사도는 m_x 는 x_i 의 평균값을, m_y 는 y_j 의 평균값을 의미할 때, 다음 식 (5)와 같이 계산된다.

$$C_{pearson}(x,y) = \frac{\sum_{i=1}^N (x_i - m_x)(y_i - m_y)}{\sqrt{\left[\sum_{i=1}^N (x_i - m_x)^2\right] \left[\sum_{i=1}^N (y_i - m_y)^2\right]}} \quad (5)$$

마지막으로, 코사인 유사도는 두 벡터간의 코사인 각도로서 계산된다. 유사도의 범위와 각 값의 의미하는 바는 피어슨의 그것과 동일하며 유사도는 다음 식 (6)과 같이 계산되어진다.

$$C_{cosine}(x,y) = \frac{1}{N} \frac{\sum_{i=1}^N x_i \times y_i}{\|x\| \times \|y\|} \quad (6)$$

여기에서 $\|x\|$ 와 $\|y\|$ 는 각각 x_i 와 y_j 의 놈(norm)이다.

그림 1에서는 위에서 기술한 유사도 측정 알고리즘의 특성을 비교하기 위한 세 개의 신호들이 그려져 있다. 이 신호들이 센서네트워크에서 발생하는 신호들이라고 가정해 보자. y_1 과 y_2 는 어떤 목표물이 이동하면서 센서노드와의 거리가 변화할 때, 발생될 수 있는 신호로 간주 될 수 있다. 즉, 거리차에 따른 소리의 높낮이만 다른 동일한 신호로 인식할 수 있다. 이는 유사도 측정 알고리즘은 이들을 동일시해야 함을 의미한다.

표 1을 살펴보면, 피어슨 알고리즘과 코사인 알고리즘은 이들을 정확히 동일시 하는 반면, 유클리디안 알고리즘은 상대적으로 그러하지 못함을 알 수 있다. 한편, y_3 와 다른 신호들 간에는 비슷하지만 정확히 같은 목표물에서 발생될 수 있는 신호들로 간주할 수는 없다. 코사인 알고리즘은 이들을 미세한 유사도 차이를 두지만 피어슨 알고리즘은 그렇지 않다. 결과적으로, 우리는 코사인 알고리즘이 다른 알고리즘들에 비해 센서네트워크에

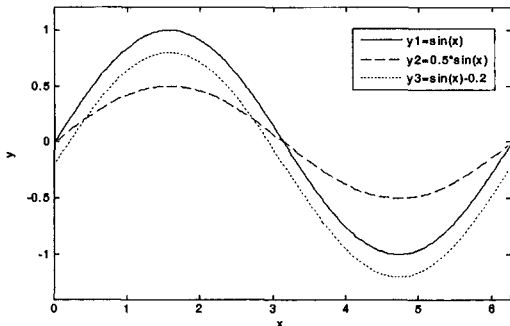


그림 1 유사도 측정 알고리즘 비교를 위한 신호들

표 1 거리/유사도 측정 알고리즘 비교

Algorithm	Euclidean	Pearson	Cosine
$y_1 \leftrightarrow y_2$	2.8025	1.0000	1.0000
$y_2 \leftrightarrow y_3$	1.5875	1.0000	0.9622
$y_3 \leftrightarrow y_1$	3.2211	1.0000	0.9622

서 발생하는 신호들의 특성을 가장 잘 대변함을 이론적으로 확인할 수 있다.

3. 실험 및 평가

3.1 실험 환경설정 및 데이터 수집

본 논문에서는 세가지 형태의 군용 목표물 - 비행기, 탱크 및 군인 - 이 식별된다. 군인 소리는 주기적으로 짧게 일어나는 발자국 소리로 구성되며 비행기 소리는 비주기의 긴 파형으로 구성된다. 또한, 탱크 소리는 단조로운 에너지를 갖는 엔진과 휠소리를 배경으로 비주기의 폭발음으로 구성된다. 도플러 효과, 히싱(Hissing) 노이즈, 에코우, 플랜저, 메카닉, 피치 변화 및 페이드인/아웃과 같은 음향 변형효과 및 잡음을 넣어 각 목표물마다 31개의 테스트 데이터 집합을 만들었다. 각 이벤트는 10 Hz로 20번씩 랜덤 시작포인트를 가지고 샘플링되어 각 목표물마다 1860번씩 식별 실험을 수행하는 것으로 음향 센서네트워크를 모델링하였다. 특징벡터는 10 Hz의 저샘플링되기 때문에 가능한 모든 데이터가 특징으로 사용되어 왜곡을 줄였다. 또한, 참조모델은 일정한 길이를 갖는 한 프레임안의 데이터들의 평균값, 즉 PAA 요소들로 구성되며 이는 원신호와 시계열적으로 매우 유사하여 참조모델로 사용하였다.

