

# 비디오 데이터 세트의 하이퍼 사각형 표현에 기초한 비디오 유사성 검색 알고리즘

이 석 룡\*

요 약

이 연구에서는 대용량 비디오 데이터 스트림에 대한 유사성 검색 알고리즘을 제시한다. 수많은 프레임으로 이루어진 비디오 스트림은 각 프레임을 다차원 벡터(multidimensional vector)로 나타냄으로써 다차원 데이터 공간 상에서 시퀀스로 나타낼 수 있다. 이 시퀀스의 특성을 분석함으로써 각 시퀀스를 비디오 세그먼트(video segment)와 이 세그먼트의 집합인 비디오 클러스터(video cluster)로 표현한다. 본 연구에서는 이러한 비디오 세그먼트와 클러스터를 사용하여 두 비디오 스트림 사이의 유사성 함수(similarity function)를 제시하고, 이 함수에 근거하여 비디오 세그먼트의 하이퍼 사각형과 대표 프레임에 기초한 두 가지의 유사성 검색 알고리즘을 제안한다. 전자는 정확성(correctness)을 보장하는 알고리즘이며, 후자는 정확성을 약간 희생하는 대신 상당한 효율성을 얻을 수 있는 알고리즘이다. 다양한 유형의 비디오 스트림 및 가상으로 생성된 스트림 데이터에 대한 실험을 통하여 제시한 알고리즘의 성능을 분석한다.

## Similarity Search Algorithm Based on Hyper-Rectangular Representation of Video Data Sets

Seok-Lyong Lee†

ABSTRACT

In this research, the similarity search algorithms are provided for large video data streams. A video stream that consists of a number of frames can be expressed by a sequence in the multidimensional data space, by representing each frame with a multidimensional vector. By analyzing various characteristics of the sequence, it is partitioned into multiple video segments and clusters which are represented by hyper-rectangles. Using the hyper-rectangles of video segments and clusters, similarity functions between two video streams are defined, and two similarity search algorithms are proposed based on the similarity functions : algorithms by hyper-rectangles and by representative frames. The former is an algorithm that guarantees the correctness while the latter focuses on the efficiency with a slight sacrifice of the correctness. Experiments on different types of video streams and synthetically generated stream data show the strength of our proposed algorithms.

키워드 : 비디오 검색(Video Search), 다차원 시퀀스(Multidimensional Sequence), 유사성 척도(Similarity Measure)

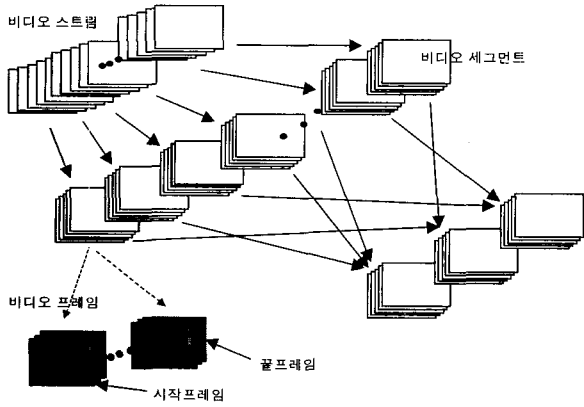
### 1. 서 론

특정 시점에서의 값을 나타내는 일 차원 실수 값들의 시퀀스인 시계열 데이터가 데이터 마이닝과 데이터 웨어하우스와 같은 다양한 데이터베이스 응용 분야에서 최근 많이 연구되고 있다. 시계열 데이터의 전형적인 예로써 시간의 흐름에 따라 변하는 주가 또는 특정 상품의 매출액, 날씨 패턴, 판매 지표와 생명의학 측정치들이 있다. 그러나 이러한 데이터들은 기본적으로 일 차원 데이터의 연속이기 때문에 기존의 연구는 일 차원 데이터 시퀀스를 인덱싱하거나 검색하는데 초점을 맞추어 왔다. 최근에 비디오 데이터

가 많은 응용 분야에서 폭 넓게 사용 됨에 따라 방대하고 복잡한 정보의 효율적인 검색이 더욱 중요해졌다. 비디오 스트림은 다차원 시계열 데이터인 다차원 데이터 시퀀스(multidimensional data sequence : MDS)로 표현될 수 있다. 이전의 연구[12]에서  $n$ 차원 공간에서  $K$ 개의 점들로 이루어진 MDS  $S$ 를 벡터 요소의 집합으로 표현하여  $S = \langle S[1], S[2], \dots, S[K] \rangle$ 의 식으로 정의하였다. 여기에서,  $S[j] (j = 1, 2, \dots, K)$ 는  $n$ 개의 스칼라 항목으로 구성되어 있는 벡터량이다. 비디오 스트림의 각 프레임은 RGB나 YCbCr 칼라 공간과 같은 특징 공간(feature space)에서 다차원 벡터로 표시된다. 따라서, 비디오 스트림은 다차원 공간에서 각 프레임을 하나의 점으로 사상함으로써 연속된 점들의 시퀀스로 모델링할 수 있다. 각 MDS는 비디오 세

\* 본 논문은 2004년 한국외국어대학교 학술연구비 지원에 의하여 연구되었음.  
† 정 회 원 : 한국외국어대학교 산업정보시스템공학부 부교수  
논문접수 : 2003년 7월 14일, 심사완료 : 2004년 3월 10일

그먼트로 시간 순으로 분할되고, 유사한 세그먼트들은 시간 순서와 관계 없이 비디오 클러스터로 군집화된다. (그림 1)은 비디오 스트림의 구조를 나타내고 있다.



(그림 1) 비디오 스트림의 구조

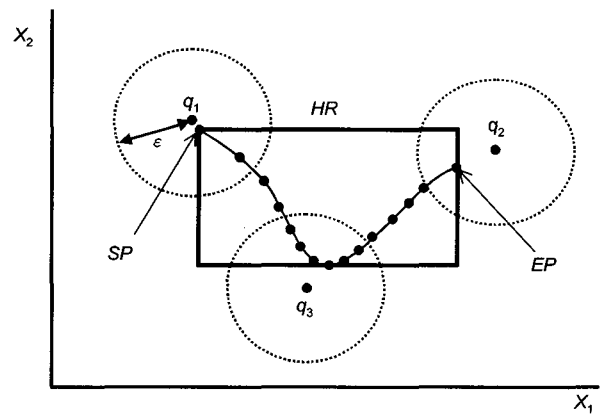
비디오 세그먼트나 비디오 클러스터는 그 안에 포함되어 있는 모든 데이터 포인트들을 둘러싸고 있는 최소 하이퍼 사각형으로 표현할 수 있으며, 최소 경계 사각형(minimum bounding rectangle : MBR)이라고도 불린다. 이 논문에서 연구하는 유사성 검색 문제는 주어진 비디오 클러스터의 집합, 질의 비디오, 그리고 임계 값에 대하여, 임계 값의 범위 안에서 질의와 유사한 비디오 세그먼트를 검색하는 것이다. 질의는 비디오 스트림의 형태로 주어지며 한 질의 비디오는 하나의 비디오 프레임, 복수개의 프레임으로 이루어진 하나의 비디오 세그먼트, 혹은 복수개의 비디오 세그먼트로 구성된다. 첫 번째와 두 번째의 경우에는 주어진 질의에 유사한 비디오 세그먼트들을 포함하는 하나의 해집합이 얻어지는 반면, 세 번째 경우에는 복수개의 해집합이 얻어진다. 비디오 세그먼트들은 하나의 비디오 스트림에 속해있을지라도 서로 상이하기 때문에 각 질의 세그먼트마다 하나의 해집합을 구하는 것이 적절하다.

일 차원 시퀀스에 대한 유사성 검색 기법들은 시계열 분석의 영역에서 광범위하게 연구되어 왔다. 그러나 이러한 기법들은 각 데이터 요소가 실수로 이루어진 일 차원 시계열 데이터에 대한 검색 기법들이다. 따라서 이러한 기법들은 다차원 데이터 시퀀스는 처리하지 못한다. 본 논문에서는 일 차원 시계열 데이터에 대한 전통적인 유사성 검색 기법을 다차원 데이터 시퀀스를 지원하기 위해 확장한다.

한편, 비디오 검색 분야에서도 다양한 검색 기법들[5, 7, 8, 21]이 제안되었다. 이러한 기법들은 주로 키 프레임에 기초한 것으로서, 키 프레임으로부터 색상(color), 질감(texture), 모양(shape) 혹은 레이아웃(layout)과 같은 특성 속성(feature attribute)들을 추출한다. 유사성 검색은 질의의 특성

속성과 데이터 베이스에 저장된 키 프레임들의 특성 속성들과 비교함으로써 수행된다. 그러나 이러한 방식은 다음과 같은 두 가지의 문제점을 가지고 있다. 첫째, 검색 메커니즘이 소수 개의 선택된 프레임에 의존하기 때문에 질의에 유사한 세그먼트들을 누락시킬 수가 있으며 따라서 정확성(correctness)을 보장하지 못한다. 다음 예는 이 점을 자세히 보여주고 있다.

