

# 효율적인 웨이블릿 기반 오디오 데이터 검색 시스템 구현

## Implementation of an Efficient Wavelet Based Audio Data Retrieval System

조 용 춘\*, 김 광 희\*, 이 배 호\*  
(Cho, Yong-Choon\*, Kim, Gwang-Hee\*, Lee, Bae-Ho\*)

\*전남 대학교 컴퓨터공학과 정보통신연구소 및 RRC  
(접수일자: 2001년 7월 30일; 채택일자: 2001년 10월 10일)

본 논문은 오디오 데이터의 검색을 위해 웨이블릿 (wavelet) 변환을 이용한 효율적인 인덱싱 방법을 제안하였다. 오디오 데이터는 그 자신이 가지고 있는 많은 저장공간의 필요, 전송에 있어서의 실시간 필요성, 큰 대역폭 등의 다양한 특성 때문에 좋은 검색효율을 위한 인덱스를 구성하기가 쉽지 않다. 신호 및 영상처리에서 각광받고 있는 웨이블릿을 이용한 인덱스는 웨이블릿 변환이 가지고 있는 여러 특징들로 인해 데이터를 블록으로 나누지 않은 상태에서의 인덱싱과 검색을 가능케 한다. 오디오 데이터의 인덱싱은 웨이블릿의 마지막 단계의 고주파 부분과 저주파 부분의 계수를 이용하여 고주파 부분은 스트링 매칭 알고리즘에 의해 스트링의 연속으로 변환하고, 저주파 부분은 영점 교차 히스토그램으로 변환한다. 구축된 인덱스를 이용한 오디오 데이터 검색은 질의 데이터와 데이터 베이스안의 인덱스 각 부분, 즉 고주파 부분과 저주파 부분의 스트링을 비교하여 가장 적은 편차를 갖는 결과를 검색 결과로 한다. 본 논문은 적절한 비교 계수 결정, 질의 길이의 변화에 따른 검색율의 변화, 데이터 각 분류별 유사도 검색 효율에 대한 실험을 하였으며, 본 논문에서 제안한 방법이 기존의 방법보다 우수한 성능 향상을 보였다.

**핵심용어:** 오디오 인덱싱, 웨이블릿 변환, 오디오 검색, DWT

**투고분야:** 음향 신호처리 분야 (1,2)

In this paper, we proposed a audio indexing method that is used wavelet transform for audio data retrieval. It is difficult for audio data to make a efficient audio data index because of its own particular properties, such as requirement of large storage, real time to transfer and wide bandwidth. An audio data index using wavelet transform make it possible to index and retrieval by using the particular wavelet transform properties. Our proposed indexing method doesn't separate data to several blocks. Therefore we use both high-pass and low-pass parts of last level coefficient of wavelet transform, Audio data indexing is made by applying the string matching algorithm to high-pass part and zero-crossing histogram to low-pass part. These are transformed to the continued strings. Through this method, we described a retrieval efficiency. The retrieval method is done by comparing the database index string to the query string and then data of minimum values is chosen to the result. Our simulation decided proper comparative coefficient and made known changing of retrieval efficiency versus audio data length. The results show that the proposed method improves retrieval efficiency compared to conventional method.

**Keywords:** Audio Indexing, Wavelet Transform, Audio Retrieval, DWT

**ASK subject classification:** Acoustic signal processing (1,2)

## I. 서론

내용기반 멀티미디어 데이터의 검색과 인덱싱(Indexing)은 주로 비디오, 이미지와 같은 가시적인 정보에 초점을 맞추고 있다[5]. 하지만 최근의 컴퓨터 저장공간과 네트워크의 비약적인 발전으로 음성신호와 같은 오디오 신호에 관심이 집중되고 있으며 많은 연구가 이루어지고 있다[6].

내용기반 멀티미디어 데이터 베이스 시스템에서 기본은 간결하고 검색이 쉽게 기술된 표현으로 각각의 멀티미디어 객체를 인덱스하는 것이다. 잘 조직된 인덱스는 빠르고 효과적인 자료의 열람과, 원하는 데이터에 대한 직접적인 접근을 가능케 한다. 이러한 것을 가능케 해주는 인덱스는 각각의 멀티미디어 객체 모델을 표현하고 명확하게 해주는 정보에 기반을 둔 의미 기술자(Descriptor)를 만들어냄으로써 구성될 수 있다[10].

오디오 데이터는 많은 저장 공간, 전송에 있어서 실시간 필요성, 큰 대역폭 등 멀티미디어 데이터가 가지고 있는 특성을 가지고 있다. 이러한 오디오 데이터의 특성은 내용기반 검색을 위한 인덱스의 작성과 가변적인 질의에 대한 인덱스의 작성을 어렵게 한다[1].

