논문 2012-49SP-2-14

시공간 순차 정보를 이용한 내용기반 복사 동영상 검출

(Content based Video Copy Detection Using Spatio-Temporal Ordinal Measure)

정 재 협*, 김 태 왕*, 양 훈 준*, 진 주 경*, 정 동 석**

(Jae Hyup Jeong, Tae Wang Kim, Hun Jun Yang, Ju Kyong Jin, and Dong Seok Jeong)

요 약

본 논문은 대용량 동영상을 관리하기 위한 빠르고 효율적인 내용기반 중복 동영상 검출 알고리즘을 제안한다. 효율적인 중복 동영상 검출을 위해 대용량의 동영상을 처리하기 쉬운 작은 단위로 나누는 동영상 장면 전환 기반 분할 기술을 적용하였다. 동영상 서비스 및 저작권 보호 관련 사업모델의 경우, 필요한 기술은 아주 작은 구간의 동영상이나 한 장의 영상 을 검색하기보다는 상당한 길이 이상 일치하는 동영상을 파악하는 기술이 필요하다. 이러한 중복 동영상 검출을 위해 본 논문에서 동영상을 장면 전환을 기준으로 분할하여, 나누어진 장면 내에서 움직임 분포 서술자와 대표 프레임을 선택하여 프레임 서술자를 추출한다. 움직임 분포 서술자는 동영상 디코딩 과정에서 얻어지는 매크로 블록의 움직임 벡터를 이용한 장면 내 움직임 분포 히스토그램을 구성하였다. 움직임 분포 서술자는 정합시 고속 정합이 가능하도록 필터링 역할을 한다. 반면 움직임 정보만는 낮은 변별력을 가진다. 이를 높이기 위해 움직임 분포 서술자를 이용하여 정합된 장면 간에 선택된 대표 프레임의 패턴 서술자를 이용하여 동영상의 중복 여부를 최종 판단한다. 제안된 방법은 실제 동영상 서비스 환경에서 우수한 인식률과 낮은 오인식률을 가질 뿐만아니라 실제 적용이 가능할 정도의 빠른 정합 속도를 얻을 수 있었다.

Abstract

In this paper, we proposed fast and efficient algorithm for detecting near-duplication based on content based retrieval in large scale video database. For handling large amounts of video easily, we split the video into small segment using scene change detection. In case of video services and copyright related business models, it is need to technology that detect near-duplicates, that longer matched video than to search video containing short part or a frame of original. To detect near-duplicate video, we proposed motion distribution and frame descriptor in a video segment. The motion distribution descriptor is constructed by obtaining motion vector from macro blocks during the video decoding process. When matching between descriptors, we use the motion distribution descriptor as filtering to improving matching speed. However, motion distribution has low discriminability. To improve discrimination, we decide to identification using frame descriptor extracted from selected representative frames within a scene segmentation. The proposed algorithm shows high success rate and low false alarm rate. In addition, the matching speed of this descriptor is very fast, we confirm this algorithm can be useful to practical application.

Keywords : content-based video retrieval, motion descriptor, video matching, near-duplicate retrieval

Ⅰ. 서 론

최근 컴퓨터와 통신 기술의 급속한 발달로 인해 온라인을 통한 멀티미디어 데이터 전송이 보편화 되었으며, 사용자들은 시간과 장소에 상관없이 다양한 정보에 접근할 수 있게 되었다. 그로 인해 엄청난 양의 디지털 동영상 콘텐츠의 생산과 재생산이 폭발적으로 증가하고 있으며, 이런 자료들을 효율적으로 관리할 수 있는 방법에 대한 필요성이 대두되고 있다. 효율적인 자료의 관리 방법 중에서 가장 기본적인 방법이라고 할 수 있는 것이 바로 검색(retrieval)이다. 동영상 콘텐츠에서의

* 학생회원, ** 정회원, 인하대학교 전자공학과
(Dept. of Electronic Engineering, Inha University)
※ 이 논문은 인하대학교 교내학술연구비의 지원을 받아 수행된 연구임
접수일자: 2011년11월8일, 수정완료일: 2012년1월2일

검색은 원하는 자료를 정확하게 찾아내는 기술이다. 대용량 동영상 콘텐츠에서 원하는 정보를 빠른 시간 내에 검색할 수 있어야 한다. 동영상 검색 기법은 크게 두 가지로 구분할 수가 있는데 첫째는 텍스트 기반 검색이고 둘째는 내용 기반 검색이다. 텍스트 기반 검색 방법은 구축자가 콘텐츠에 달아놓은 텍스트를 이용하여 검색을 하는 주석 기반 검색이다. 이 방법은 제한된 범위 내에서는 효율적인 검색 결과를 제공하지만, 색인 구축자와 검색자의 견해가 일치하지 않을 경우 잘못된 검색이 이루어지므로 검색의 효율이 저하된다. 즉, 영상이 갖는 복잡한 속성을 텍스트만으로 표현하기 어렵다는 문제가 있다[1]. 반면에 내용 기반 검색 방법은 동영상에서 내용으로 기술되는 특징(feature)을 추출하여 이를 기반으로 검색을 수행하는 것으로써, 텍스트가 아닌 동영상이나 정지영상의 입력으로 검색을 할 수 있다. 이는 대용량 동영상 콘텐츠로부터 기존의 텍스트 기반 검색보다 효율적으로 정확하게 검색할 수 있는 방법이라 할 수 있으며, 최근에는 이를 기반으로 하는 다양한 멀티미디어 검색 방법이 제안되고 있다[2]. 내용 기반 검색 방법에서는 특징으로 색상, 질감, 모양 등을 주로 사용한다. 이러한 검색에 대한 연구가 발전되어 멀티미디어 검색 국제 표준인 MPEG-7에 제정되기에 이르렀다[3]. MPEG-7에서는 동영상에 색상, 질감, 모양 외에 추가적으로 움직임을 서술자로 지정하여 영상을 서술하고 이를 기반으로 검색을 수행하는 것을 목적으로 한다[4].