예 : (그림 2)에서 보인 것처럼 시작 포인트 SP에서 끝 포인트 EP에 이르기까지 복수 개의 포인트들을 갖고 있는 시퀀스를 포함하고 있는 비디오 세그먼트의 하이퍼 사각형 HR을 고려해보자. 질의 포인트  $q_1, q_2, q_3$ 가 각각 그림에서 보인 위치에 주어지고, 임계 값(혹은 질의 반경)  $\epsilon$ 이 주어졌다 하자. SP가 키 프레임으로 선택되는 키 프레임 기반 검색의 경우, 질의  $q_1$ 에 대해서는 HR이 질의와 유사한 세그먼트로 검색된다. 그러나, 질의  $q_2$ 와  $q_3$ 에 대해서는 HR에 있는 여러 개의 포인트들이 질의 반경 내에 있음에도 불구하고 SP가 질의 반경에서 벗어나 있으므로 HR은 질의와 무관한 세그먼트로써 검색에서 제외된다. SP와 EP의 두 데이터 포인트가 키 프레임으로 선택되는 키 프레임 기반 검색의 경우에도 질의  $q_3$ 에 대해서는 마찬가지로 HR은 질의와 무관한 세그먼트로써 검색에서 제외된다. 결과적으로 '과오 누락(false dismissal)'이 발생하는 것이다. 따라서, 키 프레임 기반 검색은 정확성을 보장하지 못한다.



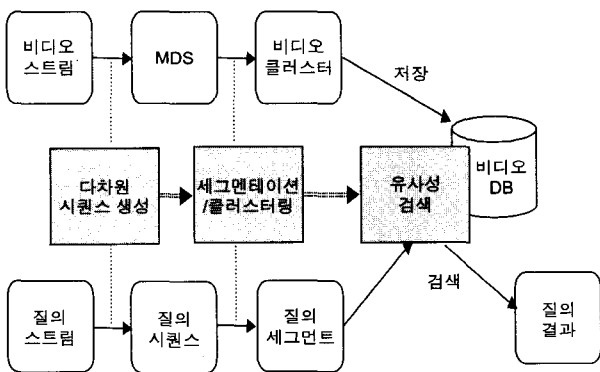
(그림 2) 키 프레임 기반 검색에 의한 과오 누락

둘째, 키 프레임 기반 검색 방식에서 키 프레임은 보통 시작 프레임 혹은 시작과 끝 프레임으로써 선택되기 때문에 해당 비디오 세그먼트를 정확히 대표한다고 볼 수 없다. 실제로 만화나 애니메이션 영화로부터 추출된 비디오 세그먼트의 경우에는 세그먼트 내의 프레임들의 내용이 역동적으로 변하므로 시작이나 끝 프레임이 해당 세그먼트를 효과적으로 대표하지 못한다. 비디오 세그먼트 내용을 보다

정확하게 대표하기 위하여 다양한 정교한 기법들이 연구되어 왔다. PanoramaExcerpts [19]가 한 예로써 카메라의 움직임을 자동적으로 감지함으로써 카메라 팬(camera pan)이나 틸트(tilt)와 같이 확장하여 세그먼트의 가능한 볼 수 있는 모든 장면을 대표 프레임으로 채택하고 있다. 그러나 이러한 방식은 대표 프레임을 합성하기 위해 상당한 오버헤드를 야기한다.

본 논문에서는 두 가지 검색 기법을 제안한다. 첫째, 하이퍼 사각형 검색 기법은 검색이 세그먼트를 표현하는 하이퍼 사각형에 기초하여 질의에 유사한 세그먼트를 찾으며, 정해성이 보장되는 방식이다. 정해성의 보장은 이 검색 기법을 제안하게 된 중요한 동기이다. 유사성 검색에서, 특히 저장된 CCTV 필름으로부터 범죄 행위를 감지하는 애플리케이션 같은 정해성에 민감한 분야에서는 유사성 검색에서 정해성의 보장이 매우 중요하다. 둘째로, 대표 프레임 검색 기법은 비디오 세그먼트나 비디오 클러스터의 대표 포인트에 근거한 방식이다. 이 기법에서 대표 포인트는 시작이나 끝 프레임을 선택하지 않고, 하이퍼 사각형의 기하학적 특성을 고려하여 효과적인 방식으로 대표 포인트를 선택한다. 그러나 이 방식은 정해성을 보장하지는 않는다. 대표 프레임 검색기법은 약간의 정해성을 희생함으로써 높은 정확율(precision)을 달성하기 위하여 설계되었다. 이 방식은 재현율(recall) 보다 정확율이 중요한 분야에서 응용될 수 있다.

다음 (그림 3)은 제안한 기법의 전체적인 구조를 도시한 것이다. 이 그림에서 음영이 있는 사각형은 프로세스를 나타내고, 라운드 사각형은 프로세스에 대한 입력/출력 데이터를 나타낸다.



(그림 3) 제안한 기법의 전체적인 구조

● **다차원 시퀀스 생성** : 비디오 스트림의 각 프레임으로부터 특성을 추출하여 각 프레임을 다차원 공간 상에서 데이터 포인트로 표현하고, 따라서 비디오 스트림은 프레임 사이의 시간적인 관계를 고려하여 다차원 데이터 시퀀스로 표현한다. 생성된 데이터 포인트가 고차원인 경우에는 고차

원으로 인하여 야기되는 문제(dimensionality curse problem)를 극복하기 위해 차원 축소(dimensionality reduction)를 수행한다.

● **세그멘테이션/클러스터링** : 생성된 다차원 시퀀스는 사전에 정의된 기하학적/의미적 조건을 고려하여 비디오 세그먼트로 분할되며, 한 비디오 스트림 내의 유사한 비디오 세그먼트들은 비디오 클러스터로 군집화된다. 이러한 방식으로 각 비디오 스트림은 소수 개의 비디오 클러스터로 표현되며, 이 비디오 클러스터가 차후의 검색을 위하여 데이터베이스에 저장된다.

● **유사성 검색** : 질의 비디오 역시 파싱(parsing)되어 하나 혹은 다수의 질의 비디오 세그먼트로 변환된다. 각 질의 세그먼트는 데이터베이스에 저장된 비디오 클러스터와 비교되어 지정된 임계 값 내의 비디오 클러스터를 후보 클러스터로 선택한다. 후보 클러스터 내의 각 비디오 세그먼트들이 질의 세그먼트와 비교되어 최종적으로 세그먼트 해집합을 구한다. 유사성 검색 알고리즘으로써 이미 언급한 하이퍼사각형 검색기법과 대표 프레임 검색기법을 사용하여 검색을 수행한다. 본 논문에서는 유사성 검색을 중점적으로 다루며, 다차원 시퀀스 생성에 관해서는 본 논문의 주제를 벗어나므로 언급하지 않는다. 세그멘테이션/클러스터링 과정에 관해서는 유사성 검색의 기초가 되므로 3장에서 간단히 기술한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 소개하고, 3장에서는 다차원 데이터 시퀀스의 세그멘테이션 및 클러스터링 기법에 대하여 간단하게 언급한다. 4장에서는 비디오 세그먼트 및 클러스터에 기초한 유사성 검색 기법을 상세히 기술하고, 5장에서 실험 결과 및 간단한 분석을 제공하며, 6장에서 결론을 맺는다.

## 2. 관련 연구

본 논문에서 제안하는 비디오 스트림의 유사성 검색 기법과 관련된 연구 분야로써 크게 두 가지 분야가 있다. 첫째는 일차원 시계열 데이터 및 다차원 시퀀스에 대한 유사성 검색 분야이며, 둘째는 비디오의 내용 기반 검색 분야이다. 일차원 시계열 데이터의 검색을 위하여 다양한 기법들이 제안되었다. Agrawal 등 [1]은 최초로 DFT(discrete Fourier transform)를 사용하여 시간 도메인의 시퀀스 데이터를 저 차원의 주파수 도메인으로 사상(mapping)하여 고차원 문제를 해결한 차원 축소 기법을 소개했다. 이 방법에서는 하나의 시간 시퀀스는 주파수 도메인에서 저 차원 상의 한 점으로 표현되며, 이 점들은 기존의 인덱싱 구조들[2,

3, 6, 9)을 사용하여 저장되고 검색된다. Faloutsos등[4]은 서로 다른 길이의 시퀀스들 사이의 유사성 검색이 가능하도록 하기 위해서 빠른 서브 시퀀스 매칭(fast subsequence matching)기법을 제안하였다. 그들은 데이터 시퀀스에 대하여 길이가  $w$ 인 슬라이딩 윈도우를 사용하여 각 윈도우에 포함된  $w$ 개의 일 차원 값들을  $w$ 차원의 한 점으로 나타내고 DFT를 사용하여 이 점들을 저 차원의 점들로 나타내었다. 저 차원으로 변환된 데이터 시퀀스는 서브 시퀀스들로 나누는데, 각각은 MBR로 표현되고 'ST-index'를 사용하여 색인되고 저장된다.

