論文99-36T-3-14

VRTEC : 내용 기반 비디오 질의를 위한 다단계 검색 모델

(VRTEC : Multi-step Retrieval Model for Content-based Video Query)

金 昌 龍 *

(Chang Ryong Kim)

요 약

본 논문은 내용 기반 비디오 질의를 위한 데이터 모델과 검색 방법을 제안한다. 하나의 비디오를 같은 길이의 프레임(frame)들의 집합 즉 비디오-윈도우로 나눈 후에 각각의 비디오-윈도우를 다차원 공간의 한 점으로 사상시킨다. 인접한 비디오-윈도우를 연결하면 하나의 비디오는 다차원 공간에서의 하나의 궤적(trajectory)이 된다. 두 비디오-윈도우의 유사성은 두 점의 유클리디안 거리로 정의되며, 비디오 단편(segment)의 유사성 비교는 궤적을 비교함으로써 검사한다. 여과(filtering), 정제(refinement) 과정을 가지는 새로운 검색 방법을 개발한다. 새로운 검색 방법을 여과/정제 과정이 없는 질의 방법과 비교하는 실험을 한 결과 질의 결과가 정확하고, 질의 처리 속도는 약 4.7배 향상되었다.

Abstract

In this paper, we propose a data model and a retrieval method for content-based video query. After partitioning a video into frame sets of same length which is called video-window, each video-window can be mapped to a point in a multidimensional space. A video can be represented a trajectory by connection of neighboring video-window in a multidimensional space. The similarity between two video-windows is defined as the euclidean distance of two points in multidimensional space, and the similarity between two video segments of arbitrary length is obtained by comparing corresponding trajectory. A new retrieval method with filtering and refinement step is developed, which returns correct results and makes retrieval speed increase by 4.7 times approximately in comparison to a method without filtering and refinement step.

I. 서 론

최근 들어 비정형 문서 관리 시스템, 설계 및 엔지니어링 시스템, 지리 정보 시스템, 의료 정보 시스템, 전자 신문, 전자 도서관, 홈쇼핑 등과 같은 새로운 응용 분야들이 각광을 받고 있으며, 이들 응용 분야에서 영상과 비디오 데이터를 중심으로 한 멀티미디어 검색에 대한 요구가 증가하고 있다. 멀티미디어에 대한 검색 과정은 멀티미디어 데이터로부터 추출된 특성과 검색에 포함된 멀티미디어에서 추출한 특성들을 비교하여 유사성을 계산하는 과정이다. 따라서 비교 대상이 되는 특성에 대한 연구와 그러한 특성을 추출하는 방법, 그리고 유사성 검사를 위한 방법 등에 대한 연구가 필요하다.

본 논문은 유사성 검사를 위한 방법에 대한 연구 결과이며, 이를 위하여 비디오 단편과 질의 비디오, 대상 비디오란 용어를 다음처럼 정의한다.

* 正會員, 富川大學 情報通信系列

(Department of Information & Communication, Bucheon College)

接受日字: 1999年1月14日, 수정완료일: 1999年3月15日

정의 1. (비디오 단편(*Video Segment*), 질의 비디오(*Query Video*), 대상 비디오(*Target Video*)).

비디오는 연속된 프레임들의 집합이며, 임의의 연속적 프레임의 집합을 비디오 단편이라고 정의한다. 질의 입력에 사용되는 비디오 단편을 질의 비디오, 주어진 비디오 데이터에서 질의 비디오와 길이가 같은 비디오 단편을 대상 비디오라고 정의한다. 질의 비디오를 질의 입력으로 하여 대상 비디오의 집합을 질의 결과로 한다. 샷(shot)과 씬(scene), 비디오(video) 등은 비디오 단편의 특수한 경우이다.

비디오 검색에서는 개별적인 프레임에 대한 검색보다는 샷 또는 씬에 대한 검색이 이루어져야 하며, 이를 위하여 연속적 프레임 집합에 대한 유사성을 검사하는 방법을 개발해야 한다. 질의 비디오는 프레임들의 연속적 집합이므로 질의 비디오의 시간적/공간적 정보가 질의 입력이 되고 질의 결과는 비디오 단편들의 집합 즉, 대상 비디오들의 집합이 된다.

본 논문의 구성은 다음과 같다. II장에서는 비디오 검색에 관련된 관련 연구, 그리고 각 연구 결과의 한계점 또는 문제점에 대하여 살펴본다. III장에서 비디오 검색을 위한 비디오 모델 *WIVE(Window-based Video Model)*를 제안하고 IV장에서는 제안된 비디오 모델을 기반으로 한 비디오 검색 모델 *VRTEC (Video Retrieval by Trajectory Comparison)*을 위한 거리 함수와 검색 알고리즘 등을 제안하며, 수학적인 증명을 통하여 제안된 방법의 적법성을 증명한다. V장에서 실험을 통하여 제안된 방법들을 검증하고 마지막으로 본 연구에 대한 결론 및 향후 연구를 VI장에서 설명한다.

II. 관련 연구

1. 비디오 검색

비디오 검색에 대한 현재 기술력은 주석을 이용한 검색과 대표 프레임의 영상 특징을 이용한 내용 기반 검색[1], 객체와 사건을 기반으로 한 내용 기반 검색[2] 등이 있다. 멀티미디어 구성요소 중 비디오 데이터는 시간적/공간적인 관계를 가진다. 즉, 비디오를 구성하는 각 프레임들은 서로 이웃하는 프레임들간에 시간적 순서를 가지며, 프레임에 등장하는 객체들간에는 공간적인 관계가 있다. 비디오에 대한 검색은 이러한 비디오 데이터의 시공간적 특성을 기반으로 하여 이루어 질 수 있다[3].

