논문 2006-43SP-2-1

비-파라미터 기반의 움직임 분류를 통한 비디오 검색 기법

(Video retrieval method using non-parametric based motion classification)

김 낙 우*, 최 종 수**

(Kim Nac-Woo and Choi Jong-Soo)

요 약

본 논문에서는 샷(shot) 기반 비디오 색인 구조에서 비-파라미터(non-parametric) 기반의 움직임 분류를 통한 비디오 영상 검색 기법을 제안한다. 본 논문에서 제안하는 비디오 검색 시스템은 장면 전환 기법을 통해 얻은 샷 단위의 짧은 비디오로부터 대표 프레임과 움직임 정보를 취득한 후, 이를 통해 시각적 특징과 움직임 특징을 추출하여 유사도를 비교함으로써 시-공간적 특징을 이용한 실시간 검색이 가능하도록 구현되었다. 비-파라미터 기반의 움직임 특징의 추출은 MPEG 압축 스트림으로부터 정규화된 움직임 벡터계(界)를 추출한 후, 각각의 정규화된 움직임 벡터를 여러 개의 각도 빈(bin)으로 양자화하고 이의 평균과 분산, 방향 등을 고려함으로써 효과적으로 이루어진다. 대표 프레임에서의 시각 특징 검출을 위해서는 에지 기반의 공간 기술자를 이용하였다. 실험 결과는 영상 색인 및 검색에 있어서 제안된 시스템이 매우 효과적임을 잘 나타내고 있다. 데이터베이스 내 영상의 색인을 위해서는 R*-tree 구조를 이용한다.

Abstract

In this paper, we propose the novel video retrieval algorithm using non-parametric based motion classification in the shot-based video indexing structure. The proposed system firstly gets the key frame and motion information from each shot segmented by scene change detection method, and then extracts visual features and non-parametric based motion information from them. Finally, we construct real-time retrieval system supporting similarity comparison of these spatio-temporal features. After the normalized motion vector fields is created from MPEG compressed stream, the extraction of non-parametric based motion feature is effectively achieved by discretizing each normalized motion vectors into various angle bins, and considering a mean, a variance, and a direction of these bins. We use the edge-based spatial descriptor to extract the visual feature in key frames. Experimental evidence shows that our algorithm outperforms other video retrieval methods for image indexing and retrieval. To index the feature vectors, we use R*-tree structures.

Keywords : video retrieval, non-parametric motion model, motion classification, key frame

I. 서 론

최근 멀티미디어 정보에 대한 관심은 인터넷 사용자의 대중화와 네트워크 기술의 발달과 더불어 점점 더 크게 증대되고 있다. 그러나 원격지에 저장된 방대한 자료에서 원하는 영상을 검색하여 취득하는 데에는 여전히 상당한 어려움이 따른다. 이에, 사용자가 보다 효과적이고 유연하게 멀티미디어 자료 검색 서비스를 이용할 수 있도록 하기 위한 비디오 검색 시스템이 점차로 요구되고 있다[1-4].

비디오 검색 시스템 구축을 위한 핵심 기술 요소는 크게 효과적인 비디오 데이터 색인 위한 비디오 파싱(parsing) 기법과 사용자가 원하는 데이터를 쉽게 검색할 수 있도록 해주는 비디오 검색(retrieval) 기법 등을 들 수 있다. 비디오 파싱은 대용량의 비디오를 보다 적은 용량의 요약형 비디오로 만들기 위한 비디오 분할 과정을 말하는 것으로 샷의 경계, 즉 장면 전환이 이루어지는 지점을 검출하여, 분할된 영역에서의 특징 정보를 효과적으로 추출할 수 있도록 하는 비디오 분석 기

* 정회원, 한국전자통신연구원 광대역통합망연구단
(BcN Research Div., ETRI)
** 정회원, 중앙대학교 첨단영상대학원 영상공학과
(Dept. of Image Engineering, GSAIM, Chung-Ang University)
접수일자: 2005년6월30일, 수정완료일: 2006년2월8일

술을 말한다. 샷이란 하나의 카메라로 촬영하여 필름이 끊이지 않고 연속적으로 연결된 비디오 내 연속된 한 구간을 뜻하며 장면 전환의 기본 단위로 사용되는 데, 이러한 샷 기반의 비디오 분석 방법은 샷에서의 대표 프레임과 움직임 정보를 검출하는 데에 매우 효과적이며 내용 기반 검색을 위한 메타 정보의 획득에 용이한 구조로서 많은 사람들에 의해 활발히 연구되고 있다[5,6]. 비디오 파싱을 통해 취득된 샷에서의 대표 프레임과 움직임 정보는 최종적으로 R*-tree[7]등의 기법을 이용하여 색인되며 이는 비디오 검색을 위해 활용된다.

오늘날의 이러한 비디오 검색 알고리즘은 Zhang 등이 1995년 제안한 비디오 파싱과 내용 기반 비디오 검색 및 브라우징 기법[8]에서 이미 체계화되었다. Zhang은 [8]에서 비디오 파싱을 통해 비디오를 샷 단위로 시분할하고, 각 샷마다의 대표 프레임을 추출 한 후에, 색상과 질감 등을 분석하여 각 샷에서의 시각적 특징을 취득함과 동시에 카메라 동작이나 시간에 따른 밝기나 색상의 변화와 같은 시간적 특징 또한 추출하였다. 이러한 두 특징을 조합하여 사용자에게 MPEG 기반 하에 동작 가능한 비디오 검색 시스템 환경을 제공한다. Video Browsing and Retrieval System(VIRE)[9]은 저수준의 비디오와 오디오 특징과 고수준의 비디오 특징을 추출하고 MPEG-7의 기술자를 이용하여 이를 저장하는 방법을 제안하였다. 검색은 수동검색과 자동검색을 모두 지원하며, 고수준의 특징을 이용한 의미 기반 비디오 검색 결과를 제공한다. IBM의 CueVideo 시스템[10]은 자동적으로 샷 기반의 오디오와 비디오의 특징을 추출하고, 비슷한 샷을 color correlogram[11]을 이용하여 클러스터링한 후 이를 통해 의미 단위의 씬(scene)을 구성하는 방법을 소개하기도 하였다. Bilinear 움직임 모델[12]과 반복적 배제 구조(iterative rejection scheme)[13]를 이용하는 특정 객체 추출을 위한 비디오 검색 방법[14]이나, VISMap[15] 시스템과 같이 전통적인 질의 방법으로부터 확장하여 사용자의 의도를 반영하는 관련-회귀법(relevance feedback)을 이용한 방법 또한 소개되고 있다.

본 논문에서는 샷 기반의 비디오 분석을 통해 영상에서의 대표 프레임과 움직임 정보를 얻은 후, 제안하는 움직임 분류 기법과 에지 기반의 공간 기술자에 취득된 정보를 적용함으로써 효과적인 내용 기반 비디오 검색 시스템을 구현하는 방법을 제안하고 있다. 먼저 장면 전환 검출 기법을 적용하여 샷 단위로 정확하게 비디오를 파싱하고, 분할된 샷의 내용을 나타내는 대표 프레임과 움직임 정보를 추출한다. 그리고 나서 대표 프레임으로부터 에지 기반의 시각 특징을 추출하고, 움직임 정보에 비-파라미터 기반의 움직임 분류 기법을 적용시킴으로써 샷 단위의 카메라 동작을 취득하고 이를 색인한다. 최종적으로, 이러한 특징 벡터의 유사도를 비교함으로써 효과적인 샷 단위의 비디오 검색 시스템을 구현한다.