오디오 데이터의 전통적인 분류에 의하면[9] 음성, 음악, 그 외의 다른 오디오 데이터들은 각각 특성이 다르다. 그러므로 다른 데이터 타입에 대한 인덱싱 및 검색 방법 또한 달라야 한다. 음성 인덱싱이 주로 자동 음성 인식 방법에 의존하는 반면에, 오디오 및 음악 데이터의 인덱싱은 오디오 신호의 통계적 방법에 기반한다[6].

웨이블릿(Wavelet) 변환은 기존의 주파수 성분의 신호를 분석하기 위한 STFT(Short Time Fourier Transform)와 같은 변환보다 오디오와 같은 신호에 효율적이다. 웨이블릿 변환은 시간 및 주파수의 국부성(localization)을 가지므로 통계적 특성을 알 수 없거나 시간적으로 예측하기 힘든 신호해석에 널리 이용되고 있으며, 이러한 장점을 이용하면 오디오 신호와 같이 시간에 따라서 신호의 성질이 변하는 신호에 대해서 더 좋은 성능을 보인다.

본 논문은 2절에서 오디오 신호의 웨이블릿 변환을 소개하고, 3절에서는 웨이블릿 변환을 이용하여 가변적인 질의에 대한 오디오 데이터 검색을 위한 인덱스를 만들고, 인덱스를 이용한 검색방법 및 검색효율에 대해서 기술한다. 4절에서는 실험결과와 분석을 하였고, 마지막으로 결론 및 향후 연구과제에 대해서 논의한다.

## II. 오디오 신호의 웨이블릿 변환

기존의 오디오 데이터 인덱싱을 위해 사용되는 방법들은 아래와 같이 분류한다[1].

### (1) 통계 기반 인덱싱

이 방법은 평균, 분산, 영점 교차 등의 통계적 성질을 이용하는 것이다.

### (2) 음향 정보를 이용한 인덱싱

이 방법은 오디오 데이터의 크기, 피치(Pitch)나 소리의 세기, 화성 등과 같은 음향정보를 이용하는 방법이다. 결과적으로 통계적인 해석은 이 특징 벡터를 만들기 위해 이러한 속성들을 이용하고 인덱싱이나 검색에 사용한다.

### (3) 변환(Transform) 기반 인덱싱

이 방법은 DCT와 같은 변환은 데이터를 변환시키고, 적절한 계수를 골라내서 그 계수를 인덱스로 사용한다. 여기서 적절한 변환방법의 적용과 최적의 계수를 찾아내서 인덱스의 적용이 중요하다.

[3]에 의하면 변환 기반 인덱싱은 멀티미디어 데이터의 인덱싱에 좋은 결과를 얻게 하고, 잡음의 영향을 최소화한다. 하지만 DCT와 같은 변환은 윈도우(Window) 크기에 주파수 정보와 시간영역 정보의 정확도에 의해 좌우된다. 웨이블릿 변환은 이러한 문제점을 해결해 준다.

오디오 신호는 시간 축에서의 진동 웨이브 형태이다. 반면에 주파수는 시간에 따라 변하므로, 신호로부터 시간 정보와 주파수 정보를 둘다 얻을 필요가 있다. 시간에 따라서 주파수가 변하지 않는 신호는 정체신호(stationary signal)라 한다. 오디오 신호는 비 정체신호(non-stationary signal)이다. FT(Fourier Transform)는 신호의 숨겨진 내용을 알 수 있게 하지만, 시간에 대한 국부적인 정보를 제공하지 않는다. 그러므로 FT는 비 안정신호에 적합하지 않다. 반면에 웨이블릿 변환은 시간과 주파수의 정보를 동시에 포함한다.

웨이블릿 변환은  $a, t$  또는  $w, t$ 의 이차원 시간함수 STFT와 유사하다. 스케일(scale)이라 불리는 변수  $a$ 는 확장/축소에 관계되는 변수이고, 변수  $t$ 는 천이와 관계되는 변수이다. 푸리에 변환은 신호를 처리함에 있어 전구간에 걸쳐 적분을 행함으로, 국부적인 특성을 만족하지 못한다. 비 정체신호에 대해서는 주파수의 축상 전체에 대해 신호를 처리함으로 인하여 신호처리에 적합하지 못

하다. 이와 같은 단점을 개선하기 위한 방법으로 STFT 변환이 나오게 되었다. 기본 윈도우 함수  $w(t-\tau)$ 를 통한 STFT의 신호해석은 시간 및 주파수에 관계없이 항상 일정한 크기의 윈도우를 사용하게 되므로, 일반적 신호의 특징인 고주파 성분은 어떤 특정한 시간 축상 또는 공간 영역에서 국부화되고, 저주파 성분은 다수의 시간 및 공간 영역에 분포하는 성질을 만족하지 못한다. 즉, 낮은 주파수를 좀 더 높은 해상도로 맞춰 놓는다면, 높은 주파수 성분의 해상도가 낮아지기 때문에 신호의 해석이 어려워진다. 웨이블릿 변환은 STFT의 이런 점을 개선하여 일반적인 신호특성을 제공한다. 즉 신호의 급격한 변화부분은 아주 짧은 고주파 기저를 이용하고, 저주파 신호에 대해서는 긴 윈도우를 적용한다.