비디오 데이터 처리 기술의 목표는, 현재 텍스트 파일을 처리하는 것처럼, 내용을 기반으로 검색을 할 수 있도록 하는 것이다[4]. 비디오 데이터를 기존의 텍스트 데이터처럼 검색하기는 매우 어려우며, 지금까지의 기술력은 비디오 데이터에 대한 주석 기반 검색이나 대표 프레임의 특징을 이용한 검색 등을 지원한다[1][5]. 객체와 사건을 기반으로 한 내용 기반 비디오 검색을 시도한 사례도[2] 있으나 이는 주석 기반 검색과 별다른 차이점을 찾을 수 없다. 주석을 통하는 대신에 다만 자료 구조를 통하여 비디오 검색을 할뿐이다. 비디오 데이터가 바뀌면 자료 구조를 재구성해야 할뿐만 아니라 자료 구조에 포함되지 않은 객체와 사건에 대한 검색은 당연히 실패한다. 영역 지식을 이용하려는 시도에는 파싱 단계에서 주요 객체들과 그들의 특징, 그리고 주요한 사건들을 추출하여 원래의 멀티미디어 데이터와 함께 저장한 후에 이를 통해 사용자 검색을 처리하고자 하는 것이 있다[9][10].

사람의 주관을 배제하기 위하여 시도된 방법이 멀티미디어 데이터 자체로부터 그 데이터가 가지고 있는 특징들을 컴퓨터에 의해 자동적으로 추출하려는 것이다. 즉, 컴퓨터에 의해 자동적으로 카메라 샷의 분할점을 찾아내고, 각각의 샷내에서 카메라 움직임을 자동으로 분류하며, 출현한 객체의 위치라든지 크기, 색상, 모양, 움직임 등을 자동으로 추출해내는 것이다[6][7][8]. 주석기반 검색은 주석에 사람의 주관이 있으며 이것은 비디오 검색을 원하는 일반 사용자의 생각과 다를 수 있으므로 컴퓨터에 의한 자동적인 특징을 추출하는 작업이 필요하다.

본 연구는 비디오 검색을 위한 기존 연구에서 발견된 문제점 중에서 주석기반 검색의 주관적인 부분과 대표프레임의 영상 특징을 이용한 방법의 잘못된 검색결과를 해결하는 것이 목표이다. 즉, 자동적인 특징 추출을 통하여 추출된 특징을 이용하여 검색을 하며, 대표프레임이 아닌 전체 프레임을 사용하여 검색하는 방법을 제안한다.

2. 시계열 데이터의 유사성 검사

시계열 데이터란 연속적인 실수값을 의미한다. 예를 들어 주식값의 변화, 어느 지역의 온도 변화 등이 있다. 시계열 데이터의 유사성 검사를 위한 연구는 검색

공간을 위한 데이터 모델과 거리 함수 설정 등 2가지로 분류할 수 있다. Faloutsos 등은 [11]에서 검색 공간을 위하여 윈도우 개념을 도입하였다. 주어진 신호를 같은 길이의 윈도우로 나눈 후에 임의의 길이의 신호를 검색하는 방법과 알고리즘을 제안하였다. Faloutsos 등이 [12]에서 *Time Warping* 개념을 이용한 거리 함수를, [13]에서 *lower bound*를 이용한 거리 함수를 제안하였고, Davood Rafiei 등은 *m-day moving average*를 이용한 거리 함수를 만들었다[14].

퓨리에 변환(Fourier transform)은 전체의 개략적인 변화의 유사성 검사에 이용된다[14][15][16]. 본 논문에서는 비디오 검색의 여과 과정에 퓨리에 변환을 사용한다. 퓨리에 변환은 시간 영역에서 정의된 데이터를 주파수 영역으로 변환하거나 반대로 주파수 영역의 데이터를 시간 영역으로 변환하는 것으로서 다음과 같은 변환식을 가진다.

$$X_f = \frac{1}{\sqrt{n}} \sum_{t=0}^{n-1} x_t \exp(-j2\Pi ft/n)$$

$$= R(f) + jI(f), \quad where, \ f = 0, 1, 2, \ldots, n-1$$

$$x_t = \frac{1}{\sqrt{n}} \sum_{f=0}^{n-1} X_f \exp(-j2\Pi ft/n)$$

$$= R(t) + jI(t), \quad where, \ t = 0, 1, 2, \ldots, n-1$$

where, $j = \sqrt{-1}$, R: 실수부, I: 허수부

어떤 임의의 벡터가 주어지면 에너지를 구할 수 있다. $S = (X_0, X_1, X_2, \ldots, X_{n-1})$일 때 벡터 S의 에너지는 $E(S) = \sum_{f=0}^{n-1} |X_f|^2$이다.

Ⅲ. 비디오 검색을 위한 비디오 모델(*WIVE*)

비디오 데이터 모델 *WIVE*는 비디오 데이터를 w 차원의 벡터 공간으로 사상시킨다. 비디오 단편 단위로 검색을 하려면 비디오 단편들을 위한 새로운 표현 방법을 개발하여야 하며, 이를 위하여 먼저 비디오-윈도우의 의미를 다음과 같이 정의한다.

정의 2. (비디오-윈도우).

비디오-윈도우(*VW:Video-Window*)를 임의의 변수 t에 대하여 다음과 같이 정의한다.

$$VW(t, w) = \{f_i \in V \mid i = t, t+1, \ldots, t+(w-1)\}$$

where, w: 비디오-윈도우크기,

f_i: i번째 프레임, V: 비디오

비디오 윈도우는 주어진 비디오 데이터를 일정한 길이로 분할한 것이며, 비디오 데이터는 고정길이 분할을 이용하여 임의 길이의 질의 비디오를 검색한다.