이 논문의 구성은 다음과 같다. Ⅱ장과 Ⅲ장에서는 제안하는 장면 전환 검출 기법과 움직임 분류 기법에 대해 설명하고, Ⅳ장에서 특징 추출 및 유사도 비교 방법에 대해 논의한다. Ⅴ장과 Ⅵ장에서는 이를 통한 실험 결과와 결론을 도출하고 있다. 그림 1은 본 연구의 대략적인 블록도를 나타낸 것이다.

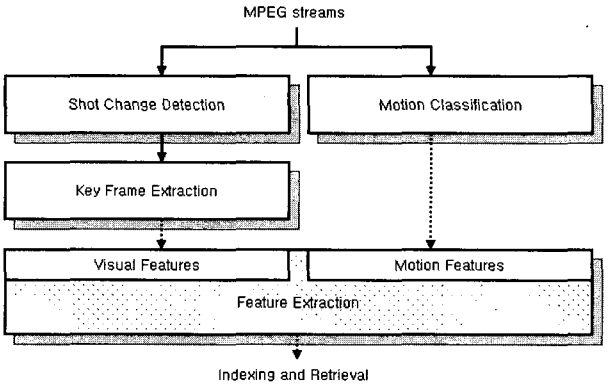


그림 1. 제안된 알고리즘의 블록도
Fig. 1. Block diagram of the proposed algorithm.

Ⅱ. 장면 전환 검출 기법

비디오에서 장면 전환의 유형은 크게 급격한 장면 전환(abrupt shot change)과 점진적인 장면 전환(gradual shot change)으로 나눌 수 있다. 급격한 장면 전환은 그림 2-(a)에서 보이듯 인접한 샷 간의 내용이 전혀 별개의 것으로 이어진 경우를 말하며, 점진적인 장면 전환은 그림 2-(b)와 같이 디졸브(dissolve), 페이드(fade) 등의 카메라 특수 효과에 의한 변화로서 인접 샷 간의 경계가 모호하게 이어진 경우를 나타낸다[16].

급격한 장면 전환 검출 기법은 이미 높은 성능을 보이는 많은 방법들이 제시되어 있지만, 점진적인 장면 전환 검출 기법은 효율성과 정확성의 측면에서 여전히 많은 문제점을 갖고 있다[16,17].

점진적인 장면 전환 구간을 찾는 대표적 방법은 화소간 분산, 프레임간 화소값 또는 히스토그램 차, 움직임

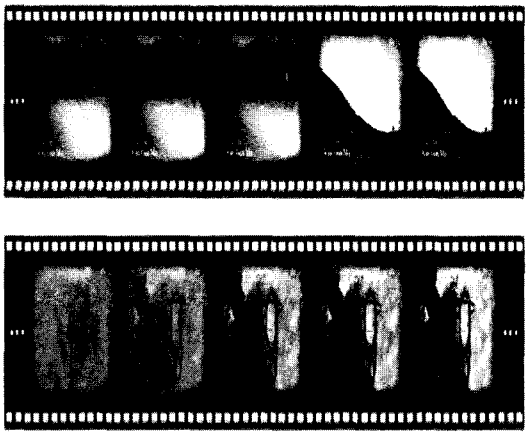


그림 2. 장면 전환의 예 (a) 급격한 장면 전환 (b) 점진적인 장면 전환
Fig. 2. Example of shot change (a) absolute shot change (b) gradual shot change.

벡터, 그리고 에지성분 등을 사용하는 것이다. Zabih[18]는 복원 영상의 에지 화소의 분포를 이용하여 디졸브, 페이드 및 와이프를 검출하는 방법을 시도하였지만, 에지를 찾기 위하여 영상을 모두 복원하는 과정에서 많은 계산량이 발생하는 단점이 있었으며, Meng[19]의 경우, MPEG 스트림의 DC 영상으로부터 분산값을 구하여 U 자형이 나타나는 구간으로 디졸브를 검출하는 방법을 제안하였으나, 일반적인 경우와는 달리 영상의 밝기 변화가 정적인 형태의 경우에만 검출되는 문제가 있었다. Yeo와 Liu의 알고리즘[16]은 움직임이 거의 없고 매우 이상적인 형태의 디졸브만이 검출 가능하였다. 또한, 뉴스, 영화, 뮤직 비디오, CF 등과 같은 영상물에서 비교적 빈번하게 나타나는 조명 변화(flashlight)의 경우, 기존의 검출 알고리즘은 넓은 구간에 걸쳐 적은 수의 조명 변화가 등장하는 경우나 한 두 프레임에 걸친 급격한 조명 변화는 비교적 잘 검출하였으나, 여러 프레임에 걸쳐 나타나는 점진적인 조명 변화에 대해서는 그 구간을 전혀 검출하지 못하는 오류를 보이고 있다. 이러한 이유로 Izquierdo[20]는 프레임 간의 움직임 벡터 분석을 통해 장면 전환 검출을 시도하기도 하였다.

본 논문에서는 MPEG 압축 영상을 대상으로 부분 복호화(partial decoding)에 의한 DCT DC 계수의 움직임 보상형 DC 이미지 재구성 방법[16]을 이용하여 처리 데이터량을 현저히 감소시키면서, 기존의 방법보다 더 효율적으로 비디오에서의 장면 전환을 검출하는 방법인 [21]을 본 논문에서의 샷 구분을 위하여 이용하고 있다.

Ⅲ. 움직임 분류 기법

비디오에서의 움직임 특징 취득은 일반적으로 움직임 벡터의 특징 패턴 분석을 통해 이루어지고, 이를 통해 해당 프레임에서의 카메라 동작을 예측할 수 있다. 카메라 동작에는 축이 고정된 카메라의 동작인 수평선회(panning), 수직선회(tilting), 확대/축소(zooming)와 움직이는 상태에서의 카메라 동작인 좌우이동(tracking), 상하이동(booming), 전후이동(dollying)이 있다. 비디오 상에서 이러한 카메라 동작을 분석하여 동일한 카메라 동작이 발생하는 장면을 샷 단위 구분으로 색인한다.

그림 3은 다양한 카메라 동작의 예를 보인 것이다. 축이 고정된 카메라에서의 수평선회 동작과 이동 카메라에서의 좌우이동 동작은 일반적으로 움직임 벡터 패턴이 유사하므로 이를 구별하기란 매우 어렵다. 따라서, 축 고정 카메라와 이동 카메라에서 서로 대응되는 카메라 동작을 서로 묶어 하나의 움직임 패턴으로 추출하게 된다. 본 논문에서는 유사 움직임 패턴을 갖는 카메라 동작을 하나로 묶어 표현하였으며, 이에 따른 다양한 카메라 동작에 대한 표기를 표 1에 나타내고 있다. 본 논문에서는 기호(symbol)를 통하여 여러 카메라 동작을 간략히 표현하고자 한다.