시간-주파수 영역에서 변하는 시간과 주파수의 해상도를 각각  $\Delta t, \Delta w$ 라 가정하고, 웨이블릿 변환을 중간 대역 통과 필터링 (band-pass filtering)으로 가정할 수 있다. 웨이블릿 변환은 시간 해상도가 중간 대역 필터의 중심 주파수에 따라서 증가하기 때문에 다음과 같은 상수 Q의 특징을 갖는다.

$$\frac{\Delta w}{w} = constant \tag{1}$$

그림 1은 시간-주파수 영역에서 STFT와 웨이블릿 변환을 비교한 것이다. 그림에서 볼 수 있듯이 웨이블릿 변환은 주파수가 높을수록 폭이 넓은 창을 사용하여 시간 해상도를 높이고, 낮은 주파수에서는 폭이 좁은 창을 사용하여 시간 해상도를 낮추는 특성을 가지고 있다. STFT에서는 시간-주파수의 해상도가 고정되는 반면, 웨이블릿 변환에서는 주파수의 변환에 따라 시간-주파수 해상도가 변하는 특징이 있다.

웨이블릿은 진동하면서 양쪽 끝으로 빠르게 소멸하는 모양을 가지고 있다. 이를 모 (mother) 웨이블릿,  $h(t)$ 라 하고, 이 모 웨이블릿이 크기 변환과 평행 이동된 형태들이 웨이블릿 변환의 기저 함수로 사용된다. 모 웨이블릿은 유일하게 결정되지 않으며, 응용 분야에 따라서 다양하게 선택하여 사용할 수 있다. 크기 변환 요소를  $a(a>0)$ , 평행 이동 요소를  $b$ 라고 하면, 웨이블릿 기저 함수  $h_{a,b}(t)$ 는

$$h_{a,b}(t) = \frac{1}{\sqrt{|a|}} h\left(\frac{t-b}{a}\right) \tag{2}$$

로 표현된다. 크기 변환된 웨이블릿은 에너지를 정규화시키는 항  $\frac{1}{\sqrt{|a|}}$ 가 포함되어 있다. 이것은 크기 변환된 웨이블릿의 에너지와 모 웨이블릿의 에너지를 같게 하기 위한 것이다. 또한 웨이블릿은  $L^2(R)$ 함수의 형태가 되어야 한다.  $L^2(R)$ 은 유한 에너지 함수, 즉 독립 변수의 범위로 적분하여 그 값이 유한한 값을 가지는 함수이다. 식 (2)와 같은 웨이블릿 기저 함수를 사용하여 어떤 입력 신호  $x(t)$ 에 대한 웨이블릿 변환  $W_x(a, b)$ 는

$$W_x(a, b) = \langle h_{a,b}(t), x(t) \rangle = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} h^*\left(\frac{t-b}{a}\right)x(t)dt \tag{3}$$

로 정의된다.  $W_x(a, b)$ 는 입력 신호를 크기 변환 및 평행 이동된 모 웨이블릿의 선형 조합으로 나타낸다. 크기 변환 요소  $a$ 가 작아지면, 웨이블릿 기저 함수의 폭이 좁아진다. 작은 크기 변환 요소는 입력 신호의 고주파 성분을 반영한다. 반대로 크기 변환 요소  $a$ 가 커지면 웨이블릿 기저 함수의 폭이 넓어져서 입력 신호의 저주파 특성을 반영한다.