본 논문에서 사용하는 심볼과 그 의미는 다음과 같다.

표 1. 심볼과 심볼 정의 요약
Table 1. Summary of Symbols and Definition.

심볼	정의
w	비디오-윈도우 크기
k	선택된 퓨리에 지배계수 개수
$Len(S)$	비디오 단편 S의 길이
M	질의 비디오 궤적길이
q	시간영역의 질의비디오 궤적
a	시간 영역의 대상비디오 궤적
Q	주파수영역의 질의비디오 궤적
A	주파수영역의 대상비디오 궤적

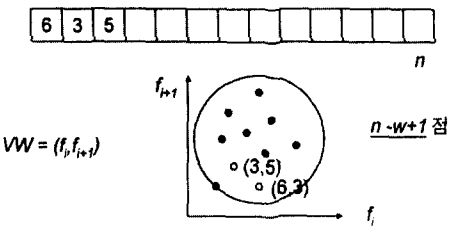


그림 3.1 새로운 비디오 표현
Fig. 3.1 A new video representation.

한 개의 비디오-윈도우는 w차원의 공간에 있는 한 점으로 사상시킬 수 있다. 예를 들어, 만약 $w=2$인 경우에 하나의 비디오를 다차원 공간상으로 사상하면 그림 3.1과 같다. 이때 n은 비디오의 전체 프레임 개수이다.

질의 비디오의 크기가 비디오-윈도우 크기와 같다고 가정하면, 질의 비디오 α와 대상 비디오 β의 유클리디안 공간에서의 거리는 다음과 같이 정의된다.

$$D(\alpha, \beta) = \left(\sum_{i=1}^{w} (a_i - b_i)^2 \right)^{\frac{1}{2}}$$

$$where, \ \alpha = \langle a_1, a_2, \cdots, a_n \rangle \ and$$

$$\beta = \langle b_1, b_2, \cdots, b_n \rangle$$

그림 3.2처럼 두 점 사이의 거리를 계산하여 거리가

짧을수록 유사하다는 결론을 내린다. 멀티미디어에서 가장 중요한 검색 형태인 영역 질의와 최근접 질의는 멀티미디어 데이터를 다차원 점으로 사상시킨 후에 그들간의 거리를 계산함으로써 유사성을 검사한다. 즉, 질의 비디오와의 차이가 주어진 임계값을 넘지 않는 대상 비디오 모두를 선택하면 영역 질의가 되고, 차이가 적은 대상 비디오부터 k개를 선택하면 k-최근접 질의가 된다.

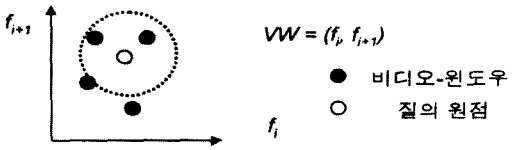


그림 3.2 비디오 검색
Fig. 3.2 Video retrieval.

위에서 정의한 비디오-윈도우는 비디오의 시간적 정보를 가지고 있다. 인접한 비디오-윈도우를 연결하면 비디오는 하나의 궤적으로 사상되고 그 궤적은 $Len(V)-w+1$개의 다차원 점으로 구성되며, 질의 비디오들도 임의의 길이의 궤적으로 표현된다. 예를 들어 질의 비디오가 p개의 프레임으로 구성되어 있다면, $p-w+1$개의 다차원 점으로 구성되는 궤적으로 사상된다(그림 3.3).

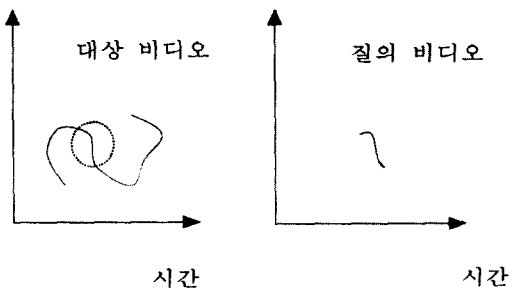


그림 3.3 궤적 비교를 통한 유사성 검사
Fig. 3.3 Similarity test using trajectory comparison.

비디오-윈도우 개념에 따라서 만들어진 비디오 모델의 장점은 두 가지이다. 첫째, 모든 질의 비디오에 대하여 왼쪽-정렬을 하므로 지원해야 할 검색 형태가 단순해지며, 이 사실은 비디오 검색 모델을 위한 알고리즘을 서술할 때 그림 3.4에서 볼 수 있듯이 두 가지 종류에 대한 알고리즘으로 충분하다는 의미이다. 둘째, $Len(Query\ Video)-w+1$개의 점을 비교함으로써 임의 길이의 질의 비디오에 대해서도 검색할 수 있다. 이것은 임의 길이의 질의 비디오에 대응하는 임의 길이의 대상 비디오를 항상 구성할 수 있음을 의미한다. 왼쪽-정렬을 통하여 단순해진 검색 형태는 그림 3.4와 같이 2가지 종류의 검색이 있다.

그림 3.4 비디오 검색의 2가지 종류
Fig. 3.4 2 types of video query.

Ⅳ. 다단계 비디오 검색 모델(*VRTEC*)

전통적인 데이터베이스와는 달리 멀티미디어 데이터베이스에서의 검색은 유사성을 기반으로 한다[17][18]. 예를 들어 "주어진 질의 비디오와 색상이 30% 이상 유사한 대상 비디오를 찾아라"(영역 질의), 그리고 "주어진 질의 비디오와 색상이 유사한 순서로 5개의 단편을 찾아라"(최근접 질의) 등을 생각할 수 있다. 내용 기반 검색 시스템은 영역 질의와 최근접 질의를 효과적으로 지원해야 하며, 본 연구에서는 비디오 단편을 하나의 시계열 데이터로 해석하고 시간적 관계에 의한 검색을 위하여 새로운 비디오 검색 모델을 제안한다.