1. 파라미터 기반의 움직임 기술자

각 프레임에서의 카메라 동작 검출을 위하여 많은 연

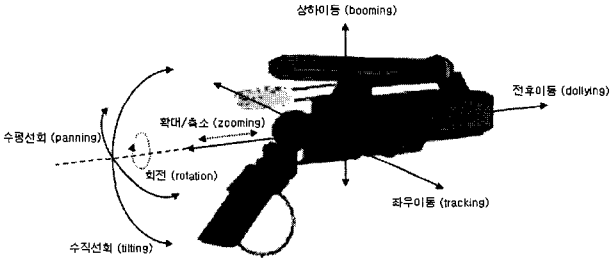


그림 3. 기본적인 카메라 동작
Fig. 3. Basic camera operations.

표 1. 다양한 카메라 동작과 기호 표시
Table 1. Various camera operations and symbols.

Camera motion		Symbol	
Stationary		S	
Pan	Pan_left	P	P_L
	Pan_right		P_R
Tilt	Tilt_up	T	T_U
	Tilt_down		T_D
Zoom	Zoom_in	Z	Z_I
	Zoom_out		Z_O
Rotation	Rotation	R	R

구에서 파라미터 모델 기반의 방법[22,23]을 이용하고 있다. 이 모델은 카메라 움직임을 평가하기 위해 P, T, Z, R 등 네 개의 파라미터를 고려한다. 수식 (1)은 파라미터 모델에서 각 프레임에서의 움직임 벡터와 카메라 동작과의 관계를 나타낼 때 사용하는 식이다.

$$\begin{pmatrix} u(x,y) \\ v(x,y) \end{pmatrix} = A\begin{bmatrix} x \\ y \end{bmatrix} + b = \begin{bmatrix} a_{zoom} & b_{rotate} \\ -b_{rotate} & a_{zoom} \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c_{pan} \\ d_{tilt} \end{bmatrix} \quad (1)$$

수식 (1)에서 $(u, v)^T$는 압축 프레임 중 $(x, y)^T$ 위치에서의 움직임 벡터를 의미한다. 파라미터 a_{zoom}, b_{rotate}, c_{pan}, d_{tilt}는 각각 Z, R, P, T를 나타낸다. 일반적으로 파라미터 기반의 방법에서 각 파라미터는 움직임 벡터계에 가우시안 필터를 적용시켜 프레임에서의 벡터 에러를 최소화 시킨 후 선형 최소 제곱법(linear least square method)을 이용하여 구할 수 있다. 각 파라미터 계수가 일정 임계치를 초과할 경우, 해당 프레임을 카메라 모션의 일부분으로 인식하고, 해당 프레임을 Z, R, P, T 중 하나의 프레임으로 분류한다. 그림 4는 파라미터 기반의 움직임 분류 성능에 대한 결과를 보여준다. 그림 4-(a)에서 4-(d)는 각각 Z, R, P, T로 기(旣) 분류된 비디오 영상을 수식 (1)에 적용하여 파라미터 계수를 추출한 결과이다. 결과에서 보이는 바와 같이 P와 T의 경우는 일정 임계치 이상의 값을 기준으로 하여 찾을 수 있으나, Z와 R의 경우는 찾기가 불가능한 것을 알 수 있다. 우리는 이러한 결과를 바탕으로 비-파라미터 기반의 움직임 기술자를 제안하고 이를 이용하였다.

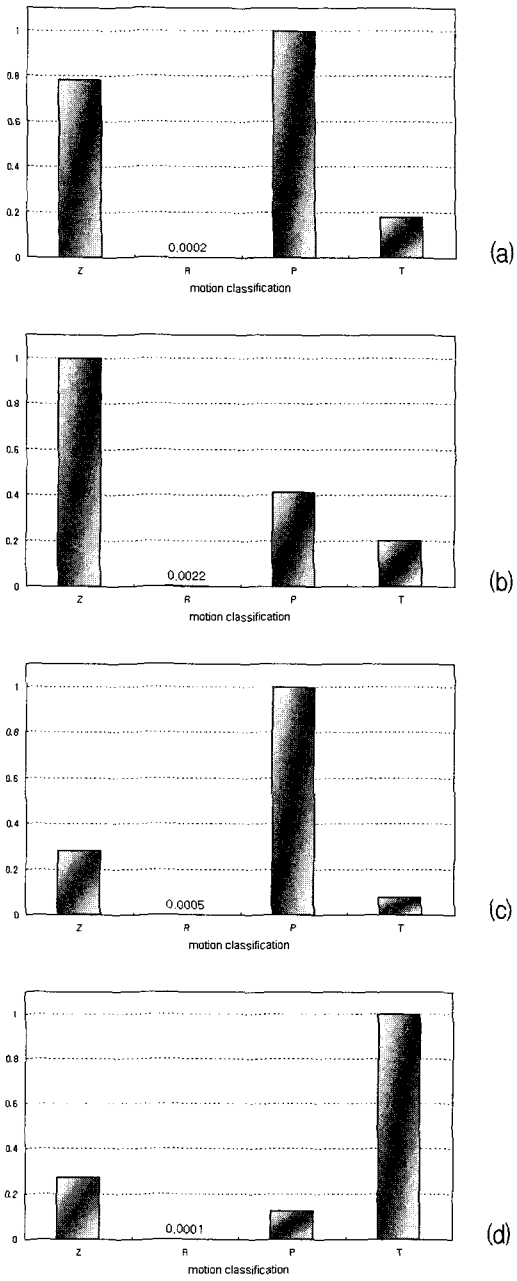


그림 4. 파라미터 기반의 움직임 분류
(a) Z (b) R (c) P (d) T

Fig. 4. Parameter-based motion classification.
(a) Z (b) R (c) P (d) T

2. 제안하는 카메라 움직임 기술자

파라미터 기반의 움직임 추정 기법은 그림 4의 실험 결과에서 나타난 바와 같이 프레임에 대한 움직임 추정 성능에 많은 문제를 보이고 있다. 본 논문에서는 이러한 문제를 풀기 위해 비-파라미터 기반의 움직임 추정 기법을 제안한다. 그림 5는 제안하는 비-파라미터 기반의 움직임 분류 기법에 대한 블록도를 나타내고 있다.

MPEG 기반 하의 매크로 블록에서의 움직임 벡터를 이용하여 프레임에서의 움직임을 예측하고자 할 때에는 효율적인 움직임 해석을 위하여 일반적인 매크로블록에서의 벡터 정보를 프레임의 유형과 예측 방향 등에 무관한 형태의 정규화된 움직임 벡터로 전환하는 작업이 먼저 선행되어야 한다. 우리는 [24]에서 제안한 움직임 벡터 재추정 기법을 이용하여 해당 프레임의 움직임 벡터를 먼저 정규화한다. 정규화된 움직임 벡터는 peer group filtering[25]을 통해 잡음이 제거되며, 해당 프레임의 유효 움직임 벡터[24]의 수($Thres\_T$)를 통해 정적 프레임인지 동적 프레임인지의 여부가 일차적으로 판명된다.