DWT는 CWT (Continuous wavelet transform)에 비해

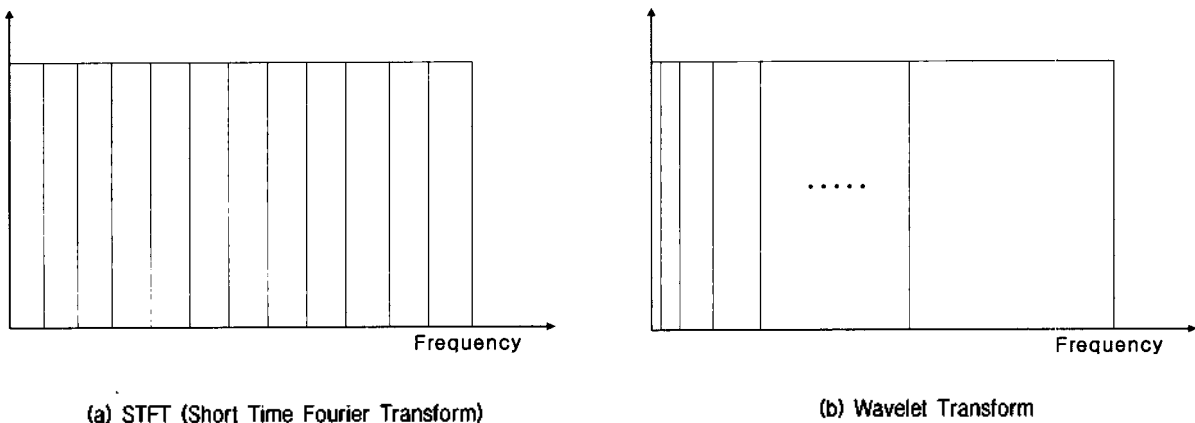
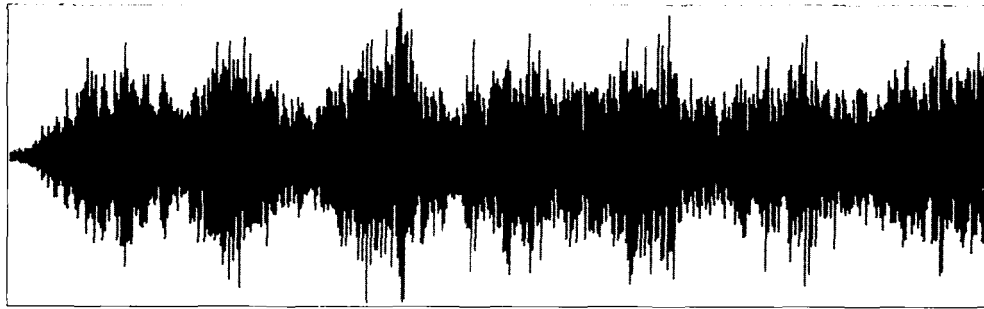
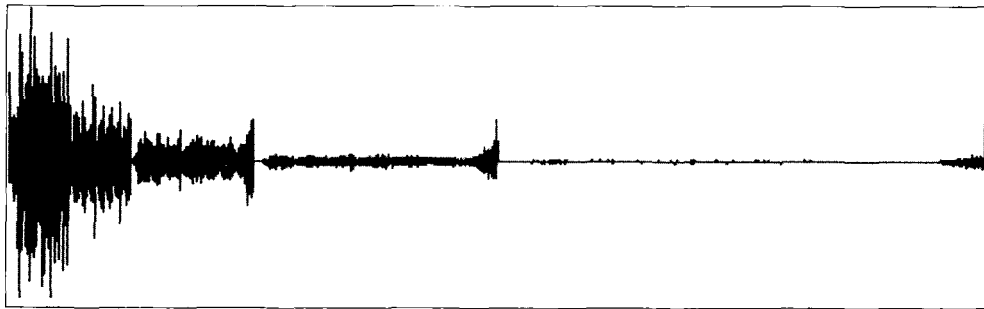


그림 1. STFT와 웨이블릿 변환  
Fig. 1. STFT and Wavelet transform.



(a) 원 신호



(b) 변환 후 신호

그림 2. 3단계 웨이블릿 변환  
Fig. 2. 3-level Wavelet transform.

서 구현이 더 쉽다. DWT에서는 다른 차단 주파수의 필터들은 다른 크기들의 신호를 해석하는데 사용된다. 이러한 필터링 단계는 오디오 신호를 다른 주파수 대역으로 구성되는 다양한 신호로 재구성한다[6].

그림 2는 오디오 데이터의 파형과, Daubechies wavelet-4 커널을 사용한 3단계 DWT 재구성을 보여준다. 전체 신호의 길이를  $L$ 이라고 하면  $n$ 단계의 웨이블릿 변환 이후에도 신호의 길이는 변화가 없다. 웨이블릿 변환의 단계가 내려 갈수록 근사 계수의 크기는 작아지지만 그만큼 세부 계수의 크기가 커져서 전체 신호의 크기는 변화가 없다.