제안된 검색 모델은 여과(filter)과정과 정제(refinement)과정으로 구성된다. 비디오 데이터는 여과 과정을 통하여 정확한 검색 결과의 슈퍼셋을 구하고 다시 정제 과정을 거치면서 잘못 선택된 검색 결과를 버린다. 여과-정제 과정을 거치지 않고 한 단계로 검색을 하는 검색 모델을 *NVRTEC(Naive-VRTEC)* 모델이라고 하여 *VRTEC*과 *NVRTEC*을 비교함으로써 제안한 다단계 검색 모델의 효율성을 검증한다.

1. 비디오 검색 과정

*DFT(이산 퓨리에 변환:Discrete Fourier Transform)*를 이용하여 시간 영역의 비디오 데이터를 주파수 영역으로 변환한 후에 저주파수의 퓨리에 계수를 지배 계수로 선택함으로써 차원 축소가 이루어진다. $\vec{X}$가 $\vec{x}$의 이산 퓨리에 변환식이라고 한다면 $E(\vec{X})=E(\vec{x})$성질을 만족한다(정리 1. Parseval의 정리[19]). $E(\vec{X})$는 벡터 $\vec{X}$의 에너지이며 2장 2절에서 설명하였다. Parseval의 정리와 이산퓨리에 변환의

선형성 성질(모든 실수 상수 a, b에 대하여, $a\vec{x}+b\vec{y}$의 이산퓨리에 변환식은 $a\vec{X}+b\vec{Y}$이다. 이때 $\vec{X}$, $\vec{Y}$는 각각 $\vec{x}$, $\vec{y}$의 이산 퓨리에 변환식이다), 그리고 따름정리 1은 k차원 공간에서 구한 검색 결과가 정확한 검색 결과의 슈퍼셋임을 의미한다.

따름정리 1. (거리 보존 법칙). $\vec{X}$와 $\vec{Y}$가 각각 $\vec{x}$, $\vec{y}$의 이산퓨리에 변환이라고 하면 다음 성질을 만족한다.

$$D(\vec{X},\vec{Y})=D(\vec{x},vcey)$$

증명) 이산퓨리에 변환은 선형성이 있는 변환이므로, Parseval 정리에 의하여 모든 실수 상수 a, b에 대하여, $E(a\vec{x}+b\vec{y})=E(a\vec{X}+b\vec{Y})$을 만족한다. $a=1$, $b=-1$일 때 $E(\vec{x}-\vec{y})=E(\vec{X}-\vec{Y})$이 성립한다. 따라서

$$D(\vec{x},\vec{y})=(E(\vec{x}-\vec{y}))^{1/2}=(E(\vec{X}-\vec{Y}))^{1/2}=D(\vec{X},\vec{Y})$$

시간 영역과 주파수 영역에서의 거리가 보존되므로 두 객체간의 유사성 검사를 시간 영역에서 하는 대신에 주파수 영역에서 할 수 있다. 정의 3은 시간 영역에서의 임의의 두 비디오 단편의 유사성을 검사하기 위한 거리 함수이며, M개의 w차원 점의 거리 합을 궤적 비교를 위한 거리 함수로 정의한다. 비디오 단편은 3장에서 제안한 비디오 모델 WIVE에 의하여 연속적인 M개의 w차원 점으로 표현된다.

정의 3. (D_t: 시간 영역에서의 거리 함수).

q, a는 시간 영역에서의 궤적이며, 다음처럼 표시한다.

그림 3.4 비디오 검색의 2가지 종류

Fig. 3.4 2 types of video query.

$$q=\left[q_t^i\right],\ a=\left[a_t^i\right]$$

where q_t^i: 질의비디오의 i번째 비디오-윈도우의 t번째 프레임,

a_t^i: 대상비디오의 i번째 비디오-윈도우의 t번째 프레임,

$i,t\in$ 정수, $0<i\le M-1$, $0<t\le w-1$,

M: 궤적길이(= $Len(Query\ Video)-w+1$),

w: 비디오-윈도우 크기

따라서, 시간 영역에서의 궤적 비교를 위한 거리 함수를 다음처럼 정의한다.

$$D_t(q,a)=\sum_{i=0}^{M-1}\sum_{t=0}^{w-1}\left(q_t^i-a_t^i\right)^2$$

정의 4. (D_f : 주파수 영역에서의 거리 함수).

Q, A는 주파수 영역에서의 궤적이며, 다음처럼 표시한다.

$$Q=\left[Q_f^i\right],\ A=\left[A_f^i\right]$$

where, Q_f^i: 질의비디오의 i번째 비디오-윈도우의 이산퓨리에 변환식의 f번째 지배계수,

A_f^i: 대상비디오의 i번째 비디오-윈도우의 이산퓨리에 변환식의 f번째 지배계수,

$i,f\in$ 정수, $0<i\le M-1$, $0<f\le k-1$,

Q^i: $DFT\ of\ q^i$, $A^i=DFT\ of\ a^i$,

M: 궤적 길이 (= $Len(Query\ Video)-w+1$)

따라서, 주파수 영역에서의 궤적 비교를 위한 거리 함수를 다음처럼 정의한다.

$$D_f(Q,A)=\sum_{i=0}^{M-1}\sum_{f=0}^{k-1}\left(Q_f^i-A_f^i\right)^2$$

Q, A는 주파수 영역에서의 궤적이며 각각 시간 영역에서의 궤적인 q, a를 구성하는 각 비디오-윈도우의 이산퓨리에 변환의 집합으로 구성되어 진다(정의 4 참조). 정의 4는 k차원에서의 궤적 비교를 위한 거리 함수를 정의하며, 정리 2와 따름 정리 2는 퓨리에 변환을 이용하여 차원 축소를 한 후에 구해지는 검색 결과가 정확한 검색 결과의 슈퍼세이 되기 때문에 비디오 검색을 위한 여과 과정이 된다는 것을 의미한다.