현실성 있고 효과적인 내용 기반 동영상 검출을 하기 위해서는 크게 강인성(Robustness), 독립성(Independence), 고속 정합(High Speed Matching)의 특성이 필요하다. 강인성은 다양한 종류와 정도의 변형에서 원본 동영상 콘텐츠를 구분할 수 있는 성능을 말한다. 독립성은 서로 다른 콘텐츠를 구별할 수 있는 성능을 말한다. 마지막으로 고속 정합은 동영상 콘텐츠끼리 서로 비교하였을 때 얼마나 빠른지를 나타낸다. 대부분의 제시된 기술은 이 3가지를 모두 만족하기 어렵다고 할 수 있다.

본 논문에서는 순차 정보를 이용한 새로운 내용 기반 동영상 검출을 제시하여 원본 동영상과 다양하게 변형, 편집된 질의 동영상의 비교를 통해 동일 콘텐츠임을 밝혀내는 것이 목적이다. 특히, 순차 정보 방법의 취약한 변형이라 할 수 있는 Crop, Pillar and Letter box, Flip에 강인하게 구현하였다. 논문은 총 Ⅳ장으로 구성되며, Ⅱ장 본론에서는 제안한 방법에 대한 설명, Ⅲ장에서는 성능 평가를 수행하고 실험 결과를 도시한다. 마지막으로 Ⅳ장에서는 결론 및 향후 과제에 대해 기술한다.

Ⅱ. 시공간 순차 정보 알고리즘

동영상은 정지영상과 다르게 공간적 정보뿐만 아니라 시간적 정보도 가지고 있다. 본 논문에서는 이러한 동영상의 시공간적인 정보(특징)을 사용하여 각 정지영상(프레임)마다 특징을 간단한 순차 정보 알고리즘으로 추출함으로써 강인성, 독립성, 고속 정합에 적합한 내용기반 동영상 복사 검출을 구현하였다. 알고리즘의 순서도는 그림 1과 같다.

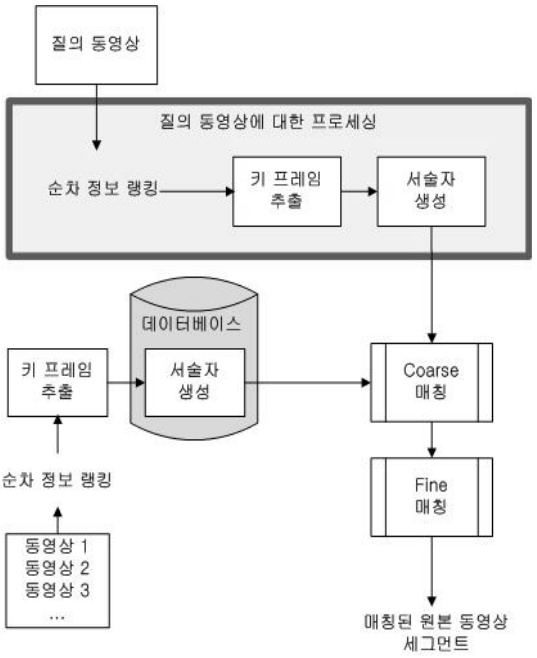

그림 1. 제안한 방법의 순서도
Fig. 1. Flowchart of the proposed method.

1. 서술자 추출 방법

서술자 추출 방법은 프레임에서의 순차 정보 추출과 추출한 순차 정보를 가진 모든 프레임에서 키 프레임을 추출하는 방법으로 나눠진다.

가. 순차 정보 추출 방법

순차 정보는 영상을 블록으로 나누고 각 블록의 화소값 평균을 계산한 후 랭킹 값을 매겨 특징으로 사용하는 것이 일반적이다. 본 논문에서도 이러한 방법을 사용하였다. 그러나 널리 알려진 Hua[5]와 Kim[6]같은 순차 정보 방법의 경우 영상전체를 3x3 또는 2x2의 블록으

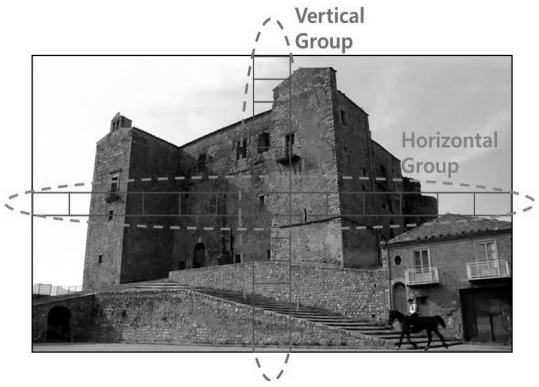

그림 2. 수직그룹과 수평그룹
Fig. 2. Vertical group and Horizontal group.