마지막 단계의 저주파 영역은 근사 계수이고, 각 단계의 고주파 영역의 연속은 세부계수이다. DWT 계수는 근사 계수와 세부계수의 조합이다. 주목할 것은 저주파 영역은 연속해서 두 개의 영역으로 분리되고, 고주파 영역은 그대로 유지된다는 것이다. 근사 계수는 웨이블릿 변환을 거듭할수록 주파수 분해능은 좋아지고 시간 분해능은 낮아진다. 근사 계수는 높은 주파수 분해능을 갖게 되고, 시간 분해능은 낮아지게 된다.

인덱싱은 데이터를 효과적으로 검색하는 과정에서 사용되는 데이터의 검색 기법이다. 멀티미디어 데이터, 특히 오디오 데이터는 많은 표현이 존재할 수 있고, 그러한 데이터를 해석해 내기도 그리 쉬운 일도 아니기 때문에

기존의 키워드 기반 인덱싱은 오디오 데이터에 적용하기는 많은 제한을 받을 수밖에 없다. 따라서 멀티미디어 데이터의 인덱싱은 내용기반 질의를 구성해야 한다. 또한 이러한 인덱싱은 검색뿐 아니라 데이터의 삭제, 추가 등의 기능에도 적합해야 한다. 본 논문은 효율적인 검색을 위한 웨이블릿 변환을 통한 유동 인덱싱에 초점을 맞추고 있다. 즉, 질의 데이터의 크기에 상관없이 검색 가능한 인덱싱에 초점을 맞추고 있다. 유동 인덱싱의 작성은 기존의 DWT 계수의 수치적인 해석으로는 제한적이다. 이는 기존의 방법은 제한된 길이의 질의를 사용과 일정한 크기의 데이터 블록의 연속으로 이루어졌기 때문이다.

### III. 인덱싱 및 검색

웨이블릿 계수를 이용한 인덱싱의 작성은 한 개의 데이터 전체의 특징을 대표하게 되며, 오디오 데이터의 어떤 부분을 찾기 위해서는 표본 데이터의 각 부분마다의 웨이블릿 계수를 가지고 있어야 한다. 즉, 하나의 데이터에 대한 하나의 계수를 가지고서는 웨이블릿을 통한 인덱싱은 적어도 10초 정도 길이에 대해서 40 블록 이상에서 60% 이상의 결과를 기대할 수 있다[1]. 따라서 표본 데이

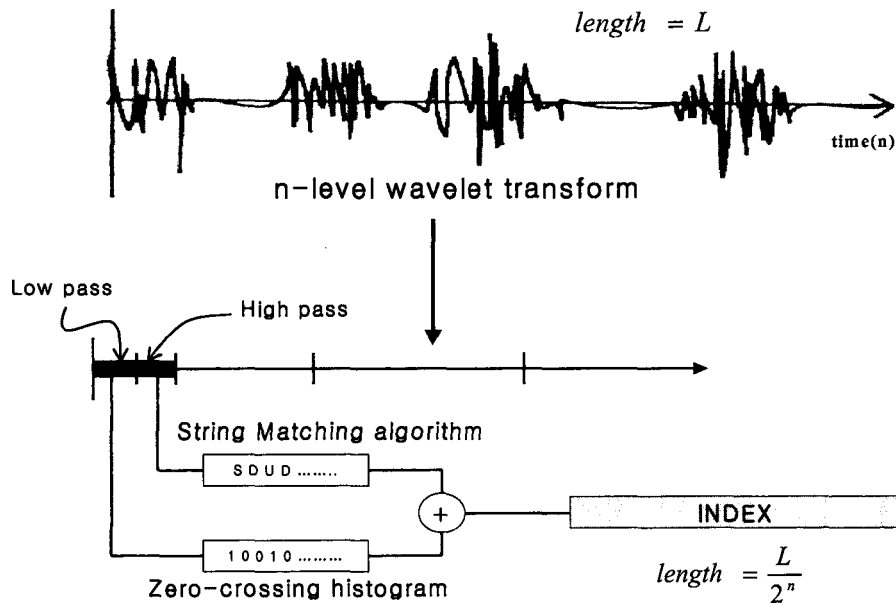


그림 3. 웨이블릿 변환을 이용한 인덱스 작성  
Fig. 3. Make index using Wavelet transform.

터를 여러 블록으로 나누지 않고 임의의 부분에 대한 검색 및 인덱싱을 위해서는 웨이블릿 계수에서 질의 데이터에 대한 블록을 설정할 수 있는 알고리즘이 필요하다.