3.2 시간 정합 알고리즘 비교 분석

LCS와 ED는 각각 두 신호 비교위한 스케일링 임계치에 따라 인식률이 변하기 때문에 세 알고리즘 - DTW, LCS 그리고 ED - 의 성능을 비교하기 전에 먼저 비교되어 진다. LCS와 ED는 각각 Low(-6.02dB), Normal(0dB)과 High(6.02dB)¹⁾ 볼륨을 갖는 신호로 임계치 0.02에서 0.2까지 실험하였다. 그림 2에서 보는 바와 같이, 최적의 임계치는 볼륨에 따라 이동한다. 즉, High 볼륨의 경우에는 작아지고 반대로 Low 볼륨의 경우에는 커진다. High 볼륨에서는 최적의 임계치가

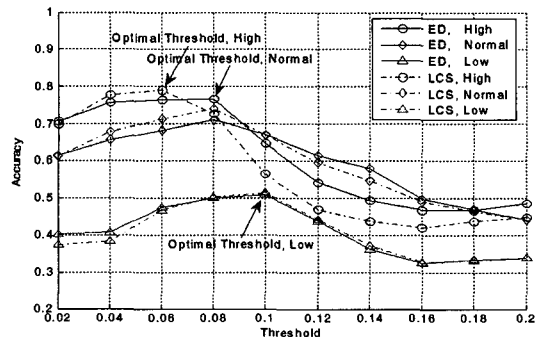


그림 2 스케일링 임계치에 따른 ED와 LCS의 비교

1) 본 실험에서 볼륨 레벨은 원 신호와 비교되는 상대적인 값임

표 2 볼륨별 시간 정합 알고리즘의 성능 비교

Volume Level	ED	LCS	DTW
High Volume(6 dB)	0.77	0.79	0.80
Normal Volume(0 dB)	0.71	0.74	0.72
Low Volume(-6 dB)	0.51	0.51	0.45

0.06이 되고 Normal 볼륨에서는 0.08 그리고 Low 볼륨에서는 0.1이 된다. 결과적으로, 전체의 볼륨에 걸친 임계치를 설정하기가 쉽지 않다.

표 2는 각각의 볼륨레벨에 따른 DTW, ED와 LCS간에 성능 비교를 보여준다. High 볼륨과 Normal 볼륨에서는 비슷한 성능을 각각 보여주고 있으며 Low 볼륨에서는 전체적으로 낮은 성능을 보여주어 이를 보상할 수 있는 즉, 공간 신호변이를 극복할 수 있는 어떤 알고리즘이 필요함을 알 수 있다. 요약하면, ED와 LCS는 볼륨에 따라 스케일링 임계치가 가변적으로 이동하기 때문에 일정하게 적용하기가 어려울 뿐 아니라, 그들의 최적의 성능 또한 DTW와 비슷한 성능을 보임에 따라 볼륨레벨에 상관없이 시간 정합 배열을 쉽게 만들어 다른 유사도 측정 알고리즘과 결합하여 성능을 보상할 수 있는 DTW 알고리즘이 최선이 선택이 된다.

3.3 시간 정합 유사도 측정 알고리즘 비교 분석

섹션 2에서 언급한 바와 같이 DTW 알고리즘과 결합하는 모든 시간 정합 유사도 측정 알고리즘들을 실험하였다. 그 결과, 그림 3과 같이 DTW-Cosine 알고리즘이 Low 볼륨영역에서 성능 손실을 최소화하는데 매우 효과적이고 전체적으로 성능이 우수함을 나타내었다.