로 나누어 계산하기 때문에 몇몇 변형에 취약하다. 영상에 Overlay가 된 로고나 자막, 그리고 일부 잘린 영상인 Crop이나 4:3 또는 16:9 비율을 맞추기 위한 Pillar Box와 Letter Box, 마지막으로 요즘 인터넷에서 쉽게 찾아볼 수 있는 좌우가 반전된 Flip변형 등에 취약할 수밖에 없다. 영상 전체를 블록으로 나누어 순차 정보를 추출하면 각 블록 당 화소값 평균은 바뀌므로 랭킹 값이 달라지기 때문이다.

본 논문에서는 이러한 단점을 해결하기 위해, 영상 전체를 블록으로 나누지 않고, 일부만을 나누어 순차 정보를 추출하는 방법을 사용하였다. 장면전환 검출의 효과적인 방법인 연속된 영상에서 수평, 수직 중심 영역의 정보만을 취하는 방법이 있는데[7], 이 방법을 참고하여 영상의 영역을 그림 2와 같이 나누었다.

블록을 나누기 전에 선행 작업으로 동영상에서 추출한 각 프레임들은 Gray Level에 해당하는 Y성분만 사용하였다. 그 다음 블록을 나누는데, 그림 2와 같이 수직 방향으로 13개의 블록을 가지는 수직그룹(Vertical group)과 수평 방향으로 13개의 블록을 가지는 수평그룹(Horizontal group)으로 나누어진다. 각 블록의 크기는 영상의 크기에 비례하며 중앙 블록은 중복된다.

각 블록 당 평균 명암 값을 구한다음 식 (1)과 같이 그룹별로 밝기 순 랭킹을 매겨 랭킹 값(R^g)을 구한다.

$$R^g = \{r_1^g, r_2^g, ..., r_n^g\} \quad (g = 1, 2) \tag{1}$$

g는 수직그룹(1)과 수평그룹(2), n은 그룹 당 블록의 수를 의미한다. 예를 들어 n = 7 일 때, 수직그룹의 블록 당 평균값이 [225, 202, 193, 115, 136, 80, 98]라고 한다면 R^1 = [1, 2, 3, 5, 4, 7, 6] 가 된다. 그룹 당 블록의 수는 성능에 영향을 미치게 된다. Ⅲ장에서는 그림과 같은 13 Vertical & 13 Horizontal group(13V13H) 뿐만 아니라 7V7H과 19V19H로 했을 때의 성능도 알아보았다.

나. 순차 정보를 이용한 키 프레임 추출

동영상에서 키 프레임의 추출은 매우 필요하다. 예를 들어, 30초짜리 30fps 동영상의 경우만 해도 900프레임이므로 이 모든 프레임의 특징 정보를 사용하는 것은 저장과 속도 면에서 큰 저하를 가져온다. 따라서 본 논문에서도 키 프레임을 추출하였고, 추출 과정은 그림 3과 같다.

기존의 다양한 키 프레임 추출 알고리즘 중에서 본 논문은 시간적으로 인접한 프레임끼리의 비교를 통해 키 프레임을 선정하였다. 이전 프레임과 현재 프레임의 특징 정보(랭킹 값)을 비교하여 정해진 임계치보다 값이 크면 현재 프레임을 키 프레임으로 선정하는 방법이다. 다음 나오는 식들을 통해 자세히 알 수 있다.

$$S = \{R_1, R_2, R_3, \cdots, R_{B-1}, R_B\} \tag{2}$$

$$d_g(R_{i-1}^g, R_i^g) = \sum_{j=1}^{n} \left| r_{i-1,j}^g - r_{i,j}^g \right| \tag{3}$$

식 (2)의 S는 동영상 시퀀스이고 R_i는 순차 정보인 랭킹 값으로 이루어진 프레임이며 B는 전체 프레임의 개수이다. 인접한 프레임끼리의 차이 값을 구하는 식이 식 (3)이다. 이전 프레임에서 추출한 랭킹 값(R_{i-1})과 현재 프레임에서 추출한 랭킹 값(R_i)을 서로 비교한다. 여기서 i의 범위는 $1 < i \leq B$ 이며, 랭킹 값을 비교할 때는 수직그룹은 수직그룹끼리 수평그룹은 수평그룹끼리 비교한다. 그룹별로 랭킹 값들의 차이에 대한 절대값의 합을 구하게 되면 최종적으로 d_g를 각각 구할 수 있다.

$$if\ [d_g(R_{i-1}^g, R_i^g) \geq Th_K]\ \text{is true},$$
$$R_i\ \text{is key-frame} \tag{4}$$

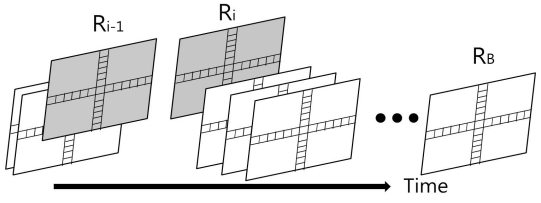

그림 3. 키 프레임 추출 과정
Fig. 3. key frame extraction.

이 d_g값과 정해진 임계치(Key-Frame Threshold = Th_K)를 비교하여 식 (4)와 같이 d_g값이 임계치보다 크거나 같을 경우 현재 프레임은 키 프레임으로 간주된다. d_g값은 2개이므로(수직,수평) 하나의 d_g값만이라도 식 (4)에 해당되면, 키 프레임이다.