Rafiei등[17]은 주어진 시퀀스의 안전한 선형 변환들의 집합을 제안하였는데, 이것들은 시계열 데이터에 대한 유사성 질의의 기본으로 사용될 수 있다. 그들은 이동 평균(moving average), 역전(reversing), 타임 워핑(time warping)과 같은 함수들을 정형화하여 표현하였다. 타임 워핑 함수를 사용한 최근 연구로써 Kim 등[11]은 유사한 서브시퀀스 검색을 위해 시퀀스 내의 각 요소 값을 임의의 수만큼 반복시켜 비교하는 기법을 제안하였다. 또한, 최근에 Yi 등[20]은 시간 시퀀스를 같은 길이의 세그먼트로 나누고 세그먼트내의 포인트들의 평균을 저장하는 근사 기법(approximating technique)을 제안하였다. 이 기법에서는 한 시퀀스가  $s$ 개의 세그먼트로 나누어지며 그 시퀀스는  $s$ 개의 세그먼트 평균들을 요소로 가지는 벡터로 표현된다. [20]을 확장하여 Keogh 등[10]은 APCA(adaptive piecewise constant approximation)로 불리는 새로운 차원 축소 기법을 소개하였다. 이 기법에서는 하나의 시간 시퀀스를 복원 오차가 최소가 되도록 하는 길이가 다른 복수 개의 세그먼트의 집합으로 근사하여 표현하였다. 또한, APCA에서는 각각의 길이가 다른 세그먼트가 다차원 인덱스 구조로 인덱싱될 수 있음을 보였고,  $D_{LB}$ 와  $D_{AE}$ , 즉 각각 하한 유클리드 거리 근사(lower bounding Euclidean distance approximation)와 비하한 유클리드 거리 근사(non-lower bounding tight approximation)를 제공하는 두 개의 거리 측정 기준을 제안하였다.

그러나 비디오 스트림과 같은 다차원 시퀀스에서 한 점의 개념은 일 차원 시계열 데이터의 점과는 다른 의미를 가진다. 다차원 시퀀스의 점 자체는 다양한 특징을 각각 차원으로 하는 다차원 공간에서의 벡터가 되며, 이 벡터의 각 차원의 요소들은 해당 특성의 추출 값들이 된다. 따라서, 다차원 시퀀스나 서브 시퀀스를 하나의 대표적인 점으로 사상할 수 없기 때문에 위의 기존 방법들은 다차원 시퀀스에는 적용될 수 없다.

한편, 1990년대 중반에 IBM Almaden 연구소에서 발표된 QBIC(Query By Image and Video Content)[5]은 내용 기

반 검색의 효시가 된 시스템으로써, 현재 상용화가 상당 부분 진행되어 여러 가지 면에서 우수한 시스템으로 평가 받고 있다. 이 시스템에서는 각 세그먼트에 대하여 키 프레임을 선택하고 선택된 프레임들에 기초하여 질의를 처리하고 있다. VisualSEEK[18]은 WWW를 이용한 내용 기반 검색 시스템으로 Columbia 대학에서 개발되었다. 이 시스템의 가장 큰 특성은 시각적 특징에 의한 내용 기반 질의와 공간 정보에 의한 공간 질의가 동시에 가능하다는 것이다. 즉, 먼저 영상에서 색상 영역에 관한 정보를 추출한 후에, 이러한 정보를 이용하여 각 색상 영역간의 공간적인 관계를 추론함으로써 공간 관계에 대한 특징을 추출하여 질의에 사용한다. 그러나 이러한 추론 과정은 시간이 많이 소요되는 작업으로 비효율적일 수 있으며, 색상 정보에 의존한 검색만 가능하다는 단점을 가진다. Chabot[15]은 객체-관계형 데이터베이스 시스템인 Postgres를 이용한 내용 기반 검색 시스템으로 시각적인 특징에 기반한 검색뿐만 아니라 문자적인 표현에 의한 검색을 할 수 있도록 설계되었다. 한편, Zhang등[21]은 비디오 데이터를 검색하기 위하여 비디오로부터 내용을 추출하고 분석하며 분석된 내용을 효율적으로 나타내기 위한 내용 기반 비디오 색인 및 검색 기법을 제안하였으며, Lee와 Xiong[14]은 비디오를 포함한 다양한 멀티미디어 자료를 관리해주는 VIMS(Video Information Management System)에 기반한 새로운 모델을 제안하였다. 또한, Jain 등[8]은 비디오 스트림을 일정한 시간 간격으로 나눈 후, 각 세그먼트에서 대표 프레임을 추출하여 검색에 이용하는 방법을 제안하였다. 이 외에도 최근에 Oh와 Hua[16]는 카메라 추적 기술(Camera Tracking Technique)에 의해 비디오를 분할하고 이에 의거하여 비디오를 브라우징하기 위한 계층 구조를 고안하여 검색에 응용하였다.

그러나 이러한 기존의 방식은 키 프레임에 기반한 검색으로써, 키 프레임은 보통 시작 프레임 혹은 시작과 끝 프레임으로 선택되기 때문에 키 프레임이 비디오 세그먼트 내의 모든 프레임들을 항상 적절하게 요약할 수 없으며, 따라서 질의와 유사한 많은 시퀀스들을 누락하게 된다. 실제로 만화나 애니메이션 영화로부터 추출된 비디오 세그먼트의 경우에는 세그먼트 내의 프레임들의 내용이 역동적으로 변하므로 시작이나 끝 프레임이 해당 세그먼트를 효과적으로 대표하지 못한다. Lee등은 [12]에서 두 개의 하한(low bounding) 거리 측정 기준인  $D_{mbr}$ 과  $D_{norm}$ 에 기초한 다차원 시퀀스를 위한 유사성 검색 방법을 제안했다. 이 검색 방법에서는  $D_{mbr}$ 과  $D_{norm}$ 을 사용하여 질의에 무관한 시퀀스들을 검색 대상에서 제외한다. 그러나 이러한 접근법 역시 다차원 시퀀스를 위한 검색은 지원하지만 비디오의 세그멘테이션 및 클러스터링 특성을 검색에 적용하지는 못하

고 있다.

### 3. 비디오 세그멘테이션 및 클러스터링

이 절에서는 제안된 기법을 논의하기에 앞서 기본적인 용어의 정의와 세그멘테이션 및 클러스터링 과정에 대하여 간략하게 언급한다. 먼저, 하이퍼 사각형을 정의하고, 이를 이용하여 비디오 세그먼트와 비디오 클러스터를 정의한다.  $n$ 차원 데이터 공간에서  $k$ 개의 점,  $P_j(1 \leq j \leq k)$ 를 포함하고 있는 하이퍼 사각형  $HR$ 은 대각선의 두 개의 끝점  $-L$  (저점)과  $H$ (고점)과 사각형 내의 점의 수로써 표현되며, 정형적인 정의는 다음과 같다:  $HR = \langle L, H, k \rangle$ . 여기에서  $L = \{(L^1, L^2, \dots, L^n) | L^i = \min_{1 \leq j \leq k} (P_j^i)\}$ ,  $H = \{(H^1, H^2, \dots, H^n) | H^i = \max_{1 \leq j \leq k} (P_j^i)\} (i = 1, 2, \dots, n)$ 이다.