DTW-Only와 DTW-Euclidean 알고리즘은 2 dB과 10 dB 사이에서 좋은 성능을 보이나 그들의 성능은 Low 볼륨영역에서는 낮을 뿐 아니라 10 dB 이후부터는 오히려 급격히 낮아지는 경향을 갖는다. 이는 이 알고리즘들이 볼륨레벨에 지나치게 영향을 받는다는 것을 의미한다. 반면, DTW-Pearson과 DTW-Cosine 알고리즘은 Low 볼륨영역에서 두드러지게 좋은 성능을 보

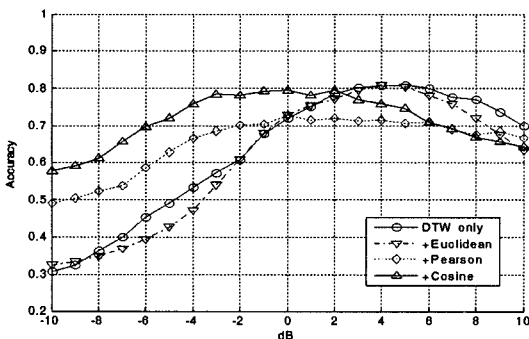


그림 3 시간 정합 유사도 측정 알고리즘 비교

이고 있으며 전체적으로 DTW-Cosine 알고리즘이 DTW-Pearson 알고리즘보다 좋은 성능을 보이고 있다. -10 dB에서 10 dB까지 고려해 볼 때, 평균적으로 DTW-Only는 68.16%, DTW-Euclidean은 62.85%, DTW-Pearson은 68.75%, 그리고 DTW-Cosine 알고리즘은 76.76%의 성능을 보였다. 그러나 아직까지도 DTW-Cosine 알고리즘이 High 볼륨과 Low 볼륨영역에서 성능의 감쇠현상이 보이는데, 이는 스무딩 기법을 통하여 보완되어진다.

3.4 DTW-Cosine 알고리즘에의 스무딩 기법 적용

스무딩 기법은 특이 데이터로 인한 왜곡의 영향을 완화시켜 성능을 증가시킨다. 이것은 환경적 잡음을 제거 시키는데 널리 이용되어져 왔다. DFT를 이용한 저대역 통과 필터(LPF)와 이동평균기법들이 그 예로서 자주 소개되어져 왔다. 그림 4에서 보는 보와 같이, 스무딩을 위한 데이터 개수를 3개로 했을 때(Smooth3), 이것은 Low 볼륨영역에서 최대로 12.74%, High 볼륨영역에서 최대로 9.19% 성능을 개선시켰다. 이는 볼륨 평가영역이 양쪽으로 커질수록 더욱 커질 것이다. 따라서, 스무딩 기법이 Low와 High 볼륨영역에서 성능을 증가시키는데 효과적임을 알 수 있다. 그러나 세 개 이상의 데이터를 이용한 스무딩은 큰 차이없이 유사함을 알 수 있으며 오히려 너무 많은 Smoothing 데이터 개수는 성능을 저하시키는데, 이는 지나친 스무딩 데이터 개수가 각 목표물 신호의 특성을 상쇄시키기 때문이다. 우리는 스무딩 기법을 다른 알고리즘들에도 적용하여 실험하였으나 DTW-Only와 DTW-Euclidean 알고리즘은 개선되지 않았고 DTW-Pearson 알고리즘은 High 볼륨에서조차도 대부분 DTW-Cosine 알고리즘에 비해 성능이 낮았다.

표 3은 각 볼륨에 따른 목표물별 혼동 매트릭스(Confusion Matrix)를 보여준다. 탱크와 비행기의 성능은 대체로 군인의 성능보다 우수하는데 이는 군인으로써

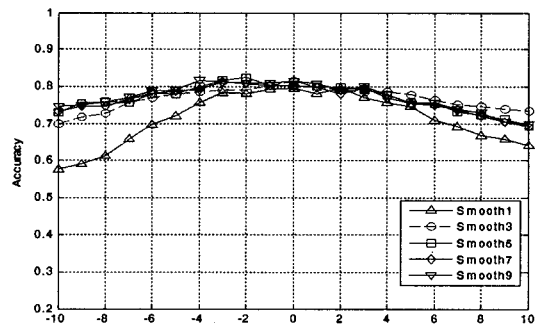


그림 4 스무딩 데이터 구성요소 개수에 따른 성능 비교 (Smooth# : #은 Smoothing data 개수)