2. 정합 방법

정합 단계는 원본 동영상과 질의 동영상을 서로 비교하여 동일한 동영상인지 판단하는 단계이다. 대용량 데이터베이스에서 다양한 변형과 편집이 가해진 질의 동영상을 가지고 원본 동영상을 정확하게 찾는 횟수가 많을수록 성능이 좋은 알고리즘이라고 한다.

정합은 총 2단계로 진행되는데, 첫 번째는 대략적인 후보영역을 선별하는 대략 정합 단계이고, 두 번째는 추출된 후보영역들을 가지고 정확한 시간상 위치 값을 찾는 과정인 정교 정합 단계이다. 여기서 S_Y={Y_1,Y_2,…, Y_N}은 키 프레임을 이용해 추출한 원본 동영상의 서술자이며, S_X={X_1,X_2,…, X_M}은 키 프레임을 이용해 추출한 질의 동영상의 서술자이다. N은 원본 동영상 키 프레임의 총 개수이고, M은 질의 동영상 키 프레임의 총 개수이다. 질의 동영상은 원본 동영상의 일부구간을 변형한 것이므로 N > M 가 된다.

가. 대략 정합 단계

(1) 타임 세그먼트 생성

보다 효과적인 정합을 하기 위해, 질의 동영상에서 타임 세그먼트를 생성한다. 앞서 제안한 알고리즘으로 키 프레임을 추출했을 때, 장면 전환이 잦은 프레임 시퀀스의 경우에는 추출된 키 프레임의 수가 많을 것이고 반대로 장면 전환이 드문 프레임 시퀀스의 경우에는 추출된 키 프레임이 적을 것이다. 또한 장면 전환의 정도에 따라 어떤 부분은 키 프레임이 많이 추출되고 어떤 부분은 상대적으로 적게 추출될 수 있다. 이런 차가 심한 질의 동영상의 경우에는 원본 동영상과의 정합 성능이 떨어진다. 특히 프레임 수(FR)가 원본과 다른 변형의 경우 더욱 그렇다. 이런 점을 보완하기 위해 모든 영상을 10초 단위로 잘라서 시간상 세그먼트로 만들어 각각 저장한다. 예를 들어, 30초 동영상의 경우에는 3개의 타임 세그먼트(S_X ={S^1_X, S^2_X, S^3_X})를 가지게 된다.

10초 단위로 자른 이유는 인터넷에서 흔히 접할 수 있는 동영상의 최소 길이는 10초 내외가 많으며, 동영상들의 총 길이를 보면 10초 단위로 끝나는 경우가 많기 때문이다. 그래서 10초를 의미 있는 단위로 보았다.

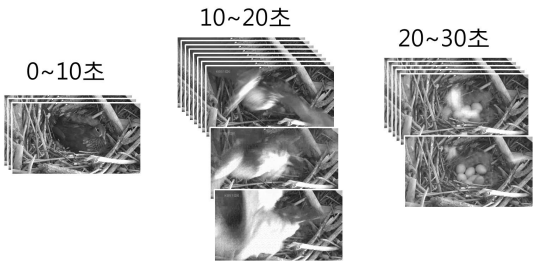

그림 4. 30초 동영상의 타임 세그먼트
Fig. 4. The time segment of 30 second video.

그림 4를 보면 30초 동영상은 3구간으로 나눠지며, 각 구간의 키 프레임의 수는 해당 구간의 장면 전환의 정도에 따라 다르게 분포한다. 10~20초 구간이 움직임이 많은 부분이어서 다른 구간보다 키 프레임이 상대적으로 많다. 이렇게 구간별로 나눠 각각 정합을 하게 되면, 프레임 수가 다른 변형의 경우 보다 좋은 강인성을 얻을 수 있다.

(2) 윈도우 정합 단계

질의 동영상은 앞에서 구한 타임 세그먼트 생성으로 10초 간격으로 나눠졌다. 이 10초 간격으로 나눠진 질의 동영상의 타임 세그먼트를 가지고 원본 동영상에 윈도우 정합을 한다. 그림 5의 경우, 질의 동영상의 길이가 30초이므로 3개의 타임 세그먼트로 나눠진다. 나눠진 타임 세그먼트들을 가지고 원본 동영상에 대해 윈도우 정합을 수행하는데, 여기서 중요한 것은 그룹별로 (수직,수평) 윈도우 정합을 하는 것이다.

$$CM^t_g(S^t_X, S_Y) = \left(\sum_{i=1}^{M_t} I(d_g(X^t_i, Y_j) \le Th_c)\right) / M_t \quad (5)$$

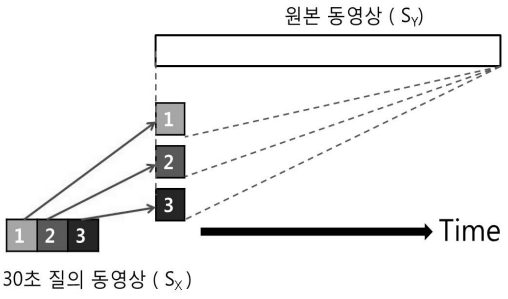

그림 5. 윈도우 정합 과정
Fig. 5. Window matching process.

$$d_g(X_i, Y_j) = \sum_{z=1}^{n} \left| x_{i,z}^g - y_{j,z}^g \right| \quad (6)$$