정리 2. (하한값 정리). 시간 영역에서의 임의의 두 궤적 q, a와 각각의 이산퓨리에 변환식 Q, A는 $D_f(Q,A)\le D_t(q,a)$을 만족한다.

증명)

$$\begin{aligned}D_f(Q,A)&=\sum_{i=0}^{M-1}\sum_{f=0}^{k-1}\left(Q_f^i-A_f^i\right)^2\\&\le\sum_{i=0}^{M-1}\sum_{f=0}^{w-1}\left(Q_f^i-A_f^i\right)^2=\sum_{i=0}^{M-1}\sum_{t=0}^{w-1}\left(q_t^i-a_t^i\right)^2\\&=D_t(q,a)\end{aligned}$$

왜냐하면, 다음 부등식이 성립하기 때문이다.

$$\sum_{i=0}^{M-1}\sum_{f=0}^{w-1}\left(Q_f^i - A_f^i\right) = \sum_{i=0}^{M-1}\left(\sum_{f=0}^{k-1}\left(Q_f^i - A_f^i\right) + \sum_{f=k}^{w-1}\left(Q_f^i - A_f^i\right)\right)$$

정리 2에 근거하여, 따름정리 2는 주파수 영역의 지배 계수를 이용하여 구한 검색 결과가 정확한 검색 결과의 슈퍼셋임을 보여 준다.

따름정리 2. (No false-dismissals). 시간 영역에서의 임의의 두 궤적 q, a와 각각의 이산 퓨리에 변환식 Q, A는 정리 2에 의하여 다음과 같은 성질이 있다.

만약 $D_t(q, a) \le \varepsilon$을 만족하면 $D_f(Q, A) \le \varepsilon$을 만족한다.

2. 검색 알고리즘

제안된 비디오 모델 *WIVE*에서 예상되는 검색 형태를 살펴보면 그림 3.4처럼 2 종류가 있다. 모든 검색은 외쪽-정렬되어 있으며 질의 비디오의 길이가 비디오-윈도우와 같은 길이이거나 긴 경우 두 2가지 종류의 검색 형태가 있다. 비디오 검색 모델 *VRTEC*은 여과 과정과 정제 과정 등 다단계 검색을 하며, 각 검색 형태에 따른 검색 알고리즘은 다음과 같다.

질의 비디오의 길이가 비디오-윈도우 길이 보다 같은 경우에는 다차원 공간상의 두 점 사이의 거리를 비교한다. 각 알고리즘은 영역질의를 위한 알고리즘이다.

경우 1 : VRTEC :질의 비디오의 길이가 비디오-윈도우 길이와 같은 경우

<u>알고리즘</u>

1. [초기화] $R \leftarrow \phi$
2. [여과]
   $X \leftarrow$ *the last video window of a given vid eo*
   $k \leftarrow$ *the number of first few coeff icient of DFT*
   $Q_f \leftarrow$ *the* f^{th} *fourier coefficien t of query video Q*
   $A_f^s \leftarrow$ *the* f^{th} *fourier coefficien t of video – window s*
   /* S : *video – window* */
   for $S = 0$ *to* X *do*
   if $\sum_{f=0}^{k-1}\left(Q_f - A_f^S\right) \le \varepsilon, R \leftarrow R \cup \{S\}$
3. [정제]
   $Y \leftarrow$ *the number of elements in R*
   $R_i \leftarrow i^{th}$ *video – window of R*
   $q_t \leftarrow$ *the* t^{th} *frame of query video q*
   $a_t^S \leftarrow$ *the* t^{th} *frame of video – window S*
   for $S = R_1$ *to* R_Y *do*
   if $\sum_{t=0}^{w-1}\left(q_t - a_t^S\right) > \varepsilon$, Subtract $\{S\}$ *from R*
4. *Return (R)*

질의 비디오의 길이가 비디오-윈도우 길이와 같은 경우의 여과 과정은 정의 4의 D_f를 사용하여 $k(\ll w)$ 차원 공간에서 검색을 수행하여 검색 결과 후보를 결정하는 과정이며, 구해진 검색 결과 후보들에 대하여 w차원의 시간 영역에서 정의 3의 D_t를 사용하여 각각 다시 검색하여 잘못된 결과를 제거하는 것이 정제 과정이다.

질의 비디오의 길이가 비디오-윈도우 길이 보다 큰 경우에는 궤적 비교를 해야 한다. 인접한 $Len(Query\ Video) - w + 1$개의 점으로 만들어지는 대상 비디오의 궤적과 질의 비디오 궤적을 비교한다(그림 3.3). 질의 비디오의 길이가 비디오-윈도우 길이 보다 큰 경우의 여과를 위한 유사성 검사는 D_f를 사용하여 k차원 공간에서 검색을 수행하며, 그 수행 결과 검색 결과 후보를 결정한다. 여과 과정에서 구해진 검색 결과 후보들에 대하여 w차원의 시간 영역에서 D_t를 사용하여 정제 과정을 수행한다.