시간 순으로  $k$ 개의 점  $P_j(j = 1, 2, \dots, k)$ 를 포함하고 있는 비디오 세그먼트  $SEG$ 는 다음과 같이 정의된다:  $SEG = \langle sid, offset, HR \rangle$ . 여기에서,  $sid$ 는 segment-id,  $offset$ 은 시퀀스의 시작 점으로부터  $SEG$ 의 오프셋,  $HR$ 은  $L, H, k$ 로 표현되는 비디오 세그먼트의 하이퍼사각형이다. 또한,  $r$ 개의 비디오 세그먼트  $SEG_j(j = 1, 2, \dots, r)$ 를 갖는 비디오 클러스터  $CL$ 는 다음과 같이 정의된다:  $CL = \langle cid, slist, HR \rangle$ . 여기에서  $cid$ 는 cluster-id,  $slist$ 는  $sid$ 의 리스트,  $HR$ 은  $\langle L, H, k \rangle$ 로 표현되는 비디오 클러스터의 하이퍼사각형이다. 세그멘테이션 과정은 시퀀스 내의 데이터 포인트를 순서대로 처리하며, 미리 정해진 조건을 만족하면 세그먼트에 반복적으로 병합하는 과정이다.  $n$ 차원 공간에서 시퀀스내의 한 점이 세그먼트에 병합되는 경우를 고려해 보자. 그 점이 세그먼트에 병합될 때 미리 주어진 조건을 만족하면 현행 세그먼트에 병합되고, 그렇지 않으면 새로운 세그먼트가 그 점으로부터 시작되는 방식으로 세그멘테이션이 진행된다. 미리 정해진 조건으로써 기하학적 한계 조건과 의미적 한계 조건을 사용한다. 전자는 세그먼트의 볼륨과 에지를 고려하여 정하고, 후자는 세그먼트내의 인접한 데이터 포인트 간의 거리를 고려하여 정한다. 그리고, 시퀀스내의 유사한 세그먼트들은 역시 이러한 조건을 고려하여 클러스터로 군집화 된다. 공간상의 제약 때문에 세그멘테이션과 클러스터링 알고리즘을 상세히 기술하지 않지만, 관심 있는 독자는 기하학적 한계 조건과 의미적 한계 조건, 알고리즘, 실험 결과 등과 같은 자세한 사항에 관하여 [13]을 참조하기 바란다.

다음으로 비디오 세그먼트 사이의 거리를 측정하는 측정 함수에 대해 소개한다. 먼저 같은 길이의 세그먼트에 대하여 측정 함수를 정의하고 다음에 다른 길이의 세그먼트의 경우로 확장한다. 각각  $k$ 개의 데이터 포인트를 갖는 같은 길

이의 두 세그먼트  $SEG_1$ 와  $SEG_2$  사이의 거리  $D_{SEG}(SEG_1, SEG_2)$ 는 데이터 포인트들간의 평균 거리로써 정의되며, 이 평균 거리는 다음 식으로 계산된다:

$$D_{VS}(SEG_1, SEG_2) = D_{mean}(SEG_1, SEG_2) = \frac{1}{k} \cdot \sum_{i=0}^{k-1} d(SEG_1[i], SEG_2[i])$$

여기에서  $d(*,*)$ 는 두 데이터 포인트 사이의 유클리디안 거리이다. 다른 길이의 세그먼트의 경우에는 데이터 포인트 거리 순서대로 직접 비교할 수 없고, 짧은 세그먼트가 긴 세그먼트의 처음부터 끝까지 슬라이딩 함으로써 연속적으로 비교된다. 비교되는 쌍들의 모든 거리 값들 중 가장 작은 값이 두 세그먼트사이의 거리로써 채택된다. 각각  $p$ 와  $q$ 개의 데이터 포인트를 갖는 서로 다른 길이의 두 세그먼트  $SEG_1$ 와  $SEG_2$  사이의 거리  $D_{SEG}(SEG_1, SEG_2)$ 를 고려해 보자. 즉,  $p = SEG_1.HR.k$ ,  $q = SEG_2.HR.k$ 이 되며, 일반성을 상실하지 않고  $p \leq q$ 라 가정할 수 있다. 이때,  $D_{SEG}(SEG_1, SEG_2)$ 는  $SEG_1$ 를  $SEG_2$ 에 대하여 슬라이딩하면서 비교하여 계산한다. 이때 비교하는 세그먼트 쌍 중 최소 평균 거리로써 정의되며, 최소 평균 거리는 다음 식으로 계산된다:

$$D_{VS}(SEG_1, SEG_2) = \min_{1 \leq j \leq q-p+1} D_{mean}(SEG_1[1:p], SEG_2[j:j+p-1])$$

### 4. 유사성 검색 기법

#### 4.1 하이퍼사각형 검색기법

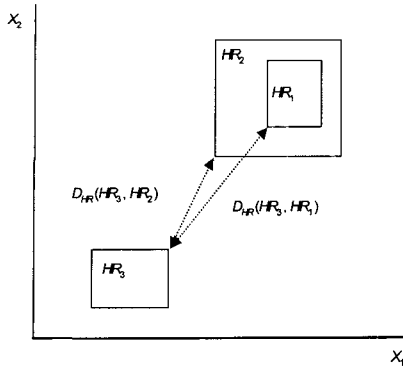
$n$ 차원 데이터 공간에서 비디오 세그먼트나 비디오 클러스터를 표현하기 위해 사용되는 하이퍼 사각형 사이의 거리  $D_{HR}$ 가 두 하이퍼 사각형 사이의 최소 유클리디안 거리라 하면, 다음과 같은 관찰이 성립함을 알 수 있다.

[관찰 1] 거리  $D_{HR}$ 은 각각  $HR_1$ 과  $HR_2$ 에 속해 있는 데이터 포인트 쌍들의 거리 중 최소가 되는 거리 보다 더 가깝다. 즉:

$$D_{HR}(HR_1, HR_2) \leq \min_{P_1 \in HR_1, P_2 \in HR_2} d(P_1, P_2)$$

여기에서  $P_1$ 과  $P_2$ 는 각각  $HR_1$ 과  $HR_2$ 에 속해 있는 데이터 포인트이다.

[관찰 2] 하이퍼 사각형  $HR_1$ 이 다른 하이퍼 사각형  $HR_2$ 에 포함되는 경우( $HR_2 \supseteq HR_1$ ), 임의의 하이퍼 사각형  $HR_3$ 에 대하여 다음 (그림 4)에서 보는 바와 같이  $D_{HR}(HR_3, HR_2) \leq D_{HR}(HR_3, HR_1)$ 이 성립한다.



(그림 4) 한 하이퍼 사각형이 다른 하이퍼 사각형에 포함될 경우의 두 하이퍼 사각형 간의 거리

[관찰 1]과 [관찰 2]에 기초하여  $D_{HR}$ 과  $D_{SEG}$ 에 대하여 하한 관계를 설정하는 다음 2개의 정리를 유도할 수 있다.

**[정리 3] 하한 거리(Low-bounding distance) :**  $D_{HR}(SEG_q, SEG_t) \leq D_{SEG}(SEG_q, SEG_t)$ .

질의 세그먼트  $SEG_q$ 와 데이터베이스에 있는 비디오 세그먼트  $SEG_t$ 를 각각 표현하는 두 하이퍼 사각형 사이의 거리  $D_{HR}(SEG_q, SEG_t)$ 은 두 세그먼트 사이의 거리  $D_{SEG}(SEG_q, SEG_t)$ 의 하한(low-bound)이다. 따라서 다음 식이 성립한다.

$$D_{HR}(SEG_q, SEG_t) \leq D_{SEG}(SEG_q, SEG_t)$$

**[증명]** 두 세그먼트  $SEG_q, SEG_t$ 가 각각  $k, l$ 개의 데이터 포인트를 갖고,  $P_q$ 와  $P_t$ 는 각각  $SEG_q$ 와  $SEG_t$ 내에 포함된 임의의 데이터 포인트라 하자. 일반성을 상실하지 않고  $k \leq l$ 라 가정할 수 있다( $k > l$ 인 경우에는  $SEG_q$ 와  $SEG_t$ 를 교환하면 된다.). 그러면, 다음이 성립한다 :

$$D_{VS}(SEG_q, SEG_t) = \min_{1 \leq j \leq l-k+1} D_{mean}(SEG_q[1:k], SEG_t[j:j+k-1]) \geq \min_{P_q \in SEG_q, HR, P_t \in SEG_t, HR} d(P_q, P_t)$$

[관찰 1]에 의하여 다음 식을 유도할 수 있다.

$$D_{HR}(SEG_q, SEG_t) \leq \min_{P_q \in SEG_q, HR, P_t \in SEG_t, HR} d(P_q, P_t)$$

따라서,  $D_{HR}(SEG_q, SEG_t) \leq D_{SEG}(SEG_q, SEG_t)$ 이 성립한다.

**[정리 4] 하한 거리(Low-bounding distance) :**  $D_{HR}(SEG_q, CL) \leq \min_{SEG_t \in CL} D_{HR}(SEG_q, SEG_t)$ .

비디오 클러스터  $CL$ 이 하나 혹은 복수 개의 비디오 세그먼트를 포함하고 있다고 하자. 질의 세그먼트  $SEG_q$ 와 테

이터베이스에 있는 비디오 클러스터  $CL$ 을 각각 표현하는 두 하이퍼 사각형 사이의 거리  $D_{HR}(SEG_q, CL)$ 은 세그먼트  $SEG_q$ 와  $CL$ 에 포함되어 있는 임의의 세그먼트  $SEG_t$  ( $SEG_t \in CL$ )를 각각 표현하는 두 하이퍼 사각형 사이의 거리  $D_{HR}(SEG_q, SEG_t)$ 의 하한(low-bound)이다. 따라서 다음 식이 성립한다 :

$$D_{HR}(SEG_q, CL) \leq \min_{SEG_t \in CL} D_{HR}(SEG_q, SEG_t)$$

**[증명]** 비디오 클러스터의 정의에서 비디오 클러스터의 하이퍼 사각형은 비디오 클러스터 내의 모든 비디오 세그먼트들의 하이퍼 사각형을 내포(tightly bounding)한다. [관찰 2]에 의하여  $SEG_t \in CL$ 인 모든  $SEG_t$ 에 대하여  $SEG_t, HR \subseteq CL.HR$ 이므로, 거리  $D_{HR}(SEG_q, CL)$ 은 모든  $SEG_t$  ( $SEG_t \in CL$ )에 대하여 거리  $D_{HR}(SEG_q, SEG_t)$ 보다 작거나 같다.