식 (5)에서 t의 범위는 $1 \le t \le T$이며, 그림 5의 경우는 T=3이 된다. $d_g(\cdot)$은 식 (6)에 나와 있으며 식 (3)과 유사한 프레임간의 랭킹 값 차이 식이다. I(cond)는 조건 cond가 만족할 경우 1, 그렇지 않을 경우 0이 된다. 식 (5)에서의 cond는 $d_g(\cdot) \le Th_c$(Coarse Threshold)이다. M_t는 각 타임 세그먼트의 프레임 수 이므로, 식 (5)에 따라 CM^t_g는 최소 0에서 최대 1의 값을 갖는다.

$$\begin{aligned} &if\,(CM_1^t > CM_2^t) \rightarrow \text{use Vertical Group} \\ &else \rightarrow \text{use Horizontal Group} \end{aligned} \quad (7)$$

타임 세그먼트를 수직그룹과 수평그룹으로 각각 윈도우 정합을 하면, CM^t_1과 CM^t_2의 값을 구하게 된다. 여기서 식 (7)에 의해 두 그룹 중, 값이 큰 그룹만 취한다. CM^t_g값이 큰 그룹만 취한다는 것은 원본과 더 유사한 그룹만 취한다는 의미이다. 하나의 타임 세그먼트가 식 (7)에 의해 수직그룹을 취하게 되면 나머지 타임 세그먼트들도 수직그룹을 취할 경우가 많다. 그 이유는 영상의 변형에 따라 모든 프레임들이 하나의 수직그룹 또는 수평그룹을 취하게 되기 때문이다. 예를 들어, 좌우가 반전된 Flip 변형의 경우에 원본 동영상과 비교하면 수평그룹 서술자보다 수직그룹의 서술자가 유사하기 때문이다. 이런 그룹선택 방법은 순차 정보 방법의 취약한 변형들에게 강인한 특징을 갖는다.

취한 CM^t_g값이 0.5를 넘으면 그 타임 세그먼트의 후보영역으로 선정되며 그 중에서 가장 큰 값을 가진 CM^t_g가 최종 후보영역이 되어 다음 정합단계인 정교 정합으로 넘어간다. 최종 후보영역은 타임 세그먼트마다 최대 하나씩 갖게 된다.

나. 정교 정합 단계

대략 정합 단계에서 얻은 각 타임 세그먼트의 최종 후보영역을 가지고 정교 정합 단계를 통해 정확한 정합 구간을 얻는다. 이때 그림 6과 같이 각 최종 후보영역의 위치를 기준으로 2M 길이를 갖는 서브 세그먼트(S^t_Y)를 원본 동영상에서 구하여 Smith -Waterman 알고리즘[8]을 이용한 정합을 각각 수행한다.

정교 정합은 다음 식 (8)과 같이 정의된다.

$$FM^t(S_X, S_Y^t) = (\alpha N_{match} + \beta N_{miss} + \gamma N_{gap}) / M \quad (8)$$

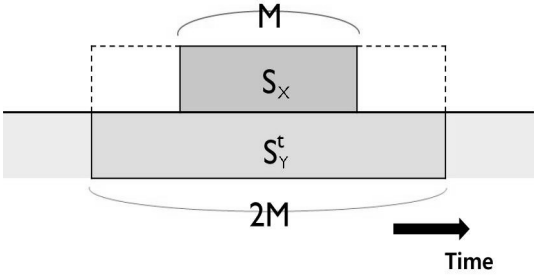

그림 6. Smith-Waterman 알고리즘 정합
Fig. 6. Matching of Smith-Waterman algorithm.

여기서 N_{match}와 N_{miss}, N_{gap}은 각각 일치된 수와 대치된 수 그리고 삽입된 수를 의미한다. α,β,γ는 각 변수에 대한 가중치이며, α=1,β=-0.5,γ=-1로 두었다. 가중치의 각 값들은 α가 β,γ보다 크고, β가 γ보다 크다는 규칙을 적용한 일정한 값이다. 일치 되었나 판단하기 위해 차이값 식인 식 (6)의 $d_g(\cdot)$가 쓰이며, 대략 정합 때 취한 그룹만 가지고 계산한다. 이때 필요한 임계치가 Th_F(Fine Threshold)이며, $d_g(\cdot) \le Th_F$일 경우 일치되었다고 판단한다. 일치가 아닌 경우, 경우에 따라 대치이거나 삽입이 된다[8].

마지막으로, 각 타임 세그먼트의 정교 정합 결과값(FM^t)을 서로 비교하여 가장 큰 값을 갖는 구간을 찾는다. 이 구간이 원본 동영상과 질의 동영상의 최종 정합 구간이 된다.

Ⅲ. 실험 및 결과

1. 실험 조건

본 논문은 다양한 변형을 가한 질의 동영상으로 원본 동영상의 정확한 위치를 찾는데 목적이 있다. 다양한 동영상 데이터베이스를 구축하여 실험에 임하였다.

먼저, 동영상의 장르는 흔희 접할 수 있는 TV 프로그램과 인터넷 동영상으로 구성되었다. 총 7종류로 애니메이션, 다큐멘터리, 드라마, 영화, 뉴스, 쇼 프로그램, 스포츠이다. 원본 동영상의 포맷은 MPEG4이고, fps는 24~30, 장르마다 약 20개씩 총 153개로 구성하였다. 크기는 704x396, 640x352, 704x400, 720x480 등으로 원래 동영상의 크기를 그대로 사용하였고 재생시간은 10분이다. 질의 동영상에는 다양한 16종류의 변형을 주었으며, 시간은 10~30초로 하였다. 변형의 종류와 정도는 표 1과 같이 구성하였다.