### 3.1. 인덱싱

본 논문에서는 작성된 웨이블릿 계수에서 질의 데이터가 위치하는 블록을 결정하기 위해서 그림 3과 같이 음악 데이터 검색에 자주 사용되는 스트링 매칭 (String Matching) 알고리즘과 영점 교차 히스토그램 (Zero Crossing Histogram) 을 이용한다.

$$Sgn(n) = \begin{cases} U & p(n) < p(n+1) \\ S & p(n) = p(n+1) \\ D & p(n) > p(n+1) \end{cases} \quad (4)$$

$$0 \leq n \leq l$$

스트링 매칭 알고리즘은 내용기반 음악 데이터 검색에 많이 쓰이는 알고리즘으로, 위의 식에서와 같이 음악데이터의 멜로디 (Melody)를 세가지 문자의 연속으로 표현한다. 즉 음악 데이터의 멜로디를 S (same pitch), U (pitch increase), D (pitch decrease)의 세 가지 문자의 연속으로 변환시킨다[2,4]. 식 (4)에서  $l$ 은 웨이블릿 마지막 단계의 세부계수의 길이이며,  $Sgn(n)$ 은 문자열이고,  $p(n)$ 은 마지막 단계 세부계수의 값이다. 스트링 매칭 알고리즘만을 이용한 인덱싱 및 검색 결과는 세밀한 결과보다는 개괄적이고 포괄적인 결과를 보인다. 이는 각 값들의 고유 멜로디보다는 각 피치 (Pitch)의 상대적인 값의

높낮이만을 이용하기 때문이다[4].

전통적인 신호의 특정방법으로는 평균값, 영점교차, autocorrelation 함수 등이 있다[7]. 비 연속적인 신호에서 영점 교차는 이어지는 신호가 다른 부호를 갖는지 판단케 한다. 오디오 신호의 각 단계 웨이블릿 계수는 진동파형을 가지고 있다. 영점 교차 히스토그램은 변환영역에서 웨이블릿 계수가 얼마나 많은 대수적인 부호의 변환이 일어나는지 알게 해준다. 본 논문에서 영점 교차 히스토그램은  $Z_C(n)$ 으로 정의하며, 이는 0과 1의 연속이다. 즉, 영점 교차가 일어난 부분은 1로 하고, 나머지 부분에서는 0으로 한다. 따라서 인덱스는  $Sgn(n) + Z_C(n)$ 으로 정의하며, 이는 문자열 S, U, D와 0, 1의 연속이다.

### 3.2. 검색

위의 인덱스를 이용한 검색은 질의 데이터를 먼저 위의 인덱싱 방법과 동일하게 처리하는 전처리 과정을 거쳐서 데이터베이스의 인덱스와 비교한다. 질의 데이터의 스트링 매칭 알고리즘에 의한 문자열과 데이터베이스의 문자열을 비교하여 해당되는 후보 블록을 결정한다. 결정된 후보 블록에 대해서 각각 영점 교차 히스토그램을 비교하여 질의 데이터와 표본 데이터의 차이를 계산하고, 최소 값을 갖는 10개의 데이터를 순서대로 결과로 얻는다. 이렇게 함으로써 불필요한 부분의 검색은 피할 수 있으므로 검색시간을 단축할 수 있다.

이러한 검색방법과 인덱싱 방법의 적용은 가변 길이의 질의 데이터를 이용하여 데이터베이스에 있는 데이터 중

에서 가장 밀접한 유사도를 갖는 데이터를 찾아내기 위한 이다. 즉, 질의 데이터가 어떤 오디오 데이터 파일의 어느 부분에서 가장 밀접한 유사도 유무와 그 데이터와 위치를 검색할 수 있다.

### IV. 실험결과 및 분석

실험용 프로그램은 Microsoft Visual C++을 이용하여 프로그래밍 하였으며, 실험은 100여개의 다양한 Wave File을 인덱싱하여 인덱스를 이용한 질의에 대한 검색율에 초점을 맞추었다. 각 데이터들은 8 bit mono, 초당 22 KHz 샘플링 비트 (Sampling Bit) 사운드를 사용하였다. 표본 데이터는 225 KByte에서 3 MByte까지의 길이의 데이터로 구성하였으며, 질의 데이터 또한 마찬가지로 가변 길이로 구성하였다.

본 논문에서 제안한 인덱스를 이용한 검색은 세부계수를 이용한 블록을 결정하고, 결정된 블록에 대해서만 근사 계수의 차이를 검사한다. 세부계수의 결정은 경계값을 이용한다.

$$\text{검색율}(\%) = \frac{\text{정상 검색 개수}}{\text{전체 질의 개수}} \times 100 \quad (5)$$