[정리 3]과 [정리 4]를 기초로, 거리 함수  $D_{HR}(SEG_q, SEG_t)$ 과  $D_{HR}(SEG_q, CL)$ 을 ‘과오 누락’없이 데이터베이스로부터 질의와 유사하지 않은 비디오 세그먼트들을 검색에서 제외시킬 수 있다. 이는 이 거리 함수들이 거리  $D_{SEG}(SEG_q, SEG_t)$ 에 대하여 하한 거리를 제공하고 있기 때문이다. 주어진 질의 세그먼트  $SEG_q$ 에 대하여 질의 “ $D_{SEG}(SEG_q, SEG_t) \leq \epsilon$ 을 만족시키는 비디오 세그먼트  $SEG_t$ 의 집합을 구하시오.”를 고려해 보자. [정리 3]와 [정리 4]에 의하여 다음 식이 성립한다 :

$$D_{HR}(SEG_q, CL) \leq D_{HR}(SEG_q, SEG_t) \leq D_{SEG}(SEG_q, SEG_t) \leq \epsilon, \text{ for all } SEG_t \in CL$$

먼저 [정리 4]를 사용하여 질의에 무관한 비디오 클러스터들을 대상에서 제외하여 후보 비디오 클러스터들을 구한다. 다음에, 후보 비디오 클러스터 내의 각 비디오 세그먼트들을 [정리 3]을 사용하여 검사하여 질의와 무관한 비디오 세그먼트들을 대상에서 제외하고 후보 비디오 세그먼트들을 구한다. 이러한 후보 비디오 세그먼트들은 최종 해 집합을 얻기 위하여 거리 함수  $D_{SEG}(SEG_q, SEG_t)$ 를 사용하여 정제 단계에서 최종적으로 검사된다. 질의를 처리하기 전에 먼저 각 비디오 스트림의 비디오 프레임들로부터 특성 벡터를 추출하고 각 비디오 스트림을 다차원 데이터 시퀀스로 사상하는 전처리 단계를 거친다. 각 다차원 데이터 시퀀스는 세그멘테이션과 클러스터링 단계를 거쳐 분할되고 군집화 된다. 그 후 데이터 베이스로부터 질의와 유사한 비디오 세그먼트를 선택하기 위하여 유사성 검색 과정이 수행된다. 하이퍼사각형 검색 알고리즘에 대하여 다음에 간단히 기술되어 있다.

**하이퍼사각형 검색 알고리즘**

**입력 :** 데이터 베이스에 있는 비디오 클러스터 집합, 질의 비디오 스트림, 임계값

**출력 :** 비디오 세그먼트 해집합

**선행처리단계 :**

- 0-1. 비디오 스트림의 각 프레임으로부터 특징 값(RGB 칼라 속성)을 추출한다.
- 0-2. 특징 값을 기반으로 하여 각 프레임을 다차원 공간 상의 벡터로 표현한다.
- 0-3. (Optional) 생성된 벡터의 차원이 고차원인 경우 차원 축소를 한다.
- 0-4. 비디오 스트림을 다차원 공간 상에서 벡터의 시퀀스로 나타낸다.
- 0-5. 다차원 공간의 시퀀스를 비디오 세그먼트로 분할하고 비디오 세그먼트를 하이퍼사각형으로 표현한다.
- 0-6. 동일 시퀀스내의 유사한 비디오 세그먼트를 비디오 클러스터로 군집화한다.
- 0-7. 위의 비디오 클러스터를 R\*-Tree 을 사용하여 인덱싱하고 데이터베이스에 저장한다.

**단계 1 :** /\* 질의 비디오 세그먼트 생성\*/

- 1-1. 질의 비디오 스트림의 각 프레임으로부터 특징 값(RGB 칼라 속성)을 추출한다.
- 1-2. 특징 값을 기반으로 하여 각 프레임을 다차원 공간 상의 벡터로 표현한다.
- 1-3. (Optional) 생성된 벡터의 차원이 고차원인 경우 차원 축소를 한다.
- 1-4. 비디오 스트림을 다차원 공간 상에서 벡터의 시퀀스로 나타낸다.
- 1-5. 다차원 공간의 시퀀스를 질의 비디오 세그먼트로 분할하여 하이퍼사각형으로 나타낸다.

**단계 2 :** /\* R\*-Tree 인덱스 검색 -1<sup>st</sup> Pruning \*/

- 2-1. 단계 1-5에서 생성된 각 질의 세그먼트에 대하여 : 데이터베이스에 저장된 비디오 클러스터 집합에 대하여 인덱스 검색을 하여 질의 비디오 세그먼트와 DHR 거리가 임계값 안에 있는 후보 비디오 클러스터를 검색한다.
- 2-2. 단계 2-1에서 생성된 각 후보 비디오 클러스터에 대하여 :
  - 2-2-1. 각 후보 비디오 클러스터에 포함된 비디오 세그먼트 집합에 대하여 : 질의 비디오 세그먼트와 DHR 거리가 임계값 안에 있는 비디오 세그먼트를 검색한다.
  - 2-2-2. 단계 2-2-1에서 검색한 비디오 세그먼트들로부터 후보 비디오 세그먼트 집합을 생성한다.

**단계 3 :** /\* 2<sup>nd</sup> Pruning \*/

- 3-1. 단계 2-2-2에서 생성된 각 후보 비디오 세그먼트에 대하여 :
  - 3-1-1. 질의 비디오 세그먼트와 DSEG 거리가 임계값 보다 큰 비디오 세그먼트를 검색한다.
  - 3-1-2. 단계 3-1-1에서 검색한 비디오 세그먼트들로부터 비디오 세그먼트 해 집합을 생성한다.

**단계 4 :** 비디오 세그먼트 해 집합을 Return

**4.2 대표 프레임 검색기법**

이 절에서는 먼저 하이퍼 사각형의 기하학적 특성을 고려하여 각 비디오 세그먼트와 클러스터로부터 대표 포인트를 선택하는 간단하고 효과적인 방법을 소개하고, 이 대표 포인트를 사용하여 검색하는 대표 프레임 검색기법에 관하

여 기술한다. 다음에 비디오 세그먼트와 클러스터에서 대표 포인트를 선택하는 방법을 기술한다.  $k$ 개의 포인트  $P_j (1 \leq j \leq k)$ 를 포함하고 있는 비디오 세그먼트  $SEG$ 를 고려해 보자.  $SEG$ 의 대표 포인트  $RF_{SEG}$ 는 다음과 같이 정의된다 :  $RF_{SEG} = (Pt | Pt \in \mathcal{S}, d(Pt, P_c) = \min_{1 \leq j \leq k} d(P_j, P_c))$ . 여기에서  $\mathcal{S}$ 는  $SEG$ 내에 있는 데이터 포인트들의 집합이고,  $P_c$ 는  $SEG.HR$ 의 중심이며 다음 식으로 계산된다 :

$$P_c = \left( \frac{SEG.HR.L^1 + SEG.HR.H^1}{2}, \dots, \frac{SEG.HR.L^n + SEG.HR.H^n}{2} \right).$$