경우 2 : VRTEC:질의 비디오의 길이가 비디오-윈도우 길이 보다 큰 경우

<u>알고리즘</u>

1.[초기화] $R \leftarrow \phi$
2.[여과]
   $X \leftarrow$ *the last window of a given video*
   $k \leftarrow$ *the number of first few coefficient of DFT*
   $w \leftarrow$ *the length of video – window*
   $L \leftarrow$ *the length of query input*
   $m \leftarrow$ *the number of video – window in query input*
   /* S : *video – window* */
   for $S = 0$ *to* X *do*
   if $\sum_{i=S}^{S+m-1}\sum_{f=0}^{k-1}\left(Q_f^{i-S} - A_f^i\right) \le \varepsilon, R \leftarrow R \cup \{S\}$
3. [정제]
   $Y \leftarrow$ *the number of elements in R*
   for $S = R_1$ *to* R_Y *do*
   if $\sum_{i=S}^{S+m-1}\sum_{t=0}^{w-1}\left(q_t^{i-S} - a_t^i\right) > \varepsilon$, *Subtract* $\{S\}$ *from R*
4.*Return(R)*

NVRTEC 모델은 시간 영역에서의 거리를 이용하는 방법으로서 정의 3의 D_t를 사용하여 검색한다. 질의 비디오의 길이가 비디오-윈도우의 길이와 같거나

긴 경우로 나누어서 경우 1과 경우 2의 정제 과정과 유사하게 검색한다.

V. 실 험

비디오 검색의 성능을 평가하기 위하여 검색의 정확도와 처리 과정의 속도를 검증한다. 본 논문의 *VRTEC* 검색 모델은 여과 과정과 정제 과정으로 구성되는 다단계 모델이며 여과 과정이 없는 검색 모델 *NVRTEC* 모델과 비교한다. 비교할 내용은 검색 알고리즘의 정확성과 처리 속도에 대한 평가이다. 그러나 현재 비디오 검색에 대한 표준과 비디오 검색 방법의 평가에 대한 표준이 없다[20]. 즉, 비디오 검사 집합(test set)에 대한 표준과 표준 벤치마크 등이 없으며 그 이유는 비디오 검색 시스템에 대한 연구와 개발은 새로운 연구 과제로 진행중인 분야이고, 비디오 데이터의 다양성과 사용자의 다양한 요구 사항 등 때문이다. 따라서 정보 검색(*IR : Information Retrieval*) 분야에서 사용하는 *Precision/Recall metrics*를 본 논문에서 제안한 데이터 모델 WIVE와 검색 모델 *VRTEC* 에 대한 검색의 정확도와 처리 속도를 알아보기 위한 실험에 사용한다.

1. 실험 준비

한 개의 비디오를 위하여 257개의 실수 데이터를 100번 생성한다. 이는 약 10초 정도의 샷을 가정한 것이며 데이터베이스에 있는 비디오는 약 17분(25700개 프레임이므로 1028초, 즉 17.1분) 정도의 비디오가 된다. 본 실험에서는 비디오의 전체 프레임 개수를 25700개, 비디오-윈도우의 크기를 200개의 프레임으로 설정하였으므로 주어진 비디오를 위한 *WIVE* 모델은 25501개의 200차원 점으로 표현된다. 검색의 성능을 위하여 각각의 비디오-윈도우에 대한 *DFT* 변환식의 지배 계수를 구함으로써 차원을 줄이며, 만약 선택하는 지배계수의 개수가 3개이면 200차원의 점이 3차원 점으로 바뀌어 검색 공간을 줄이게 된다.

2. VRTEC 검색 모델의 정확성과 처리 속도를 위한 실험

VRTEC 검색 모델의 정확성을 평가하기 위하여 *NVRTEC* 모델의 검색 결과들과 비교하였다. 데이터 파싱 과정에서 정확한 특징 추출을 하였다면 *NVRTEC* 모델의 검색 결과가 정확한 검색 결과이므로 이를 이용하여 *Precision/Recall* 값을 구한다.

2.1 실험 과정

질의 비디오는 200 프레임, 300 프레임의 2종류를 생성한다. 일양분포를 만족하는 데이터를 생성하며, 같은 실험을 100회 시행하여 그 평균값을 한 번의 실험 결과로 한다. 이러한 실험을 10번 수행하여 각 결과를 비교하고 평균값을 구한다.

질의 비디오와 대상 비디오 사이의 유사성은 정규화된 유클리디안 공간에서의 거리를 사용한다.

정의 5. ($d_t(q,a)$, $d_f(Q,A)$: 유사성 비교를 위한 정규화 거리 함수)

$$d_t(q,a)=\frac{1}{1+D_t(q,a)} \quad : \text{시간 영역}$$

$$d_f(Q,A)=\frac{1}{1+D_f(Q,A)} \quad : \text{주파수 영역}$$

따라서, d_t: $D^{w-1}\rightarrow(0,1)$, d_f: $D^{k-1}\rightarrow(0,1)$ 이 되며, 0 값은 질의 비디오와 대상 비디오가 정확하게 일치함을 의미하고, 1은 전혀 일치하지 않음을 뜻한다.

정의 6. ($S_t(q,a)$, $S_f(Q,A)$: 유사성 비교를 위한 유사성 함수)

$$S_t(q,a)=1-d_t(q,a),$$
$$S_f(Q,A)=1-d_f(Q,A),$$

실험에 사용된 유사성 함수의 함수값이 1 일 때는 Exact Matching을 의미하며, 함수값 0는 전혀 유사하지 않음을 뜻한다.