먼저 세부계수를 이용하여 블록을 결정하기 위해서는 경계값을 결정하여야 한다. 경계값 결정을 위한 실험은 경계값을 인덱스로 구성한 세부계수와 질의 데이터의 세부계수를 상호 매칭시켜 매칭된 개수를  $c$ 라 하고, 질의 데이터의 세부계수의 개수를  $a_c$ , 매칭 계수를  $\tau$ 라고 한다면  $c > a_c \times \tau$  일 때, 유사블록으로 결정하며 이렇게 결정된 유사블록에 대해서만 영점 교차 히스토그램을 비교하게 된다. 표 1은  $\tau$  값의 변화에 따른 검색율을 보여준다. 표 1에서  $\tau$  값이 0.5일 때와 0.6일 때의 검색율은 그리 차이가 나지 않는다. 하지만 검색 시간에 있어서 많은 차이가 있다.  $\tau$  값을 0.5로 했을 경우 너무 많은 유사블록이 존재하게 되므로 많은 검색 시간을 소모하게 된다. 즉 본 논문에서 제안한 방법을 통한 검색은  $\tau$ 가 0.6일 때 좋은 효율을 낸다는 것을 볼 수 있다.

표 1.  $\tau$  값에 따른 검색율  
Table 1. Retrieval ratio to  $\tau$ -value.

질의길이 \ $\tau$	0.5	0.6	0.7	0.8
10초	70.4	81.5	55.6	18.5
6초	66.7	69.8	44.4	18.5

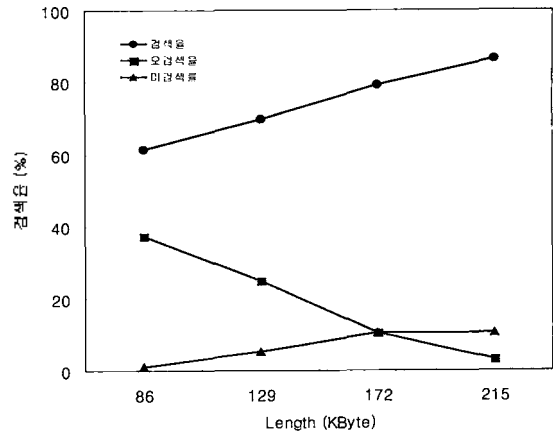


그림 4. 질의길이에 따른 검색율  
Fig. 4. Retrieval ratio to the query length.

그림 4는  $\tau$ 를 0.6으로 설정하고 100여 개의 표본 데이터에 대해 질의 데이터의 길이에 따른 검색율과 오 검색율, 그리고 미 검색율을 보여준다. 여기서 오 검색율은 검색하였으나 질의한 데이터의 위치가 틀렸거나 잘못된 검색결과를 도출한 것이고, 미 검색율은 검색하지 못해서 없는 데이터로 결과를 낸 것이다.

표 2는 인덱싱 데이터를 종류별로 나누고 종류별로 5초의 질의 데이터를 가지고 질의했을 때의 상위 10개의 검색결과 예이다. 즉 여러 개의 범주 중에서 Storm, Dog, Rain 웨이브 파일을 가지고 질의했을 때의 상위 결과 10개를 보여주며, 표에서 음영으로 나타난 결과가 같은 범주에 있는 결과를 보여준다. 여기서 전체적인 질의의 응답율은 ARD (Average Rank Difference)를 통해 알 수 있다.

$$ARD = \frac{1}{k} \sum_{i=1}^k (r_i - i) \quad (6)$$

식 (4)에서  $k$ 는 결과로 찾아진 개체의 개수이며,  $r_i$ 는

표 2. 종류별 질의 예제에 대한 검색 결과  
Table 2. Retrieval result to the categorized query examples.

질의 예	Storm	Dog	Rain
검색결과 (상위 10개)	loudthun	신비물기	nuttaff
	cement	smalldog	두꺼비
	badstorm	숨털오리	evlstrom
	bigbark	Dogsbark	dogsbark
	hurricane	갈기미귀	갈기미귀
	smalldog	kitten	kitten
	hahaha	고니	두꺼비
	호이친새	갈기미귀	-
	flowater	배꾸기	-
	sick0	나이팅게일	-
ARD	0.3	0.3	0.1

올바른 검색결과와 순위이고,  $i$ 는 이상적인 검색결과와 위치이다. ARD 값은 0값과 가까울수록 더 좋은 검색 결과를 나타낸다[8].

표 1과 표 2, 그림 4의 결과를 볼 때 경계값 계수  $\tau = 0.6$ 이고, 질의 데이터의 길이가 길수록 정확한 검색결과를 얻을 수 있다. 80% 이상의 검색율은 질의 길이가 172KByte 이상일 때 가능하였다. 이것은 [1]과 같이 오디오 데이터를 여러 블록으로 나눈 후 검색 및 인덱싱을 하는 방법보다 개선된 결과를 보여준다.