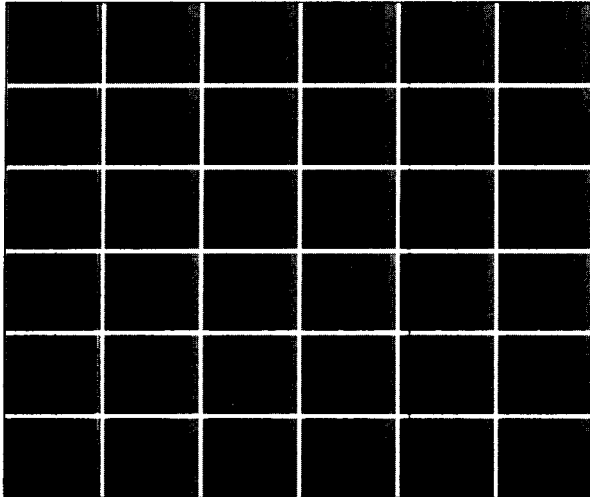
또한,  $r$ 개의 비디오 세그먼트  $SEG_j (1 \leq j \leq r)$ 를 포함하고 있는 비디오 클러스터  $CL$ 를 고려해 보자.  $CL$ 의 대표 포인트  $RF_{CL}$ 는 다음과 같이 정의된다 :  $RF_{CL} = (Pt | Pt \in \mathcal{S}, d(Pt, P_c) = \min_{1 \leq j \leq r} d(RF_{SEG_j}, P_c))$ . 여기에서  $\mathcal{S}$ 는  $CL$ 내에 있는 비디오 세그먼트들의 대표 포인트들의 집합이고,  $RF_{SEG_j}$ 는  $CL$ 내에 있는  $j$ 번째 비디오 세그먼트의  $RF_{SEG}$ 이며,  $P_c$ 는  $CL.HR$ 의 중심이며 다음 식으로 계산된다 :

$$P_c = \left( \frac{CL.HR.L^1 + CL.HR.H^1}{2}, \dots, \frac{CL.HR.L^n + CL.HR.H^n}{2} \right).$$

하이퍼 사각형의 중심으로부터 가장 가까운 포인트는 항상 그렇지는 않지만 비디오 세그먼트나 비디오 클러스터내의 모든 점들의 특성을 비교적 잘 대표한다고 볼 수 있다. 기본적인 아이디어는 이러한 대표 포인트가 유사성 검색에서 사용된다면 시작이나 끝 프레임이 대표 프레임으로 선택되는 경우보다 하이퍼 사각형의 기하학적 특성을 보다 잘 표현함으로 검색 효율이 증가하리라는 것이다. 본 논문에서 실험한 결과, 실제로 대표 포인트에 기초한 유사성 검색 기법은 전통적인 키 프레임 검색 기법 보다 상당히 향상된 효율을 보여 주고 있다. 대표 포인트를 [정의 2]와 [정의 3]에 적용하면 비디오 세그먼트와 비디오 클러스터의 정의를 각각  $SEG = \langle sid, offset, HR, RF_{SEG} \rangle$ 과  $CL = \langle cid, slist, HR, RF_{CL} \rangle$ 로 확장할 수 있다.

다음 (그림 5)는 한 인물이 벽 기둥 사이에서 나타났다가 사라지려고 하는 장면을 담은 하나의 비디오 세그먼트를 나타낸 것이며 36개의 프레임으로 구성되어 있다. 샘플링 비율은 초당 6 프레임으로 하였다. 이 그림에서 보는 바와 같이 시작 프레임을 키 프레임으로 선택하는 것은 장면 속의 인물이 아직 보이지 않기 때문에 합리적이지 못하다. 마지막 프레임 역시 적당한 선택이라고 보기 어렵다. 세그먼트

트의 중간 부근에 있는 프레임을 대표 프레임으로 선택하는 것이 합리적이며, 제안한 기법은 비교적 이에 적합한 것으로 여겨진다.



(그림 5) 비디오 세그먼트의 대표 프레임 선택

비디오 세그먼트나 비디오 클러스터의 대표 포인트(프레임) 사이의 거리  $D_{RF}$ 는 대표 프레임 검색기법에서 사용되며  $D_{PF}$ 가 기본적으로 다차원 데이터 공간에서 두 데이터 포인트 사이의 유클리디안 거리이므로 간단하게 구해질 수 있다.  $SEGCL_1$ 과  $SEGCL_2$ 가 각각 데이터 공간 내의 임의의 비디오 세그먼트나 클러스터라 할 때 거리  $D_{RF}$ 는 다음 식으로 정의된다.

$$D_{RF}(SEGCL_1, SEGCL_2) = d(SEGCL_1.RF, SEGCL_2.RF)$$

대표 프레임 검색기법에서 검색은 비디오 세그먼트와 비디오 클러스터의 대표 포인트에 기초하고 있다. 하이퍼사각형 검색기법과는 달리 각 비디오 세그먼트의 대표 포인트만이 유사성 검색 처리를 위하여 인덱스된다. 검색 알고리즘은 다음과 같다.

**대표프레임 검색 알고리즘**

**입력 :** 데이터 베이스에 있는 비디오 클러스터 집합, 질의 비디오 스트림, 임계값  
**출력 :** 비디오 세그먼트 해집합  
**선행처리단계 :**  
 0-1~0-4. (하이퍼사각형 검색 알고리즘과 동일)  
 0-5. 다차원 공간의 시퀀스를 비디오 세그먼트로 분할하고 비디오 세그먼트를 대표 포인트로 표현한다.  
 0-6~0-7. (하이퍼사각형 검색 알고리즘과 동일)  
**단계 1 :** /\* 질의 비디오 세그먼트 생성\*/  
 1-1~1-4. (하이퍼사각형 검색 알고리즘과 동일)  
 1-5. 다차원 공간의 시퀀스를 질의 비디오 세그먼트로 분할하여 대표 포인트로 나타낸다.

**단계 2 :** /\* R-Tree 인덱스 검색 - 1<sup>st</sup> Pruning \*/  
 2-1. 단계 1-5에서 생성된 각 질의 세그먼트의 대표 포인트에 대하여 :  
 데이터베이스에 저장된 비디오 클러스터 집합에 대하여 인덱스 검색을 하여 질의 비디오 세그먼트와의 거리가 임계값 안에 있는 후보 비디오 클러스터를 검색한다.  
 2-2. 단계 2-1에서 생성된 각 후보 비디오 클러스터에 대하여 :  
 2-2-1. 각 후보 비디오 클러스터에 포함된 비디오 세그먼트 집합에 대하여 :  
 질의 비디오 세그먼트와  $D_{RF}$  거리가 임계값 안에 있는 비디오 세그먼트를 검색한다.  
 2-2-2. 단계 2-2-1에서 검색한 비디오 세그먼트들로부터 후보 비디오 세그먼트 집합을 생성한다.  
**단계 3 :** /\* 2<sup>nd</sup> Pruning \*/  
 (하이퍼사각형 검색 알고리즘과 동일)  
**단계 4 :** 비디오 세그먼트 해 집합을 Return

**5. 실험 및 고찰**

**5.1 실험 환경 및 과정**

제안한 기법의 효과와 성능을 실험적으로 검증하기 위하여 실제 비디오 스트림 뿐만 아니라 프래탈 함수를 사용해서 생성된 가상 데이터 시퀀스를 이용하여 실험을 수행하였다. 이는 실 세계의 데이터 세트는 일반적으로 크기와 다양한 형태의 실험 파라미터 설정에 있어 제한이 있는 반면에 가상으로 생성된 데이터 세트는 보다 유연한 환경에서 실험이 가능하기 때문이다.

비디오 데이터 스트림은 다수의 TV 뉴스, 드라마 및 다큐멘터리 필름들로 구성되어 있다. 비디오 스트림에 대한 데이터 시퀀스는 다양한 비디오 데이터 소스의 각 프레임의 픽셀로부터 색상 특징(RGB) 값을 추출하고 그들을 평균함으로 생성되었다. RGB특징들의 값의 범위는 [0, 255]이며 이를 255로 나누면  $[0, 1]^3$  단위 큐브 내에 포함되게 된다. 따라서, 각 프레임은 단위 큐브 내의 한 점으로 사상된다. 실험 시스템은 Windows 2000 Server 환경의 Microsoft VC++를 사용하여 구현되었다. 실험에서 사용된 모든 데이터 세트는 편의상 3차원 데이터 세트이나 제안된 기법은 차원을 제한하지 않는다. 실험 데이터 세트의 구성은 <표 1>에 나타나 있다.

<표 1> 실험 데이터 세트의 구성

데이터 세트	가상 스트림	비디오 스트림
시퀀스 개수	21,500	7,350
시퀀스 길이	64-1024 points	60-850 frames
임계 값 ( $\epsilon$ )	0.01~0.35	0.01~0.35
차 원	3	3
질의 의 수	100	50



질의 시퀀스의 수는 가상 데이터에 대하여 100, 실제 비디오 데이터에 대하여 50개이다. 복수 개의 질의 결과로부터 최종 결과를 얻기 위하여 평균하였다. 임계 값은 0.01로부터 0.35까지의 범위로부터 선택되었다. 이 범위는 본 실험에서의 검색 공간이 각 차원에 대하여 [0,1]의 범위로 정규화 되었고, 임계 값 0.35은 직경 0.70인 검색 구(sphere)에 해당되므로 적절하다고 판단된다. 평가는 주로 유사성 검색 분야에서 잘 알려진 정확율과 재현율, 그리고 처리 속도에 대하여 이루어졌다. 질의에 의하여 검색된 세그먼트의 집합을  $Set(ret)$ 라 하고, 질의와 관련된 세그먼트 해 집합을  $Set(rel)$ 라 하자. 집합  $S$ 의 요소의 수를  $|S|$ 라 표기하면 정확율과 재현율은 다음 식으로 표시된다.

$$\text{정확율} = \frac{|Set(ret) \cap Set(rel)|}{|Set(rel)|},$$

$$\text{재현율} = \frac{|Set(ret) \cap Set(rel)|}{|Set(ret)|}$$

제안한 기법의 효과를 측정하기 위하여 검색의 결과를 평가하는 근거를 제공하는 그라운드 트루스(ground truth)를 설정할 필요가 있다. 일반적으로 비디오 데이터는 수많은 프레임으로 구성된 대용량의 자료이다. 따라서 인간이 특정 비디오 데이터 세그먼트의 쌍이 얼마나 유사한지를 결정하기는 쉽지 않고, 또한 이 경우에는 관찰자의 주관이 개입될 여지가 매우 높다. 본 실험에서는 거리 척도  $D_{SEG}$ 를 사용하는 순차 스캐닝(sequential scanning) 방법에 의해 검색된 세그먼트 집합을 그라운드 트루스로 가정한다.