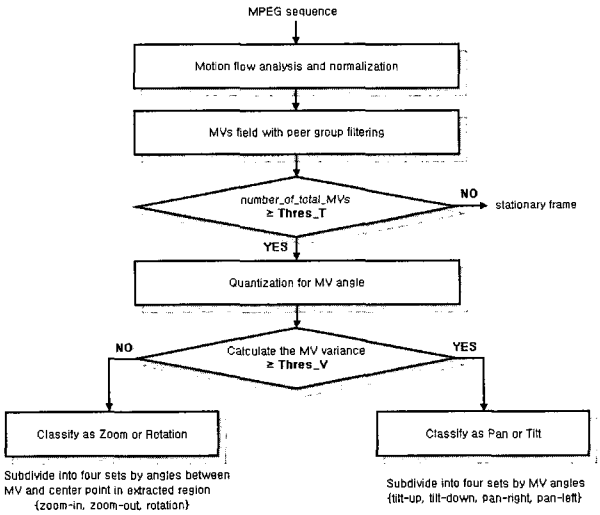


그림 5. 제안하는 움직임 분류 기법의 블록도
Fig. 5. Block diagram for proposed motion classification method.

다음 단계로서 각 프레임에서의 움직임 벡터를 여러 개(B_T)의 각도 빈으로 양자화한다. 수식 (2)로부터 i번째 프레임의 (u, v)에서의 움직임 벡터 각도를 구하고, 수식 (3)에서와 같이 최대 빈 값을 B_T로 하여 각도를 양자화한다. 여기에서, $\tilde{\alpha}$는 각도 α에 대한 양자화 값이다.

$$\alpha(u,v)^i = tan^{-1}\left(\frac{mv_x(u,v)^i}{mv_y(u,v)^i}\right) \tag{2}$$

$$1 \le \tilde{\alpha}(u,v)^i \le B_T \tag{3}$$

각도 빈에 대한 히스토그램을 $H^i(k), k \in \{1,2,\cdots,B_T\}$ 할 때, 우리는 히스토그램 H 로부터 i번째 프레임의 움직임 벡터 각도에 대한 평균(E^i)과 분산(σ^i)을 수식 (4)와 수식 (5)로부터 얻을 수 있다.

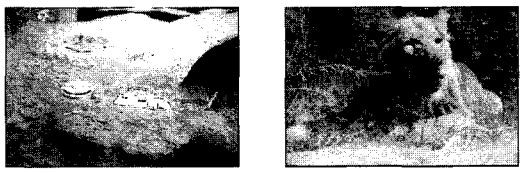
$$E^i = \frac{1}{B_T}\sum_{k=1}^{B_T} H^i(k) \tag{4}$$

$$\sigma^i = \left(\frac{1}{B_T}\sum_{k=1}^{B_T}\left(E^i - H^i(k)\right)^2\right)^{1/2} \tag{5}$$

여기에서, 수식 (5)에서 구한 σ^i을 기준으로 입력 프레임을 Z/R과 P/T로 양분한다. 그림 7은 그림 6의 두 비디오 영상에 대한 움직임 벡터 히스토그램을 나타낸 것이다. 그림 7-(a)는 그림 6-(a)의 Z영상에 대한 것이고, 그림 7-(b)는 그림 6-(b)의 P영상에 대한 움직임 벡터 히스토그램을 보이고 있다. Z나 R의 경우는 $H^i(k)$가 각 빈마다 일정한 값을 가지며, P나 T의 경우는 어느 한 빈의 값이 다른 빈에서의 값보다 월등히 큰 값을 가지게 된다.

가. Z와 R의 움직임 분류 기법

Z나 R 프레임으로 판명된 프레임은 Z_I, Z_O, R의 세 프레임 중 하나로 세부 분류된다. 카메라에 의해 촬영된 Z프레임은 대체로 영상의 중심 부근에 목적 객체를 갖는데, 본 알고리즘에서는 이러한 목적 객체의 위치와 외곽 움직임 벡터와의 상관관계를 고려해야 하므로 세부 프레임 분류에 앞서 영상에서의 객체 중심점을 먼저 추출한다. 객체 중심점 추출 알고리즘은 다음과 같다. 첫 단계로, Z 영상에서의 중심 객체를 찾기 위해 움직임 벡터 영상에 3 x 3윈도우 마스크를 씌우고, 마스크 영역 내의 움직임 벡터 집합 $\{\overline{MV_1}, \overline{MV_2},\cdots, \overline{MV_9}\}$ 중 최대값이 $Thres_D$를 넘지 않을 경우 우리



(a) (b)

그림 6. 다양한 비디오 영상 (a) Z (b) P
Fig. 6. Various videos (a) Z (b) P.

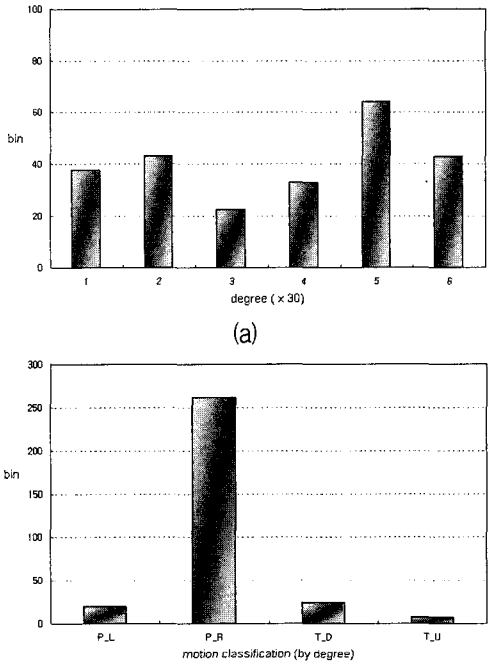
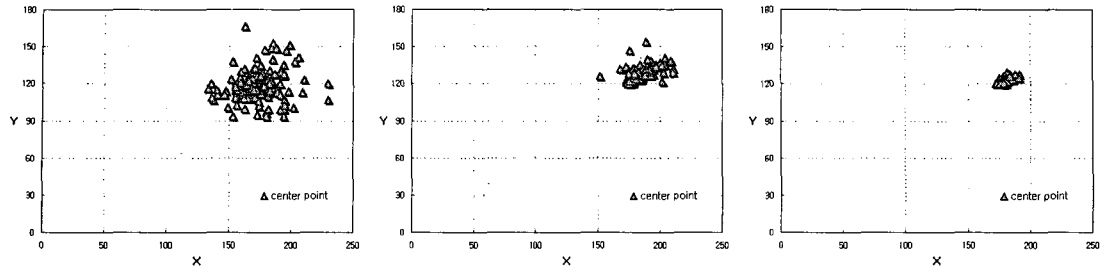


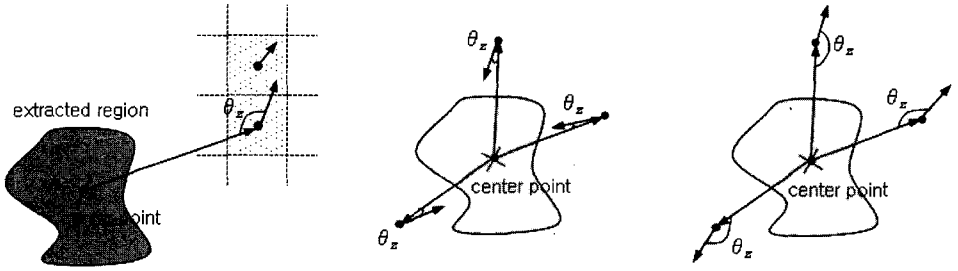
그림 7. 움직임 벡터 히스토그램 (a) Z (b) P
Fig. 7. Motion vector histogram (a) Z (b) P.