2.2 실험 결과 및 분석

실험의 분석에서 사용된 값들은 각각 다음과 같이 구해진다

$$Precision=\frac{|QuRe\_Nvrtec\cap QuRe\_Vrtec\_Ref|}{|QuRe\_Vrtec\_Ref|}$$

$$Recall=\frac{|QuRe\_Nvrtec\cap QuRe\_Vrtec\_Ref|}{|QuRe\_Nvrtec|}$$

$$Query\ Speed=(Num\_Frames\times Num\_Video\_Window)+(D\_FT\_Coef\times Num\_Video\_Window+|QuRe\_Vrtec\_Fil|\times Num\_Frames)$$

$$Filtering\ Ratio=\frac{|QuRe\_Vrtec\_Fil|}{Num\_Video\_Window}$$

where,

QuRe_Nvrtec : *NVRTEC* 검색 모델의 검색 결과 집합

QuRe_Vrtec_Fil : *VRTEC* 검색 모델의 여과 과정 결과 집합

QuRe_Vrtec_Ref : *VRTEC* 검색 모델의 정제 과정 결과 집합

Num_Frames : 질의 비디오의 길이

Num_Video_Window : 전체 비디오-윈도우의 개수

D_FT_Coef : 선택된 지배 계수의 개수

아래 테이블들은 실험 결과들이다. 질의 비디오의 길이가 비디오-윈도우와 같은 경우와 큰 경우에 대한 실험 결과이다. 단위는 모두 %이며, 각각은 여과 정도를 다르게 했을 때의검색의 정확도와 검색 속도이다.

경우 1:질의 입력의 길이가 비디오-윈도우와 같은 경우

	실험 1	실험 2	실험 3	실험 4	실험 5	실험 6	실험 7	실험 8	실험 9	실험 10
Precision	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999
Recall	99.845	99.999	99.973	99.982	99.898	99.977	99.992	99.866	99.865	99.989
검색속도	673.97	603.33	681.54	634.25	637.74	609.54	709.80	620.06	625.23	596.70
여과정도	13.672	15.411	13.509	14.600	14.514	15.239	12.924	14.963	14.828	15.592

평균 : Precision = 99.999, Recall = 99.939,
검색속도 = 639.22, 여과정도 = 14.525

	실험 1	실험 2	실험 3	실험 4	실험 5	실험 6	실험 7	실험 8	실험 9	실험 10
Precision	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999
Recall	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999
검색속도	501.33	454.67	511.69	471.31	476.77	457.77	516.04	462.00	471.57	443.60
여과정도	18.784	20.827	18.376	20.051	19.809	20.678	18.213	20.481	20.039	21.379

평균 : Precision = 99.999, Recall = 99.999,
검색속도 = 476.68, 여과정도 = 19.864

	실험 1	실험 2	실험 3	실험 4	실험 5	실험 6	실험 7	실험 8	실험 9	실험 10
Precision	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999
Recall	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999
검색속도	361.47	334.02	372.24	342.72	348.48	336.79	368.85	335.95	345.74	325.24
여과정도	26.501	28.772	25.698	28.014	27.531	28.527	25.945	28.600	27.759	29.580

평균 : Precision = 99.999, Recall = 99.999,
검색속도 = 347.15, 여과정도 = 27.693

경우 2: 질의 입력의 길이가 비디오-윈도우보다 큰 경우

	실험 1	실험 2	실험 3	실험 4	실험 5	실험 6	실험 7	실험 8	실험 9	실험 10
Precision	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999
Recall	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999
검색속도	662.25	573.36	653.42	621.42	622.18	596.56	625.84	595.72	596.28	588.71
여과정도	13.936	16.276	14.138	14.926	14.906	15.596	14.812	15.620	15.605	15.821

평균 : Precision = 99.999, Recall = 99.999,
검색속도 = 613.57, 여과정도 = 15.164

	실험 1	실험 2	실험 3	실험 4	실험 5	실험 6	실험 7	실험 8	실험 9	실험 10
Precision	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999
Recall	99.883	99.838	99.878	99.864	99.866	99.888	99.871	99.832	99.852	99.859
검색속도	470.53	419.58	477.67	451.21	452.41	432.46	455.32	431.80	434.08	426.32
여과정도	20.088	22.667	19.768	20.996	20.940	21.956	20.796	21.993	21.872	22.290

평균 : Precision = 99.999, Recall = 99.999,
검색속도 = 445.14, 여과정도 = 21.337

	실험 1	실험 2	실험 3	실험 4	실험 5	실험 6	실험 7	실험 8	실험 9	실험 10
Precision	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999
Recall	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999	99.999
검색속도	340.34	299.11	342.03	318.89	316.99	307.14	325.95	304.10	303.99	301.16
여과정도	29.107	32.269	28.070	30.195	30.379	31.395	29.513	31.717	31.730	32.039

평균 : Precision = 99.999, Recall = 99.999, 검색속도 = 314.97, 여과정도 = 30.641

예상되는 모든 경우에서 *VRTEC* 검색 모델은 99.999% 의 검색 정확도를 가지며, 이는 제안한 검색 모델이 정확한 검색을 함을 의미한다. *VRTEC* 검색 속도는 *NVRTEC* 검색 모델보다 평균적으로 469% 빠르고, 여과 정도가 작을수록 검색 속도가 증가한다는 것을 알 수 있다.

다차원 공간에서의 궤적 비교를 함에 있어서 여과/정제 과정이 없는 경우와 있는 경우사이의 반복 횟수를 비교하였으며, 속도 향상의 이유는 퓨리에 변환을 이용하여 실제 검색 대상을 축소하였기 때문이다(여과 과정). 축소하여 구한 결과에 정확한 검색 결과가 반드시 포함되어 있음은 4장에서 수학적으로 증명하였으며, 이러한 이유로 인하여 검색의 정확성은 유지하면서 검색 속도가 향상된다. 4장 2절 알고리즘의 영역값 (Range) ε가 여과 정도를 결정한다.