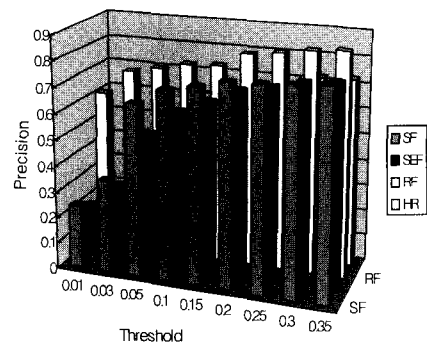
5.2 실험 결과 분석

본 논문에서의 유사성 검색은 주어진 질의 시퀀스의 각 세그먼트에 대해서 지정된 임계 값 안에 포함되는 유사한 세그먼트들을 데이터베이스에서 찾는 것이다. 제안한 방법의 효과를 평가하기 위해서 정확률, 재현율, 그리고 처리 속도에 대하여 기존의 방법들과 비교하였다. 비교 대상이 되는 기법들은 대표 프레임으로써 시작 프레임을 채택한 시작 프레임 검색기법(SF)과 시작과 끝 프레임 둘 다를 채택한 시작/끝 프레임 검색기법(SEF), 그리고 본 논문에서 제안한 대표 프레임 검색기법(RF)과 하이퍼사각형 검색기법(HR)이다.

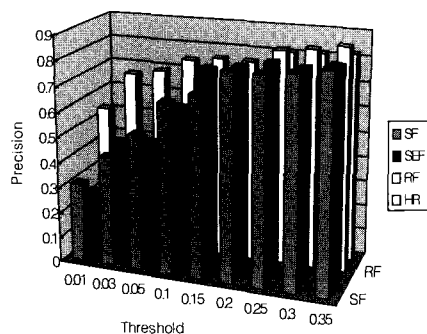
**정확율 :** (그림 6)과 (그림 7)은 각 기법의 정확율을 비교한 것이다. 비교는 가상으로 생성된 데이터와 실제 비디오 데이터에 대하여 수행되었다. 그래프에서 볼 수 있는 것처럼

대표 프레임 검색기법이 가장 높은 정확율을 보여 주며, 가상 시퀀스에 대해서 0.61~0.86, 실제 비디오 데이터에 대해서 0.54~0.88이다. 대표 프레임 검색기법은 임계 값의 전 범위에서 다른 기법들에 비하여 우수하다.

한편, 시작/끝 프레임 검색기법과 하이퍼사각형 검색기법은 비교적 낮은 정확율을 보여주고 있다. 이 기법들에서는 검색을 위하여 두 개의 포인트 - 시작/끝 프레임 검색기법에서는 시작/끝 포인트, 하이퍼사각형 검색기법에서는 하이퍼사각형의 저점과 고점이 관여하며, 이는 질의와 무관한 세그먼트가 상대적으로 많이 검색되게 한다. 이 두 가지 기법의 정확율이 낮은 것은 이러한 이유 때문이다. 하이퍼사각형 검색기법의 정확율은 가상 시퀀스에 대해서 0.18~0.72 이고 실제 비디오에 대해서는 0.25~0.82이다. 임계 값이 0.1이상인 경우에는 가상 시퀀스에 대해서는 0.47이상, 실제 비디오에 대해서는 0.51이상이다. 이 정도의 정확율은 실제 환경에서 상당히 유용하게 사용될 수 있다고 여겨진다. 또한 실제 비디오가 가상 시퀀스에 비하여 전반적으로 높은 정확율을 보여주고 있다. 이것은 실 세계의 비디오 데이터가 샷 경계에서 자연스럽게 분할되는 경향을 보이기 때문이다.



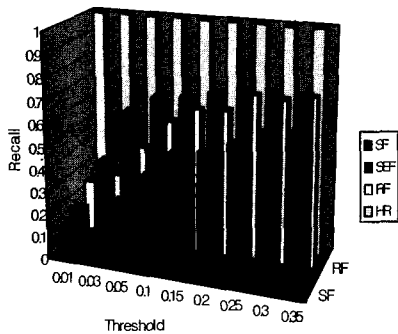
(그림 6) 가상으로 생성된 시퀀스에 대한 정확율



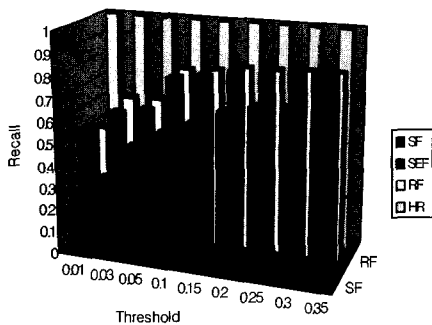
(그림 7) 실제 비디오 데이터에 대한 정확율

**재현율 :** (그림 8)과 (그림 9)는 가상 시퀀스와 실제 비디오

오 데이터에 대한 각 기법의 재현율을 보여주고 있다. 두 그래프에서 하이퍼사각형 검색기법은 재현율이 모두 1.0이다. 이것은 이 기법이 어떠한 '과오 누락'도 허용하지 않는다는 것을 의미한다. 시작/끝 프레임 검색기법이 다음으로 우수하며 가상 시퀀스에 대해서는 0.17~0.79, 실제 비디오 데이터에 대해서는 0.52~0.87 범위이다. 정확율의 경우와는 달리 시작/끝 프레임 검색기법과 하이퍼사각형 검색기법처럼 검색에 두 개의 포인트가 관여하는 기법들은 검색에 하나의 포인트만 사용하는 기법들에 비해서 높은 재현율을 보여 주고 있다. 일반적으로 정확율과 재현율 사이에는 트레이드 오프(trade-off)가 있다고 여겨진다. 정확율이 높아지면 재현율은 낮아지고 반대로 정확율이 낮아지면 재현율은 높아진다. 대표 프레임 검색기법은 비교적 우수한 재현율을 보여주고 있다. 가상 시퀀스에 대해서는 0.24~0.74, 실제 비디오 데이터에 대해서는 0.49~0.84 범위이다. 실제 비디오 데이터의 경우 대표 프레임 검색기법의 재현율은 시작/끝 프레임 검색기법에 매우 근접하며, 때때로 특정 임계 값 범위에서는 더 우수하다. 그러나 시작 프레임 검색기법은 가상 시퀀스와 실제 비디오 모두에 대해서 매우 낮은 재현율을 보여 주고 있으며 따라서 재현율에 민감한 분야에서는 사용되기 어렵다.



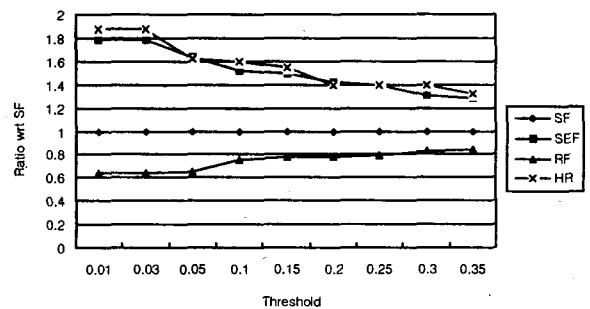
(그림 8) 가상으로 생성된 시퀀스에 대한 재현율



(그림 9) 실제 비디오 데이터에 대한 재현율

정확율과 재현율의 실험으로부터 정확율에 대해서는 대표 프레임 검색기법이 가장 바람직한 반면, 재현율에 대해서는 하이퍼사각형 검색기법이 가장 우수하다고 결론지을 수 있다.