변형의 정도는 너무 약하거나 심하게 주지 않고 현실

표 1. 변형 종류와 변형 정도
Table 1. Modification type and degree.

변형 종류	변형 정도
밝기 증가 (BC_H)	전체 밝기 10% 증가
밝기 감소 (BC_L)	전체 밝기 10% 감소
좌우 잘림 (C_LB)	영상 크기 10% 좌우 잘림
상하 잘림 (C_PB)	영상 크기 10% 상하 잘림
대비 증가 (CT_H)	전체 대비 10% 증가
대비 감소 (CT_L)	전체 대비 10% 감소
리사이징 (DS)	영상 크기 x 1/2
좌우 반전 (FP)	영상의 좌우 반전
프레임 율 (FR)	프레임 율 변화 (15 fps)
흐림 효과 (GB)	가우시안 Blur x 1.2
로고 삽입 (Logo)	영상의 5%크기 로고 삽입
흑백 영상 (Mono)	Y 성분 추출 영상
PL 박스 (PLB)	Pillar 또는 Letter Box 적용
압축 변경 (SC)	AVI 포맷의 중화질로 인코딩
선명 효과 (Sh)	Sharpen x 1.2
자막 삽입 (TLO)	영상 하단에 자막 삽입

성있게 하였다. 로고나 자막은 인터넷에서 흔히 볼 수 있는 정도로 적용하였고, Pilar Box와 Letter Box의 경우는 이미지가 4:3 또는 16:9에 비례하도록 Pilar Box 또는 Letter Box 중 하나만 적용하였다.

본 논문에서 실험 평가의 대한 기준은 식 (9)과 (10)에서 정의한 강인성을 측정하는 Recall과 독립성을 측정하는 Precision을 사용하였다.

$$Recall = \frac{D}{D + D_{miss}} \tag{9}$$

$$Precision = \frac{D}{D + D_{false}} \tag{10}$$

D는 원본을 검출한 횟수, D_{miss}는 원본을 검출하지 못한 횟수, D_{false}는 잘못된 영상을 검출된 횟수를 나타낸다. Recall은 100%에 가까워질수록 정확한 원본을 검출한 결과를 나타내며, Precision은 100%에 가까워질수록 잘못 검출한 횟수가 적다는 것을 의미한다. 그리고 정합 속도는 모든 원본 동영상과 질의 동영상을 정합한 후 나온 속도를 전체의 정합 횟수로 나누어 평균을 낸 속도이다.

2. 실험 결과

실험 결과는 본 논문에서 제시한 3개의 임계치에 따른 성능 결과와 제안한 방법과 기존 방법과의 비교 성능 결과로 나누었다.

가. 임계치에 따른 결과

본 논문에서 쓰이는 임계치는 키 프레임 추출 시 쓰이는 Th_K, 대략 정합 시 쓰이는 Th_C, 정교 정합 시 쓰이는 Th_F이다. 모든 실험은 13V13H에서 하였다.

(1) Key-Frame Threshold (Th_K)

표 2는 Th_K의 값에 따른 평균 Recall 과 정합 속도를 비교한 표이다. 값은 2, 4, 6, 8, 10으로 두었다.

Th_K 값이 작으면 강인성은 높아지나 정합 속도는 느려진고, 값이 크면 강인성은 낮아지나 정합 속도는 빨라진다. 그러나 이러한 상충(Trade-off)관계는 선형적이지 않으므로 표 2를 통해 적절하게 고려하여 4가 적당한 값이라고 판단된다.

표 2. 키 프레임 임계치에 따른 Recall과 정합 속도
Table 2. Recall & Matched speed according to Key-frame Threshold.

	2	4	6	8	10
평균 Recall(%)	96.4	96.2	93.1	87.8	84.8
정합 속도(s)	2.3	0.9	0.6	0.2	0.1

(2) Coarse Threshold (Th_C)

그림 7은 변형에 따른 Th_C Recall 비교 그래프이다. Th_C값을 16, 24, 32로 두었으며, 변형에 따라 강인성에서 차이가 있었다. 차이가 두드러지는 변형은 FR과 PLB인데 먼저 FR을 살펴보면 Th_C값이 클수록 좋은 성능을 보인다. Th_C값이 크다는 것은 Fine 정합에 쓰일 후보영역을 더 대략 뽑는다는 의미이다. FR변형의 경우에는 원본 동영상이랑 키 프레임을 뽑아 쓴다고 해도 프레임 수에 차이가 있기 때문에, Coarse 정합에서 대략 뽑아야 Smith-Waterman 알고리즘을 쓰는 정교 정합에서 좋은 성능을 얻을 수 있다. 반대로 다른 변형들은 Th_C값일 작을수록 좋은 성능을 보이며, 특히 PLB변형의 경우 정도가 크다. 보다 세밀하게 정확한 후보영역을 선정해야 정교 정합 때 좋은 결과가 나오는 것은 당연하며, PLB나 C_LB, C_PB같은 영상 크기의 한쪽 비율에 영향을 주는 변형의 경우에는 더욱 그렇다. 본 논문에서 제안한 그룹선택에 따른 정합을 쓴다 하더라도 변형 서술자의 각 블록 크기는 원본과 조금 차이가 나기 때문에 Th_C값이 크다면 엉뚱한 영역이 후보영역