VI. 결론 및 향후 연구

비디오 데이터에 대한 내용을 기반으로 한 비디오 단편의 유사성을 검사할 수 있는 데이터 모델(*WIVE*)과 검색 모델(*VRTEC*)을 제안하고 그 방법의 적법성과 효율성을 검증하였다. 비디오 단편의 유사성을 검사함은 비디오가 시공간 정보를 가지는 데이터이고 사용자의 검색 요구가 샷과 씬, 스토리 등이기 때문이다. 실제 질의 비디오의 길이는 임의적이기 때문에 비디오 단편을 검색하기 위한 거리 함수로 다차원 공간에서의 궤적을 비교하는 함수를 제안하였다. 새로 제안한 검색 모델 *VRTEC*은 정확한 검색 결과를 구하면서 검

색 속도는 *NVRTEC* 모델보다 평균적으로 469% (4.69배) 빠르다.

본 논문에서는 예제 비디오 단편을 통한 QBE (Query By Example) 검색 방법을 이용하여 비디오 검색을 실험하였다. 사용자에게 다양한 방법의 검색 방법을 제공하기 위한 검색 인터페이스에 대한 설계 및 구현에 대한 연구가 필요하다. 또한 예제 비디오 단편이나 사용자의 그림 등과 같은 시각적 특징 (Visual feature)을 이용한 검색뿐만 아니라 비시각적 특징(Non-visual feature)을 이용한 검색이 필요하고 두 가지 방법을 통합하는 복합 검색(Complex Query)에 대한 연구도 필요하다. 또한 주어진 비디오 데이터의 시각적/비시각적 정보를 구하기 위한 파싱에 대한 연구와 비디오 인덱스에 대한 연구가 필요하다.

참 고 문 헌

[1] M. Flickner et. al., "Query by Image and Video Content : The QBIC System," IEEE Computer, pp.23-32, September 1995.

[2] Sibel Adali, K. Selcuk Candan, Su-Shing Chen, Kutluhan Erol, V. S. Subrahmanian, "The Advanced Video Information System : data structure and query processing," ACM Multimedia Systems, Vol.4, pp.172-186, 1996.

[3] B. Prabhakaran, *Multimedia Databases Management Systems*, pp.141-143, Kluwer Academic Publishers, 1997

[4] H. J. Zhang and Q. Tian, "Digital Video Analysis and Recognition for Content-Based Access," ACM Computing Surveys, Vol.27, No.4, Dec. 1995.

[5] V. E. Ogle and M. Stonebraker, "Chabot : Retrieval from a Relational Database of Images," IEEE Computer, pp.40-48, September 1995.

[6] S. W. Smoliar and H. J. Zhang, "Content-Based Video Indexing and Retrieval," IEEE Multimedia, pp.62-72, Summer 1994.

[7] Yihong Gong, Hong J. Zhang, H. C. Chuan, M. Sakauchi, "An Image Database System with Content Capturing and Fast Image Indexing Abilities," ICMCS, May, 1994.

[8] H. J. Zhang and S. W. Smoliar, "Developing Power Tools for Video Indexing and Retrieval," SPIE Vol.2185, pp.140-149, 1994.

[9] Yihong Gong, L. T. Sin, H. C. Chuan, Hong J. Zhang, M. Sakauchi, "Automatic Parsing of TV Soccer Programs," ICMCS, May 1995.

[10] Hong J. Zhang, Yihong Gong, S. W. Smoliar, Shuang Yeo Tan, "Automatic Parsing of News Video," ICMCS, May 1994.

[11] Christos Faloutsos et. al., "Fast Subsequence Matching in Time-Series Databases," ACM SIGMOD, pp.419-429, 1994.

[12] Byoung-Kee Yi, et. al., "Efficient Retrieval of Similar Time Sequences Under Time Wraping," Int'l Conf. on Data Engineering, pp.201-208, 1998.

[13] Christos Faloutsos et. al., "*FastMap*: A Fast Algorithms for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets," ACM SIGMOD, pp.163-174, 1995.

[14] D. Rafiei, A. Mendelzon, "Similarity-Based Queries for Time Series Data," ACM SIGMOD, pp.13-25, 1997.

[15] R. Agrawal, C. Faloutsos, and A. Swami, "Efficient Similarity Search In Sequence Databases," FODO Conference, pp.69-84, Oct. 1993.

[16] R. Agrawal, K. I. Lin, H. S. Sawhney, and K. Shim, "Fast Similarity Search in the Presence of Noise, Scaling, and Translation in Time-Series Databases," VLDB, pp.490-501, 1995.

[17] A. P. Sistla, C. Yu, C. Liu, and K. Liu, "Similarity based Retrieval of Pictures Using Indices on Spatial Realtionships," VLDB, pp.619-629, 1995.

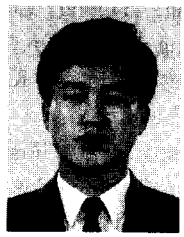
[18] A. P. Sistla, C. Yu, and R. V. Subrahmanian, "Similarity Based Retrieval of Videos," Int'l Conf. of Data Engineering, pp.181-190, 1997.

[19] Richard J. Higgins, Digital Signal Processing In VLSI, pp. 61-62, Prentice-Hall,

1990.

[20] Shih-Fu Chang, William Chen, Horace J. Meng, Hari Sundaram, Di Zhong, "VideoQ: An Automated Content Based Video Search System Using Visual Cues," ACM Multimedia, pp.313-324, 1997.

저 자 소 개



金 昌 龍(正會員)

1990年 2月 서울대학교 계산통계학과 졸업(이학사). 1992年 2月 서울대학교 계산통계학과 졸업(이학석사). 1993年 9月~현재 한국과학기술원 전산학과 박사과정 수료. 1992年 6月~1993年 2月 삼성종합기술원 정보시스템 연구소 연구원. 1993年 3月~1998年 2月 부천대학 전자계산과 조교수. 1998年 3月~현재 부천대학 정보통신계열 조교수. 주관심 분야는 멀티미디어 데이터베이스, 시스템 소프트웨어, 데이터 마이닝 등임