처리속도 : (그림 10)은 비디오 데이터 세트에서의 4가지 방법의 평균 질의 처리 속도를 보여준다. SF 기법을 기준(1.0)으로 하였을 때, 각 기법의 상대적인 처리 속도의 비율을 나타낸 것이다. 실험 결과, 가장 빠른 처리 속도를 보여준 기법은 RF 기법으로써 SF 기법에 비하여 0.64~0.84의 처리 속도를 보여주고 있다. 이는 RF 기법의 정확율이 다른 기법들보다 우수하므로 후속 정제 과정에서 처리해야 하는 세그먼트 수를 줄일 수 있어 처리 오버헤드가 감소하기 때문이다. SEF 기법과 HR 기법의 경우, 처리 속도 비율이 각각 1.28~1.78과 1.32~1.87로써 시작/끝 프레임 및 저점/고점에 해당하는 두 포인트를 검색에 이용하기 때문에 처리 속도 면에서 다소 느리다. 그러나 HR 기법의 경우에는 SEF 기법과 거의 비슷한 처리 속도이지만 재현율 면에서 훨씬 뛰어나기 때문에 실제 응용에서 유용하게 사용될 수 있다. 한편, 모든 기법에서 임계 값이 커질수록 처리 속도의 차이가 줄어드는 경향을 보이는데, 이는 임계 값이 커지면 수많은 세그먼트들이 검색 대상에 포함되므로 오버헤드가 기하급수적으로 늘어나게 되어 상대적으로 처리 속도의 차이가 줄어들게 되기 때문이다.



(그림 10) 실제 비디오 데이터에 대한 처리속도

5. 결 론

시퀀스 데이터 세트에 대한 유사성 검색은 시계열 데이터, 디지털, 아날로그 신호와 같은 다양한 애플리케이션 영역에서 광범위하게 적용될 수 있기 때문에 데이터베이스 분야에서 잠재 가능성이 높은 분야 중 하나이다. 본 논문에서는 일 차원 시계열 데이터에 대한 전통적인 유사성 검색 기법을 다차원 데이터 시퀀스로 확장하였고, 두 가지의 유

사성 검색 기법을 제안하였다. 첫째는 하이퍼사각형 검색기법으로써 질의가 비디오 세그먼트와 클러스터의 하이퍼 사각형에 기초하여 처리되며 정해성을 보장하는 기법이다. 다음은 대표 프레임 검색기법으로써 비디오 세그먼트와 클러스터의 대표 포인트에 기초한 검색 기법이다. 유사성을 측정하기 위하여 두 하이퍼 사각형 간의 거리 함수  $D_{HR}$  과 비디오 세그먼트와 클러스터의 대표 포인트 사이의 거리 함수인  $D_{RF}$  를 정의하였다. 하이퍼사각형 검색기법은 정해성을 보장하며 상당한 수준의 정확도를 유지하고 있으며, 대표 프레임 검색기법은 전통적인 기법에 비하여 매우 높은 정확도와 상당한 수준의 재현율을 유지하고 있다. 따라서 이 기법은 정확율이 재현율보다 더 중요하게 여겨지는 분야에서 유용하게 사용될 수 있다.

이 논문에서 제안한 기법들은 데이터 형식이 다차원 시퀀스로 표현될 수 있는 애플리케이션 영역에서 널리 사용될 수 있으리라 기대된다. 본 논문에서 강조된 분야는 대용량 비디오 데이터 스트림에 대한 유사성 검색이지만 다른 영역에서도 역시 사용될 수 있을 것이라 여겨진다. 향후 연구로써 음성 신호 매칭(voice signal matching)과 영역 기반 이미지 검색(region-based image retrieval)과 같이 각 분야의 고유한 특성을 고려한 다양한 애플리케이션 영역에 대하여 제안한 방법을 적용하는 문제에 대하여 연구할 계획이다.

참 고 문 헌

[1] R. Agrawal, C. Faloutsos, A. Swami, "Efficient Similarity Search in Sequence Databases," Proc. of Foundations of Data Organizations and Algorithms, Illinois, pp.69-84, 1993.  
 [2] N. Beckmann, H. Kriegel, R. Schneider, B. Seeger, "The R\*-tree : an efficient and robust access method for points and rectangles," Proc. of ACM SIGMOD, New Jersey, pp.322-331, 1990.  
 [3] S. Berchtold, D. Keim, H. Kriegel, "The X-tree : an index structure for high-dimensional data," Proc. of Int'l Conference on VLDB, India, pp.28-39, 1996.  
 [4] C. Faloutsos, M. Ranganathan, Y. Manolopoulos, "Fast Subsequence Matching in Time-Series Databases," Proc. of ACM SIGMOD, Minnesota, pp.419-429, 1994.  
 [5] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, P. Yanker, "Query by Image and Video Content :

The QBIC System," IEEE Computer, Vol.28, No.9, pp.23-32, 1995.  
 [6] A. Guttman, "R-trees : a dynamic index structure for spatial searching," Proc. of ACM SIGMOD, Massachusetts, pp.47-57, 1984.  
 [7] A. Hampapur, A. Gupta, B. Horowitz, C. F. Shu, C. Fuller, J. Bach, M. Gorkani, R. Jain, "Virage Video Engine," Proc. of SPIE : Storage and Retrieval for Image and Video Databases V, San Jose, pp.188-197, 1997.  
 [8] A. K. Jain, A. Vailaya, X. Wei, "Query by Video Clip," Multimedia Systems, Vol.7, pp.369-384, 1999.  
 [9] N. Katayama, S. Satoh, "The SR-tree : an index structure for high-dimensional nearest neighbour queries," Proc. of ACM SIGMOD, Arizona, pp.369-380, 1997.  
 [10] E. J. Keogh, K. Chakrabarti, S. Mehrotra, M. J. Pazzani, "Locally Adaptive Dimensionality Reduction for Indexing Large Time Series Databases," Proc. of ACM SIGMOD, California, pp.151-162, 2001.  
 [11] S. W. Kim, S. H. Park, W. W. Chu, "An Index-Based Approach for Similarity Search Supporting Time Warping in Large Sequence Databases," Proc. of IEEE Int'l Conference on Data Engineering, 2001. pp.607-614,  
 [12] S. L. Lee, S. J. Chun, D. H. Kim, J. H. Lee, C. W. Chung, "Similarity Search for Multidimensional Data Sequences," Proc. of IEEE Int'l Conference on Data Engineering, California, pp.599-608, 2000.  
 [13] S. L. Lee, C. W. Chung, "Hyper-Rectangle Based Segmentation and Clustering of Large Video Data Sets," Information Sciences, Vol.141, No.1-2, pp.139-168, 2002.  
 [14] John Chung-Mong Lee, Qing Li, and Wei Xiong, "VIMS : A Video Information Management System," Multimedia Tools and Applications, Vol.4, No.1, pp.7-28, 1997.  
 [15] Virginia E. Ogle and Michael Stonebraker, "Chabot : Retrieval from a Relational Database of images," IEEE Computer, Vol.28, No.9, pp.40-48, 1995.  
 [16] J. H. Oh, K. A. Hua, "Efficient and Cost-effective Techniques for Browsing and Indexing Large Video Databases," Proc. of ACM SIGMOD, pp.415-426, Dallas, 2000.  
 [17] D. Rafiei, A. Mendelzon, "Similarity-Based Queries for Time Series Data," Proc. of ACM SIGMOD, Arizona, pp. 13-25, 1997.  
 [18] J. R. Smith, S. F. Chang, "VisualSEEK : A Fully Automated Content-based Image Query System," Columbia

University, URL : <http://www.ctr.columbia.edu/VisualSEEk>.

- [19] Y. Taniguchi, A. Akutsu, Y. Tonomura, "PanoramaExcerpts : Extracting and Packing Panoramas for Video Browsing," Proc. of ACM Multimedia, Washington, pp. 427-436, 1997.
- [20] B. K. Yi, C. Faloutsos, "Fast Time Sequence Indexing for Arbitrary Lp Norms," Proc. of Int'l Conference on VLDB, Cairo, pp.385-394, 2000.
- [21] H. J. Zhang, J. Wu, D. Zhong, S. W. Smoliar, "An Integrated System for Content-Based Video Retrieval and Browsing," Pattern Recognition, Vol.30, pp.643-653, 1997.



### 이 석 통

e-mail : [slee@hufs.ac.kr](mailto:slee@hufs.ac.kr)

1984년 연세대학교 기계공학과 학사

1993년 연세대학교 산업공학과 전자  
계산전공 석사

2001년 한국과학기술원(KAIST) 정보 및  
통신공학과 박사

1984년~1995년 한국IBM 소프트웨어 연구소 선임연구원

1995년~2002년 안산1대학 웹프로그래밍과 조교수

2002년~현재 한국외국어대학교 산업정보시스템공학부 부교수

관심분야 : 데이터베이스, 데이터웨어하우스, 데이터마이닝, 시  
계열 데이터 검색, 멀티미디어 데이터베이스