표 3 볼륨레벨에 따른 Smooth3를 적용한 DTW-Cosine 알고리즘의 혼동 매트릭스

Classified Object \ Volume	Volume	Soldier	Tank	Airplane
Soldier	Low	0.60484	0.24355	0.15161
	Normal	0.58548	0.23710	0.17742
	Double	0.47258	0.35484	0.17258
Tank	Low	0.09677	0.75968	0.14355
	Normal	0.02742	0.93065	0.04193
	Double	0.00484	0.99032	0.00484
Airplane	Low	0.01613	0.03710	0.94677
	Normal	0.00806	0.06452	0.92742
	Double	0.00484	0.16452	0.83065

터의 유효신호의 두께가 매우 좁아 샘플링 시에 생략되어지는 경우가 발생하기 때문이다. 볼륨의 영향 또한 소리의 종류에 따라 다르게 나타났다. 다시 말하면, 군인이나 비행기와 같은 단조로운 신호는 High 볼륨에 부정적인 영향을 받는 반면, 복잡하고 가변성이 큰 탱크 소리는 긍정적인 영향을 받았다. 이는 단조로운 신호일수록 Low 볼륨영역에서 상대적으로 더 잘 모델링되기 때문이다. 결론적으로, 스무딩 기법이 적용된 DTW-Cosine 알고리즘은 Low 볼륨영역에서 77.04%, Normal 볼륨 영역에서 81.45%, 그리고 High 볼륨영역에서 76.45%의 성능을 나타내었으며 이는 ED, LCS 그리고 DTW-Only 알고리즘에 비해 전체적으로 각각 12.01%, 10.31%, 12.64% 우수하였다. 게다가, Low와 High 측정 볼륨영역이 더욱 커질수록 그 차이는 더욱 커질 것이다.

4. 결론

지금까지 음향 센서네트워크에서의 시계열 분석 방법을 이용한 전체적인 목표물 식별 시스템을 디자인하였고 최적의 시간 정합·유사도 측정 알고리즘으로서 DTW-Cosine 알고리즘의 성능이 우수함을 실험을 통해 증명하였다. 그것은 ED, LCS 및 DTW-Only에 비해 전체적으로 최소 10.31% 이상의 우수한 성능을 보였다. 도플러 효과와 같은 다양한 음향효과가 가미된 데이터 집합을 바탕으로 제안된 알고리즘이 볼륨레벨 전체에 걸쳐 만족할 만한 성능을 보임으로써, 음향 센서네트워크에서의 저비용 샘플링 데이터를 가지고 목표물 탐지 시에 시계열 분석이 유효한 방법이 될 수 있다고 말할 수 있다. 더우기, 이러한 시계열 분석 방법은 때때로 주파수 분석방법과 협력적으로 운용되어질 수 있어 에너지의 제약이 심한 센서네트워크에 유용하게 적용될 수 있다.

참고 문헌

- [1] D. Li, K. Wong, Yu Hen Hu and A. Sayeed, "Detection, Classification and Tracking of Targets," IEEE Signal Processing Magazine, pp. 17-29, 2002.
- [2] C. Meesoookho, S. Narayanan, and C. S. Raghavendra, "Collaborative classification applications in sensor networks," Second IEEE Sensor Array and Multichannel Signal Processing Workshop, pp. 370-374, 2002.
- [3] F. D. Marco, H. Yu-Hen, Vehicle Classification in Distributed Sensor Networks, Journal of Parallel and Distributed Computing, Vol.64 No.7, pp. 826-838, 2004.
- [4] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, Hongwei Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. R. Choi, T. Herman, S. Kulkarni, U. Arumugam, Mikhail Nesterenko, A. Vora, M. Miyashita, "A Line in the Sand: A Wireless Sensor Network for Target Detection, Classification, and Tracking," Computer Networks (Elsevier), 46(5): pp. 605-634, 2004.
- [5] H. Sakeo, S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," IEEE Trans. On Acoustics, Speech and Signal Processing 26, pp. 43-49, 1978.
- [6] L. Chen and R. Ng, "On the Marriage of Lp-Norm and Edit Distance," VLDB, pp. 792-803, 2004.
- [7] A. Guo, H. T. Siegelmann, "Time-Warped Longest Common Subsequence Algorithm for Music Retrieval," International Conference on Music Information Retrieval (ISMIR), 2004.
- [8] E. Keogh, K. Chakrabarti, M. Pazzani, S. Mehrotra, "Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases," Knowledge and Information Systems Journal, Vol.3, No3, pp. 263-286, 2000.