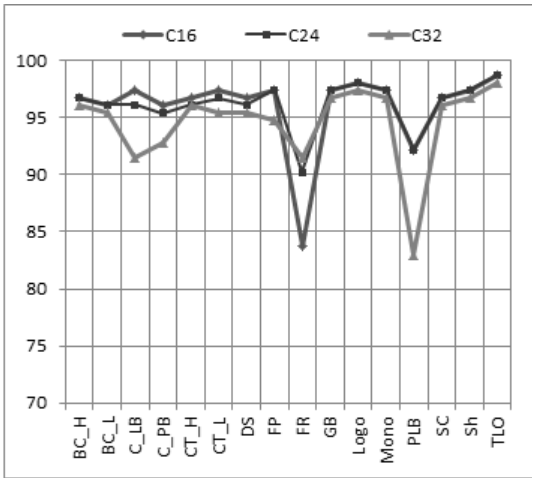

그림 7. Coarse Threshold에 따른 Recall 그래프
Fig. 7. Recall graph according to Coarse Threshold.

으로 선정될 수 있기 때문이다. 이러한 상충관계를 고려하여, 모든 변형에서 90%이상 좋은 성능을 보인 24가 적당한 값이라고 판단된다.

(3) Fine Threshold (Th_F)

그림 8은 변형에 따른 ThF Recall 비교 그래프이며, ThF값을 2, 6, 10로 두었다. 변형에 따라 강인성 차이가 있었으며, ThC때와 마찬가지로 변형들에서 상충관계가 있다. 결과적으로, 너무 크지도 작지도 않은 6을 적당한 값이라고 판단된다. 값을 6으로 하면 10으로 했을 때보다 우수한 성능을 보이며, 값을 2로 했을 때 몇몇 변형에서는 조금 낮았으나 C_LB, C_PB, PLB에서 훨씬 우수한 강인성을 보인다.

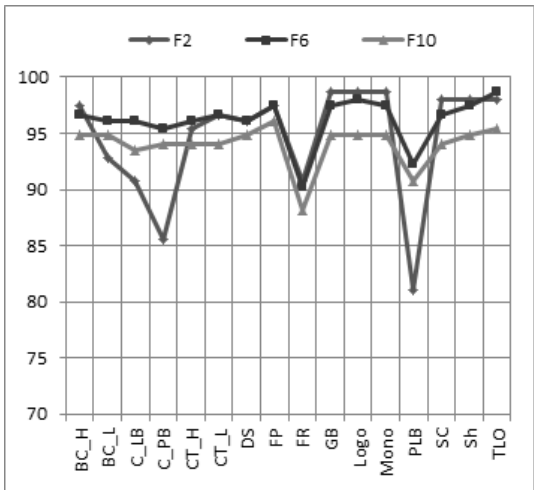

그림 8. Fine Threshold에 따른 Recall 그래프
Fig. 8. Recall graph according to Fine Threshold.

나. 기존 방법과 비교 결과

블록 수에 따른 제안한 방법과 기존의 방법[5]과의 성능을 비교한 것이 그림 9와 표 3이다. 제안한 방법의 블록 수는 7V7H, 13V13H, 19V19H이며 각각의 3개 임계치들은 최적으로 조절하였다.

그림 9는 변형에 따른 Recall 그래프이다. 그림을 보면 블록 수에 따른 제안한 방법 모두 기존의 방법(Hua)보다 뛰어난 강인성을 보인 것을 알 수 있다. 특히, 제안한 방법은 그룹선택 정합 방법을 사용하였기 때문에 순차 정보 방법의 취약한 변형인 C_LB, C_PB, PLB, FP에 모두 뛰어난 강인성을 보인다. 또한 FR변형에서는 타임 세그먼트 방법을 사용하였으므로 기존 방법보다 뛰어난 강인성을 보인다.

표 3에 평균 강인성, 평균 독립성, 정합 속도를 모두 비교해 보았다. 먼저, 강인성을 살펴보면 제안한 방법 3가지 모두 기존 방법보다 뛰어난 성능을 보였다. 독립성은 제안한 방법이나 기존 방법 모두 두 단계의 정합(대략, 정교)으로 구분하여 일정 임계치를 넘어서야만 다음 단계로 넘어가는 방법을 통해 100%라는 좋은 결과가 나왔다. 정합 속도에서 제안한 방법은 블록 수가

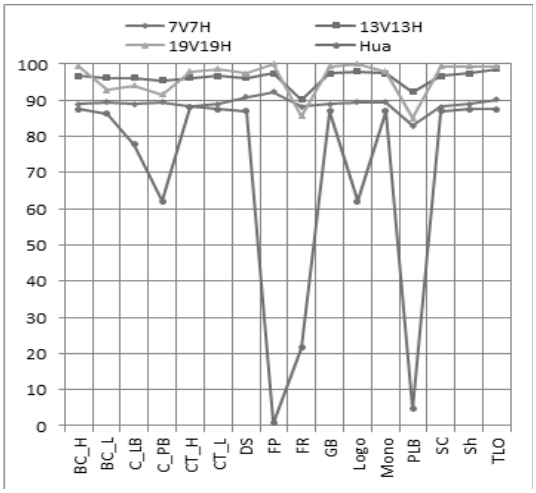

그림 9. 변형에 따른 Recall 그래프
Fig. 9. Recall graph according to Modification.