(a) $Thres\_D = 5$ (b) $Thres\_D = 3$ (c) $Thres\_D = 1$

그림 8. $Thres_D$의 변화에 따른 시퀀스에서의 객체 중심 좌표 추출

Fig. 8. Extraction of object center position by the variation of $Thres\_D$ in sequences.



(a) (b) (c)

그림 9. Z 프레임의 판별 (a) θ_z 계산 (b) Z_O에서의 θ_z (c) Z_I에서의 θ_z

Fig. 9. Discrimination of Z frame (a) the calculation of θ_z (b) θ_z in Z_O (c) θ_z in Z_I.

는 이 화소를 객체 중심 가능점($\hat{R}$)으로 지정한다. 수식 표현은 (6)과 같다.

$$\hat{R} = \begin{cases} 1 & if \ \underset{i=1\sim 9}{Max} \lfloor \overrightarrow{MV_i} \rfloor < Thres\_D \\ 0 & otherwise \end{cases} \quad (6)$$

이렇게 추출된 $\hat{R}$ 점들을 대상으로 하여 8-근방 레이블링을 적용하고 레이블링 된 영역 중 하나를 중심 영역으로 선택한다. 레이블 된 영역 중 중심 영역의 선택은 가장 큰 영역, 영상의 중심에서 가까운 영역, 신장율이 1에 가까운 영역을 우선 기준으로 한다. 그림 8은 그림 6-(a)의 비디오 영상에 $Thres_D$를 1, 3, 5로 각각 적용시켜 프레임 별로 Z의 중심 영역 좌표를 추출한 것을 보여주고 있다. 그림에서 보는 바와 같이 $Thres_D$를 1로 하였을 때, 정확한 시퀀스 영상의 중심 좌표가 추출되는 것을 볼 수 있다.

다음 단계로 추출된 객체의 중심 좌표를 이용하여 Z 영상을 세부 분류한다. 추출된 객체의 중심 좌표와 외곽 움직임 벡터들 간의 사이각(θ_z)을 계산함으로써 해당 프레임이 Z_I인지 Z_O인지의 여부를 판별할 수 있다(그림 9 참조). Z_I영상은 객체의 중심 좌표를 기준으로 하여 움직임 벡터가 바깥쪽으로 향하게 되고, Z_O영상은 반대로 움직임 벡터가 안쪽을 향하게 된다. 따라서, 사이각 θ_z 는 180도에 가깝거나 0도에 가깝게 된다. 본 논문에서는 각 프레임에서 다수의 움직임 벡터가 $|\theta_z| \le \frac{\pi}{6}$ 사이에 있을 경우는 Z_O프레임으로, $\frac{5}{6}\pi \le |\theta_z| \le \pi$ 사이에 있을 경우는 Z_I프레임으로 일차 분류하였다. 분류된 Z_I, Z_O프레임의 재확인을 위해 객체 중심 좌표에서 각 사분면에 속해있는 움직임 벡터의 이동방향을 그림 10에서와 같이 고려한다. 그림 10-(a)에서와 같이 영상의 기준 좌표는 좌측 상단이 (0, 0)이므로, 객체의 중심 좌표에서 각 움직임 벡터의 방향은 그림 10-(b), 그림 10-(c)에서 보듯이 Z_O와 Z_I프레임에 따라 달라야 하고 표 2에서와 같이 프레임 별로 사분면에 따라서 재해석되어야 한다.

R프레임의 경우는 객체의 중심이 아닌 영상의 중심 좌표와 외곽 움직임 벡터 간의 사이각(θ_r)을 계산하여

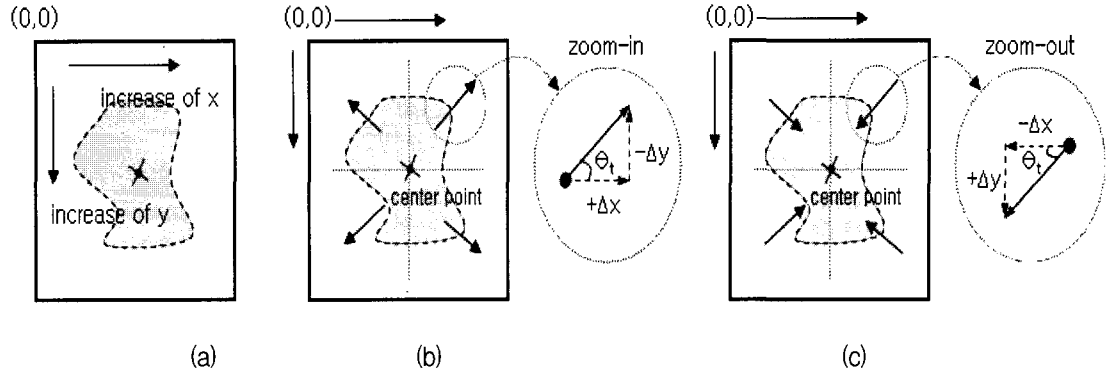


그림 10. 객체 중심 좌표와 움직임 벡터 방향을 이용한 Z프레임의 판별 (a) x, y의 증가 방향 (b) Z_I 프레임에서의 움직임 벡터 방향 (c) Z_O프레임에서의 움직임 벡터 방향

Fig. 10. Discrimination of Z frame using the object center position and the direction of motion vector (a) the increase of x, y position (b) the direction of motion vector in Z_I frame (c) the direction of motion vector in Z_O frame.

표 2. 각 사분면에서의 벡터 방향을 통한 Z프레임의 판별

Table 2. Discrimination of Z frame using the vector direction in each quadrant.

	Z_O		Z_I	
	x	y	x	y
1사분면	감소	증가	증가	감소
2사분면	증가	증가	감소	감소
3사분면	증가	감소	감소	증가
4사분면	감소	감소	증가	증가

판별한다(그림 11 참조). 이 경우 사이각 $|\theta_r|$ 은 $\frac{\pi}{2}$에 가깝게 된다. 본 논문에서는 영상의 중심 좌표와 각 움직임 벡터를 고려하여 $\frac{5}{12}\pi \le |\theta_r| \le \frac{7}{12}\pi$ 사이에 다수의 벡터가 있을 경우 R프레임으로 분류하였다.

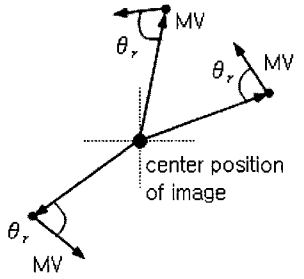


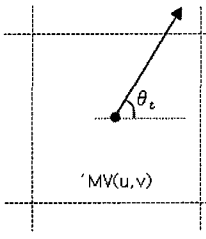
그림 11. θ_r을 이용한 R 프레임 판별

Fig. 11. Discrimination of R frame using θ_r.

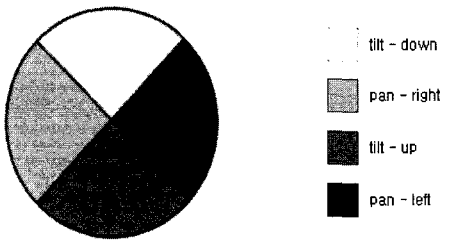
나. P와 T의 움직임 분류 기법

P와 T의 움직임 분류는 간단하게 이루어진다. 수식 (5)를 이용한 전처리를 통해 P/T 프레임으로 분류된 프레임에 대하여 B_T의 빈으로 양자화된 움직임 벡터를 각도 θ_t에 따라 다시 4방향으로 양자화하고, 수식 (7)를 만족하는 빈 n을 찾아 P_R, P_L, T_D, T_U을 예측한다(그림 12-(b) 참조).

$$H(n) > \left(\sum_{m=0}^{3} H(m)\right) - H(n) \qquad (7)$$



(a)



(b)

그림 12. P/T 프레임 (a) θ_t의 계산 (b) P/T프레임의 분류

Fig. 12. P/T frame (a) the calculation of θ_t (b) the classification of P/T frame.