### V. 결론 및 향후 연구과제

오디오 데이터를 블록으로 나누지 않고 인덱싱하는 방법은 기존의 데이터를 여러 개로 나눈 후 인덱싱하는 방법은 가변 길이의 질의를 행할 수 있다는 점에서 상이하며, 임의의 데이터에서 잘려나온 질의나 임의의 질의 데이터를 가지고 유사 블록을 찾을 수 있다는 장점을 가지고 있다. 또한 스트링 매칭 알고리즘을 이용하여 후보 블록을 결정하는 것은 기존의 사람이 블록을 결정하는 수작업의 한계를 극복할 수 있으며, 영점 교차 히스토그램을 이용하여 검색 시간을 단축시키고, 검색율에 있어서도 상당히 향상된 결과를 보였다. 이러한 검색율의 향상에도 불구하고 아직은 질의의 길이와 모든 오디오 데이터 부류에 안정된 검색율을 위한 경계값 계수를 구하는 문제는 아직 많은 실험과 연구를 필요로 한다.

이 방법은 오디오 데이터가 멀티미디어 데이터의 일부인 점을 감안한다면, 멀티미디어 데이터의 워터마킹(Water marking)이나 비디오 데이터의 검색의 일부분으로 적용하여 기존의 방법과 더해져서 생체 인식과 같은 분야에서 쓰일 수 있으며, 더 좋은 검색 결과를 도출할 것이다.

### 감사의 글

이 논문은 2000년도 전남대학교 연구소육성과제연구비 지원에 의하여 연구되었음.

### 참고 문헌

1. S. R. Subramanya and A. Youssef, "Wavelet-based Indexing of Audio Data in Audio/Multimedia Databases," *Proc. of the*

*Int. Workshop on Multimedia Database Management Systems*, pp. 46-53, 1998.  
 2. C. C. Liu, J. L. Hsu and A. L. P. Chen, "An Approximate String Matching Algorithm for Content-Based Music Data Retrieval," *Proc. of the IEEE Multimedia System'99*, pp. 451-456, 1999.  
 3. S. R. Subramanya, "Experiments in Indexing Audio Data," Tech. Report, GWU-IIST, January 1998.  
 4. A. Ghios, J. Logan, D. Chamberlian and B. C. Smith, "Query by Humming: Musical Information Retrieval in an Audio Database," *Proc. of ACM Multimedia*, pp. 231-236, 1995.  
 5. T. C. Chiueh "Content-Based Image Indexing," *Proc. of the 20th VLDB Conf.*, pp. 582-593, 1994.  
 6. G. H. Li and A. A. Khokhar, "Content-based Indexing and Retrieval of Audio Data Using Wavelets," *Proc. of the 2000 IEEE Int. Conf. on Multimedia and Expo*, vol. 2, pp. 885-888, 2000.  
 7. C. Rowden, *Speech Processing*, McGraw-Hill Int. Limited, 1992.  
 8. K. Melih and R. Gonzalez, "Audio Retrieval Using Perceptually Based Structures," *Proc. of the IEEE Int. Conf. on Multimedia Computing and Systems*, pp. 338-347, 1998.  
 9. S. R. Allem, *Audio in Media*, Wadsworth Publishing Company, 1999.  
 10. J. H. Nam and A. H. Tewfik, "Progressive Resolution motion indexing of video object," *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 6, 1998.  
 11. S. R. Subramanya, R. Simha, B. Narahari, and A. Youssef, "Transform-based indexing of audio Data for multimedia databases," *Int. Conf. on Multimedia Computing System*, Ottawa, Ontario, Canada, pp. 211-218, June 3-6, 1997.

### 저자 약력

• 조 용 춘 (Cho, Yong-Choon)



1994년 2월 ~ 현재: 전남대학교 컴퓨터공학과 재학중  
 ※ 주관심분야: 멀티미디어 시스템, 데이터베이스, 정보검색

• 김 광 회 (Kim, Gwang-Hee)



1998년 2월: 전남대학교 컴퓨터공학과 졸업  
 2000년 2월: 전남대학교 컴퓨터공학과 석사  
 2000년 3월 ~ 현재: 전남대학교 컴퓨터공학과 박사 과정  
 ※ 주관심분야: 멀티미디어 시스템, 영상처리, 패턴 인식, 정보검색

• 이 배 호 (Lee, Bae-Ho)



1978년 2월: 한양대학교 전자공학과 졸업  
 1980년 2월: KAIST 전기 및 전자공학과 석사  
 1993년: University of Missouri, 전기 및 컴퓨터공학과 박사  
 1980년 ~ 1983년: 국방과학연구소 연구원  
 1993년 ~ 현재: 전남대학교 컴퓨터공학과 부교수  
 ※ 주관심분야: 멀티미디어 통신, 컴퓨터비전, 영상 처리, 인공지능, 정보검색