표 3. 제안된 방법과 기존 방법의 비교
Table 3. Comparison with other method.

	평균 강인성(%)	평균 독립성(%)	정합 속도(s)
7V7H	88.96	100	0.2
13V13H	96.16	100	0.9
19V19H	96.10	100	2.2
Hua	68.84	100	0.2

많을수록 상대적으로 속도가 느렸다. 기존 방법과 비교해보면, 7V7H는 정합 속도는 같으나 강인성에서 더 뛰어난 결과를 가져왔다.

Ⅳ. 결 론

본 논문에서는 시공간 순차 정보를 사용하여 내용기반 동영상 복사 검출에 적합한 방법을 제안하였다.

공간적으로 프레임마다 순차 정보를 수직그룹과 수평그룹으로 나눠 추출하였고, 시간적인 특성을 고려하여 프레임끼리의 비교를 통해 키 프레임을 추출하였다. 정합은 대략 정합과 정교 정합 두 단계로 이루어지며, 먼저 대략 정합 때 질의 동영상에 타임 세그먼트를 한 후 원본 동영상과의 윈도우 정합을 하였다. 그 결과로 나온 최종 후보영역과 선택된 순차 정보 그룹을 통해 최종적으로 정교 정합을 하여 취약한 변형인 Frame rate, Crop, Pillar&Letter Box, Flip에 좋은 성능을 가져왔다. 강인성에서 기존 Hua방법은 평균 68.84%의 결과가 나왔으나 제안한 방법은 최고 96.16% 최저 88.96%라는 우수한 결과가 나왔다. 같은 정합 속도에서도 뛰어난 결과를 보였다.

본 논문은 키 프레임의 추출이 성능을 크게 좌우한다. 보다 적고 정확한 키 프레임을 추출하면, 강인성이 좋아질 뿐만 아니라 서술자가 간결해지므로 정합 속도도 빨라진다. 추후 연구에서 키 프레임의 성능 개선이 필요하다.

참 고 문 헌

[1] C. Jacob, A. Frinkelstein, and D. Salesin “Fast multiresolution image query,” Technical Report 95-01-06, University of Washington, 1995.

[2] Y. Aslandogan, T. Yu, “Techniques and systems for image and video retrieval,” *IEEE Trans. on Knowledge and Data Engineering*, Vol. 11, No.1, p 56-63, Jan. 1999.

[3] J. M. Martinez, “Overview of the MPEG-7 Standard,” ISO/IEC JTCI/SC29/WG11/N4031, Mar. 2001.

[4] L. Cieplinski, M. Kim, J.-R. Ohm, M. Pickering and A. Yamada, “Text of ISO/IEC 15938-3/FCD Information Technology - Multimedia Content Description Interface - Part 3 Visual”, ISO/IEC JTC1/SC29/WG11/N4062, Mar. 2001.

[5] X. S. Hua, X. Chen, and H, J, Zhang, “Robust video signature based on ordinal measure,” International Conference on Image Processing, 2004.

[6] C. Kim and B. Vasudev, “Spatiotemporal sequence matching for efficient video copy detection,” *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 15, No. 1, pp. 127-132, Jan. 2005.

[7] C. W. Ngo, T. C. Pong, and R. T. “Video partitioning by temporal slice coherency.” *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 11, No. 8, Aug 2001.

[8] TF. Smith, MS. Waterman, “Identification of Common Molecular Subsequences,” *Journal of Molecular Biology*, Vol. 147, No. 1, pp. 195 - 197, 1981

저 자 소 개

정 재 협(학생회원)
2009년 인하대학교 전자공학과 학사 졸업
2011년 인하대학교 전자공학과 석사 졸업
2011년~현재 인하대학교 박사 과정
<주관심분야: 영상처리, 패턴인식, 내용 기반 검색, 컴퓨터 비전>

김 태 왕(학생회원)
2011년 인하대학교 전자공학과 학사 졸업
2011년~현재 인하대학교 석사 과정
<주관심분야: 패턴인식, 얼굴인식, 영상처리>

양 훈 준(학생회원)
2011년 인하대학교 전자공학과 학사 졸업
2011년~현재 인하대학교 석사 과정
<주관심분야: 패턴인식, 비디오압축, 영상처리>

진 주 경(학생회원)
2003년 인하대학교 전자공학과 학사 졸업
2005년 인하대학교 전자공학과 석사 졸업
2011년 인하대학교 전자공학과 박사 졸업
2011년~현재 인하대학교 포스트닥
<주관심분야 : 영상처리, 멀티미디어 신호처리, 패턴인식, 내용 기반 검색>

정 동 석(정회원)
1977년 서울대학교 전기공학과 학사 졸업
1985년 Virginia Tech 전자공학과 공학 석사
1988년 Virginia Tech 전자공학과 공학 박사
1988년~현재 인하대학교 전자공학부 교수
1990년~1994년 전자공학회 논문지 편집위원
1990년~1994년 통신학회 논문지 편집위원
2000년~2004년 정보전자공동연구소 소장
2010년~현재 인하대학교 IT공대학장
<주관심분야 : 영상처리, 컴퓨터 비전, 패턴인식, 내용기반 멀티미디어검색>