이 때, n은 m에 속하는 빈 중 하나이다 ($n \in m$).

Ⅳ. 특징 추출 및 유사도 비교

비디오 검색 시스템을 이용하는 사용자는 검색을 위해 질의 영상을 선택하거나 질의 움직임을 선택해야 한다. 선택된 질의 영상에서의 대표 프레임을 통해 시각적 특징을 추출하고, 질의 움직임을 통해 검색할 영상의 움직임 특징을 얻는다.

1. 샷 구간의 시각 특징 기술자와 움직임 특징의 추출

우리는 해당 샷 구간 내에 가장 움직임이 적은 프레임을 선택하여 이를 대표프레임으로 하고[26], 추출된 대표프레임에 대한 시각적 특징의 기술자로서 [27]에서 제안한 ESD(Edge-based Spatial Descriptor) 방법을 이용하였다. ESD는 edge correlogram[28]과 CCV(Color Coherence Vector)[29]를 이용한 에지 기반의 공간 기술자이다. ESD기법은 영상에서 조명 효과를 최소화하기 위해 color vector angle기법에 기초하여 각 화소를 평탄 화소나 에지 화소로 분류하고, 영상에 3 x 3 윈도우 마스크를 적용시킨 후, 최대의 color vector angle을 만들어내는 중심 화소와 이에 이웃한 화소를 이용하여 에지를 검출한다. 추출된 에지 화소 집합에서의 그레이 쌍의 분포는 RGB 색상 영역 상에서 양자화된 색상 간의 edge correlogram에 의해 표현되며, 평탄 화소로서 분류된 집합군의 경우 색상의 분포는 CCV에 의해 나타내어 진다. 이렇게 추출된 특징맵으로부터 에지 기반의 공간 기술자를 취득할 수 있다. ESD는 영상 에지 상의 색상 상관관계를 효과적으로 표현할 수 있고 영상에서의 많은 색상 변화 등에 대해서도 강건함을 제공한다. 또한 영상에서 추출된 평탄 영역에서의 전역적인 색상 분포에 대하여 효과적으로 표현이 가능하다. Ⅲ장에서 제안한 움직임 분류 기법을 통해 각 샷은 {S, P_L, P_R, T_U, T_D, Z_I, Z_D, R}의 8가지 움직임 인덱스 중 하나를 갖게 되고, 이를 해당 샷의 움직임 특징자로 이용한다. 특징 기술자의 색인을 위하여 본 논문에서는 R-tree기법의 효율성을 크게 개선한 R*-tree 색인 기법[7]을 이용하였다. R*-tree 색인 기법은 이미 QBIC[1]에서 그 성능에 대한 평가를 받았으며 현재도 주로 사용되고 있는 기본적인 색인 기법 중 하나이다.

2. 유사도 비교

제안하는 비디오 검색 시스템은 사용자의 질의에 대하여 같은 움직임 인덱스를 갖는 영상을 일차적으로 DB에서 검색한다. 각 샷마다 고유의 움직임 인덱스 번호를 가지고 있으므로, 움직임 검색은 거의 시간을 소요하지 않는다. 다음으로 움직임 검색을 통해 추출된 영상들 간의 시각 특징을 비교한다. 이러한 시각 특징은 ESD알고리즘을 통해 추출된 것으로써 시각 특징 간 유사도 측정은 평탄 영역과 에지 영역에서 각각 추출된 특징 *simC(R, Q)*와 *simS(R, Q)*와의 유사도 함수를 계산함으로써 이루어진다[27].

Ⅴ. 실험 결과

본 논문에서는 MPEG 압축된 다양한 종류의 비디오 영상으로 DB를 구성하고, 제안한 검색 기법을 평가 하였다. 실험 DB는 자연 영상, 드라마, 뮤직 비디오, 교육용 비디오 등의 다양한 영상으로 구성되어 있으며, 총 24개의 동영상에서 취득된 780여 개의 샷으로 이루어져 있다. 그림 13은 실험에 사용된 검색 시스템을 보여주고 있다.

본 논문에서 제안한 비디오 검색의 정확성은 recall, precision과 ANMRR(Average of the Normalized Modified Retrieval Rank)을 통해 이루어진다. 먼저, 각 비디오 질의 q에 대한 DB 에서의 유사 샷의 개수를 NG(q)라 한다. 질의에 대한 검색 시 DB에서 찾은 최초 M개의 검색 결과 중에 유사 샷으로 기(旣) 지정된 표준 기저 영상(ground truth videos)과 비교하여 정확하게 찾은 영상의 수를 n_c, 놓친 영상의 수를 n_m, 잘못 찾은 영상의 수를 n_f라 정의할 때, 질의 비디오 q에 대한 precision과 recall 은 각각 다음과 같은 수식으로 얻어진다.

$$Precision = \frac{n_c}{n_c + n_f} = \frac{n_c}{M} \tag{8}$$

$$Recall = \frac{n_c}{n_c + n_m} = \frac{n_c}{NG(q)}. \tag{9}$$

수식에서 알 수 있듯이 precision은 사용자의 질의를 통해 실제로 검색된 결과 영상과 사용자의 질의와 관계된 DB에서의 표준 기저 영상과의 비율을 뜻하며, recall은 표준 기저 영상과 실제 검색을 통해 사용자에게 보여진 영상과의 비율을 나타내고 있다. Precision과 recall은 모두 [0.0~1.0]의 값을 가지며 높은 값을 가질수록 좋은 검색 성능을 나타낸다. 또한, 다른 성능 평가 기준으로서 MPEG-7에 정의된 ANMRR 을 이용한다.

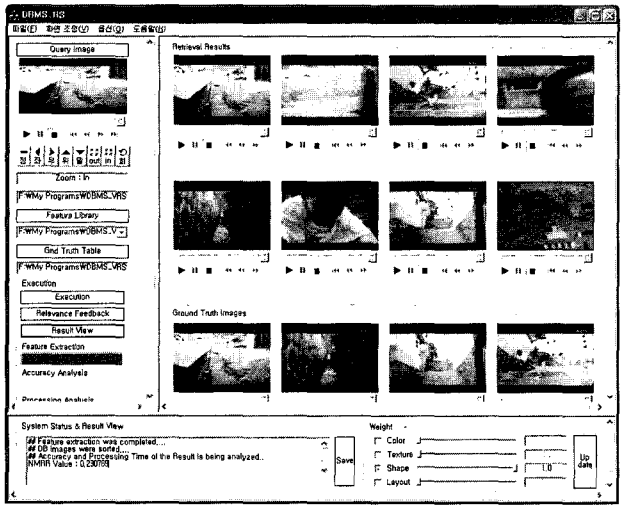


그림 13. 비디오 검색 시스템
Fig. 13. Video retrieval system.

ANMRR은 모든 질의에 대하여 표준화된 수정 검색 순위의 평균을 나타낸다. 표준화된 수정 검색 순위는 [0.0~1.0]의 값을 가지며 낮은 값을 가질수록 높은 검색성능을 나타낸다.

표 3 는 움직임 분류 알고리즘의 성능 평가를 위한 실험 결과를 보여준다. 기존의 파라미터 기반의 방법은 제안하는 방법과 비교 시에 Z, R의 검출 성능이 크게 떨어짐은 물론 P, T의 추출에서도 정확한 임계치 측정이 어려운 단점이 있다. 특히, recall값이 크게 떨어지는 이유는 검색하지 못하고 놓친 Z, R프레임이 매우 많았기 때문이다. 본 논문에서 제안하는 기법은 기존의 방식과 달리 더 다양한 움직임 분류가 가능하며, 보다 정확한 움직임 측정을 가능하게 한다. 그러나, 제안한 방법에 있어서도 R영상의 검색에 있어 잘못 검출한 프레임의 수가 많아 precision의 수치가 높지 않았다. 표 3 에서는 각 방법에 의한 특징 추출 시간 (feature extraction time : FET)의 차이를 또한 비교하고 있다. 파라미터 기반 방법은 선형 최소 제곱법을 이용하고 있기 때문에 제안한 방법보다 더 많은 특징 추출 시간이 소요됨을 알 수 있다.

최종적으로, 제안된 움직임 분류 인덱스와 다른 여러 시각 특징자를 결합함으로써 제안하는 비디오 검색 시스템의 성능을 표 4에서 서로 비교하고 있다. 표 4에서 보듯이, CCV나 correlogram과 같은 시각 특징자를 적용할 때보다 제안된 시스템에서 사용된 ESD특징자를 이용할 때 보다 향상된 시각 내용에 대한 비디오 검색 성능을 보이고 있다. 이는 ESD 방법이 대표 영상을 화소의 주파수에 따라 분할하여 각 분할된 영상의 특징에 의해 CCV와 correlogram을 각각 따로 적용시키고 있기 때문인데, 보통의 자연 영상 내에서 평탄 영역에서의 히스토그램은 일반적으로 에지 영역에서의 히스토그램과 비교할 때 서로 간에 큰 색상 불일치를 보이기 때문에 이러한 영역 사이의 색상분포 차이를 고려하여 영상의 평탄 영역이나 에지 영역으로부터의 특징 추출 방법을 각각 분리함으로써 영상의 검색 성능이 크게 향상된 것이다. 실험에서 CCV와 correlogram은 각각 128개의 빈을 이용하며, 제안된 ESD 방법은 각각 256개의 빈을 이용한다. 비디오 영상에 대한 시각 특징자의 특징 추출 시간을 비교할 때, ESD 기법은 다른 방법들보다 비교적 느린 처리 시간을 가지지만, 온라인 상에서는 처음 추출된 특징 정보가 이미 R*-tree에 의해 정보 색인이 되어 있기 때문에, 결과 검색된 최초의 M개의 영상에 대한 특징 추출 시간은 저장된 자료를 읽기 위한 아주 작은 시간만이 필요하게 되므로, 오프라인 상에서의 특징 추출시간은 사실상 큰 의미를 갖지 않게 된다. 결과적으로, 모든 검색 기법에 대해서 1ms 이하의 값을 가지므로 오프라인 상에서의 CCV나 correlogram에 대한 제안 기법에서의 특징 추출 시간의 증가는 사실상 큰 의미를 갖지 않는다. 추출된 특징에 대한 인덱스 시간은 온라인이나 오프라인 상에서 같은 값을 갖고, 약 1ms 이하의 작은 시간만이 소요된다.

그림 14는 카메라 줌과 시점 변화 등을 포함하고 있는 관련 영상에 대한 검색 결과를 보여준다. 그림 14-(a)는 질의 영상이고, 그림 14-(b)~그림 14-(d)는 검색된 결과이다. 제안된 방법과 여러 다른 검색 기법

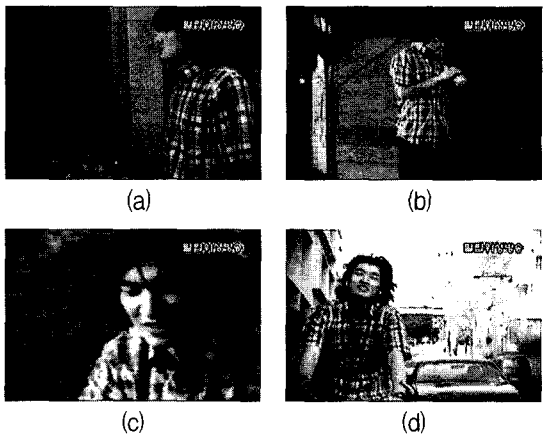


그림 14. 검색 결과 (a) Correlogram (CG) : rank 1, CCV (CV): rank 1, Proposed (P): rank 1 (b) CG : rank 3, CV: rank 7, P: rank 3 (c) CG: rank 5, CV: rank 4, P: rank 5 (d) CG: rank 12, CV: rank 11, P: rank 6.
Fig. 14. Retrieval results with rank.

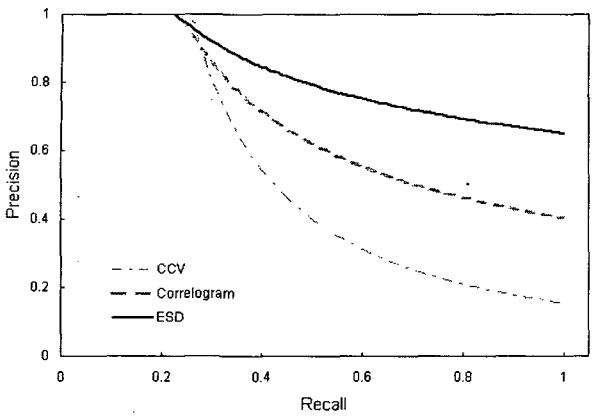


그림 15. Recall과 precision
Fig. 15. Recall and precision.

표 3. 움직임 분류 기법 비교
Table 3. Comparison of motion classification method.

method	precision	recall	FET (ms)
parametric-based[17]	0.44	0.26	0.254
proposed	0.49	0.51	0.113

표 4. 여러 시각 특징 기술자를 이용한 비디오 검색 기법 비교
Table 4. Comparison of content-based retrieval method using various feature descriptors.

method	precision	recall	ANMRR	FET(ms)
CCV	0.32	0.78	0.33	27
correlogram	0.56	0.80	0.28	42
ESD	0.77	0.81	0.22	53

들과의 비교를 위해 각 영상마다 검색된 순위를 숫자로 표기하였다. 즉, 그림 14-(b)에서 'CG: rank 3'의 의미는 시각 특징자로서 correlogram (CG)을 이용하였을 경우 질의 영상(그림 14-(a))에 대한 결과로서 그림 14-(b)의 검색 순위가 3임을 나타낸 것이다. 그림 15는 recall과 precision에 기초한 영상 검색 성능 비교 그래프를 표현하고 있다. 그림 15에서의 각각의 값은 전체적인 성능 비교를 위한 평균값을 나타낸다. 결과 영상이나 그래프에서 보이는 바와 같이, ESD를 이용한 제안 방법이 CCV나 correlogram 등을 이용한 방법과 비교하여 좀 더 우월한 성능을 가진다는 것을 보여주고 있다.

VI. 결 론

본 논문에서는 장면 전환 기법과 움직임 분류 기법을 통해 얻은 시공간적 특징을 이용하여 MPEG 압축된 비디오 영상을 효과적으로 검색하는 방법을 제안하였다. 장면 전환 검출을 통해 분할된 샷에서의 대표 프레임 영상을 이용하여 시각 특징을 추출하고, 움직임 벡터의 분산 등을 이용하여 샷에서의 움직임을 특징지었다. 본 논문에서는 시각 특징자로 color vector angle 방법을 통해 영상에서의 에지를 추출하고, 이를 이용하여 영상의 평탄 영역과 에지 영역 각각에 CCV와 correlogram 기법을 적용시키는 ESD 방법을 이용한다. 또한, 연속된 MPEG 프레임의 움직임 벡터를 프레임의 형태에 맞게 정규화한 후 이를 통해 움직임 객체의 중심좌표와 각 움직임 벡터의 사이각 등을 추출하고, 움직임 벡터를 각도에 따라 양자화 하는 등의 방법을 이용, 분할된 샷의 움직임을 8개의 움직임 인덱스로 분류하여 움직임 특징자로 이용한다. 실험 결과는 제안하는 알고리즘이 영상 색인 및 검색에 있어 매우 효과적임을 잘 보여주고 있다. 현재 우리는 장면 전환 검출 모듈과 움직임 특징 분류 모듈, 영상 검색 시스템을 하나로 통합하고, 사용자에게 편리한 인터페이스를 구축하는 것을 목표로 연구를 진행 중에 있다.

참 고 문 헌

[1] M. Flickner et al., "Query by image and video content: The QBIC system," IEEE computer, vol. 28, no. 9, pp. 23-32, 1995.
[2] V. Ogle and M. Stonebraker, "Chabot: Retrieval from a relational database of images," IEEE computer, vol. 28, no. 9, pp. 40-48, 1995.
[3] J. R. Smith and S.-F. Chang, "VisualSEEK: A fully automated content-based image query system," in ACM Multimedia Conf., 1996.
[4] A. Pentland, R. Picard, and S. Sclaroff, "Photobook: Content-based manipulation of image databases," IJCV, vol. 18, no. 3, pp. 233-254, 1996.
[5] R. Brunelli, O. Mich, C.M. Modena, "A survey on video indexing," IRST-Technical Report 9612-06, 1996.
[6] 이미숙, 황본우, 이성환, "내용 기반 영상 및 비디오 검색 기술의 연구 현황," 정보과학회지, 제15권, 제9호, pp. 10-19, 1997.
[7] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger, "The R*-tree: An efficient and robust access method for points and rectangles", Proc. ACM SIGMOD, pp. 322-331, 1990.
[8] H.J. Zhang, C.Y. Low, S.W. Smoliar, and J.H.

Wu, "Video parsing retrieval and browsing: An integrated and content-based solution," in Proc. ACM Multimedia, pp. 15-24, 1995.
[9] M. Rautiainen, M. Hosio, I. Hanski, M. Varanka, J. Kortelainen, T. Ojala, and T. Seppnen, "TRECVID 2004 experiments at MediaTeam Oulu," Proc. TRECVID Workshop at Text Retrieval Conference TREC 2004, in press, 2004.
[10] B. Adams et al., "IBM Research TREC-2002 video retrieval system," Proc. Text Retrieval Conference TREC 2002 Video Track, 2002.
[11] J. Huang, S. R. Kumar, M. Mitra, W. J. Zhu, and R. Zabih, "Image indexing using color correlograms," CVPR, pp. 762-768, 1997.
[12] S. Mann and R.W. Picard, "Video orbits of the projective group: A simple approach to featureless estimation of parameters," IEEE Trans. Image Processing, vol. 6, pp. 1281-1295, 1997.
[13] T. Yu and Y. Zhang, "Motion feature extraction for content-based video sequence retrieval," Internet Image II, SPIE-4311, pp. 378-388, 2001.
[14] T. Yu and Y. Zhang, "Retrieval of video clips using global motion information," Electron. Lett., vol. 37, no. 14, pp. 893-895, 2001.
[15] W. Chen and S.-F. Chang, "VISMap: An interactive image/video retrieval system using visualization and concept maps," in Proc. IEEE int. Conf. Image Processing, vol. 3, pp. 588-591 2001.
[16] B.L. Yeo, B. Liu, "Rapid scene analysis on compressed video," IEEE Trans. on Circuits and Systems for Video Technology, vol. 5, no. 6, pp. 533-544, 1995.
[17] Y. Nakajima, K. Ujihara, A. Yoneyama, "Universal scene change detection on MPEG-coded data domain," in Proc. SPIE Visual Comm. and Image Proc., pp. 992-1003, 1997.
[18] R. Zabih, J. Miller, K. Mai, "A feature-based algorithm for detecting and classifying scene breaks," ACM International Conference on Multimedia, pp. 189-200, 1995.
[19] J. Meug, Y. Juan, S.F. Chang, "Scene change detection in a MPEG compressed video sequence," Digital Video Compression: Algorithms and Technologies, SPIE-2419, pp. 14-25, Feb. 1995.
[20] E. Izquierdo, J. Xia, and R. Mech, "A generic video analysis and segmentation system," in Proc. IEEE Int., Conf. Acoustics, Speech, and Signal Processing, vol. 4, pp. 3592-3595, 2002.
[21] N.W. Kim, E.K. Kang, et al., "Scene change detection and classification algorithm on compressed video streams," Proc. of the ITC-CSCC 2001, vol. 1, pp. 279-282, 2001.
[22] R. Wang R., T. Huang, "Fast camera motion analysis in MPEG domain," International Conference on Image Processing, vol. 3, pp. 691-694, 1999.
[23] E. Ardizzone, M.L. Cascia, A. Avanzato, and A. Bruna, "Video indexing using MPEG motion compensation vectors," IEEE Int. conf. on multimedia computing and systems, vol. 2, pp. 725-729, 1999.
[24] N.W. Kim, T.Y. Kim, and J.S. Choi, "Probability-based motion analysis using bi-directional prediction-independent framework in compressed domain," Optical engineering, vol. 44, no. 6, 2005.
[25] Y. Deng, C. Kenney, M.S. Moore, and B.S. Manjunath, "Peer group filtering and perceptual color image quantization," Proc. of IEEE Intl. Symposium on Circuits and Systems, vol. 4, pp. 21-24. 1999.
[26] W. Wolf, "Key frame selection by motion analysis," in Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc., 1996.
[27] N.W. Kim, T.Y. Kim, and J.S. Choi, "Edge-based spatial descriptor using color vector angle for effective image retrieval," LNAI, vol. 3558, 2005.
[28] J. Huang, S. R. Kumar, M. Mitra, W. J. Zhu, and R. Zabih, "Image indexing using color correlograms," CVPR, pp. 762-768, 1997.
[29] G. Pass and R. Zabih, "Histogram refinement for content-based image retrieval," IEEE WACV, pp. 96-102, 1996.

저 자 소 개

김 낙 우(정회원)
전자공학회논문지 제 41권 SP편 제 5호 참조

최 종 수(정회원)
전자공학회논문지 제 41권 SP편 제 5호 참조