

특집논문-08-13-5-10

# DTV 방송프로그램의 온라인 불법전송 차단을 위한 비디오 복사본 검출 알고리즘

김 주 섭<sup>a)</sup>, 남 제 호<sup>b)‡</sup>

## Video Copy Detection Algorithm Against Online Piracy of DTV Broadcast Program

Joosub Kim<sup>a)</sup> and Jeho Nam<sup>b)‡</sup>

### 요 약

본 논문은 불법복제된 DTV 방송프로그램의 온라인 불법전송을 차단하기 위한 비디오 복사본 검출 알고리즘을 제안한다. 제안한 알고리즘은 급격한 휘도 변화가 나타나는 방송영상의 비디오 프레임을 선별하여 키프레임(keyframe)을 구성한 후, 키프레임의 시·공간적 특징정보를 추출하여 사전에 등록된 방송영상 저작물들의 특징정보와 비교함으로써 해당 방송영상의 불법복제 여부를 판별하는 동영상 필터링(filtering) 기능을 제공한다. 특히 모든 프레임을 이용하지 않고 키프레임만을 추출함으로써 대용량 방송영상물의 저작권 침해 여부를 효율적으로 판단할 수 있으며, 온라인 전송과정에서 빈번히 적용되는 영상비 변환, 로고 삽입, 자막 삽입, 화질 열화 및 해상도 감소 등의 다양한 동영상 편집효과에도 강인함을 실험을 통하여 확인하였다. 또한 기존 알고리즘들과의 성능비교실험을 통하여 상대적 우수성을 평가하였다.

### Abstract

This paper presents a video copy detection algorithm that blocks online transfer of illegally copied DTV broadcast programs. Particularly, the proposed algorithm establishes a set of keyframes by detecting abrupt changes of luminance, and then exploits the spatio-temporal features of keyframes. Comparing with the preregistered features stored in the database of DTV broadcast programs, the proposed scheme performs a function of video filtering in order to distinguish whether an uploaded video is illegally copied or not. Note that we analyze only a set of keyframes instead of an entire video frame. Thus, it is highly efficient to identify illegal copied video when we deal with a vast size of broadcast programs. Also, we confirm that the proposed technique is robust to a variety of video edit-effects that are often applied by online video redistribution, such as aspect-ratio change, logo insertion, caption insertion, visual quality degradation, and resolution change (downscaling). In addition, we perform a benchmark test in which the proposed scheme outperforms previous techniques.

Keyword : copyright protection, content-based, video copy detection, video filtering.

a) 과학기술연합대학원대학교(UST) 이동통신 및 디지털방송공학

School of Mobile Communications & Digital Broadcasting Engineering, University of Science & Technology

b) 한국전자통신연구원(ETRI) 방통미디어연구부

ETRI, Broadcasting & Telecommunications Media Research Department

‡ 교신저자 : 남제호(namjeho@etri.re.kr)

※본 연구는 지식경제부 및 정보통신연구진흥원의 IT신성장동력핵심기술 개발사업의 일환으로 수행하였음. [2007-S-003-02, 지상파 DTV방송프로그램 보호 기술 개발]

## I. 서론

한류열풍의 침범인 국내 방송프로그램은 최근 인터넷 등을 통하여 국내뿐 아니라 해외로도 대량 불법유통됨으로써 국가적으로 큰 손실이 발생하고 있다. 특히 방송과 통신의 융합이 가속화 될수록 그 저작권 침해의 규모는 더욱 증가할 것으로 우려되고 있다. 이에 따라 최근 정부와 저작권보호단체는 저작권법을 전면 개정하고 불법저작물의 단속과 처벌을 강화하는 등 불법복제 근절을 위한 다양한 노력을 경주하고 있다. 특히, 전면 개정된 저작권법에는 P2P, 웹하드 등 특수한 유형의 온라인서비스제공자(Online Service Provider; OSP)의 '불법저작물 전송을 차단하는 기술적 조치(필터링) 의무화' 조항이 포함되었다<sup>1,2)</sup>.

불법저작물 필터링 기술은 온라인에서 유통되는 동영상, 음악, 게임 등이 저작권자의 허락을 받지 않은 불법복제물인지 여부를 판별함으로써 저작권 침해물의 불법적 인터넷 전송을 차단하는 기술이다. 불법저작물 필터링 기술은 기본적으로 콘텐츠를 식별하는 기능을 가지고 있어야 한다. 즉, 각각의 콘텐츠로부터 고유의 특성이 될 수 있는 특징정보<sup>1)</sup>(feature)를 추출한 후 불법유통되는 저작물의 특징정보와 비교하여 동일한 콘텐츠 인지를 판별하게 된다. 여기서 어떠한 특징정보를 어떻게 이용하는지가 필터링 기술의 성능, 즉 불법저작물 차단율을 결정하는 중요한 역할을 하게 된다.

필터링 기술은 크게 제목, 문자열, 확장자 등에 기반한 텍스트 필터링 방식과 파일마다 존재하는 고유의 헤시값(hash) 비교를 통한 필터링, 그리고 오디오/비디오 인식기술을 활용한 내용기반(content-based) 필터링 등으로 분류될 수 있다<sup>1, 2)</sup>. 텍스트 필터링은 특정한 금치어를 설정하여 검색에 사용하지 못하게 하는 방식으로서, 제목 필터링, 문자열 비교방식, 확장자 필터링 등으로 구분될 수 있다. 금치어 설정에 의한 텍스트 필터링은 다양한 우회와 회피 방법이 가능하기 때문에 쉽게 무력화될 수 있는 소지가 크다. 반면, 내용기반 필터링(content-based filtering)은 텍스트 필

터링과는 다르게 검색어를 사용하지 않고 콘텐츠 자체의 고유한 패턴을 특징정보로 추출하여 필터링을 적용한다. 또한 헤시값에 의한 특징비교와는 달리 파일이 복제되더라도 콘텐츠의 내용 그 자체는 쉽게 변하지 않으므로 가장 우수한 성능을 갖는 필터링 기술로 인식되고 있다<sup>1, 2)</sup>.

비디오의 불법복사본 검출을 위한 내용기반 필터링의 수행 과정은, 우선 오프라인에서 저작권이 있는 동영상(이하 참조 동영상; reference video)의 특징정보를 추출한 후 데이터베이스에 저장한다. 그 후 P2P, 웹하드 등 온라인 사이트에서 업로드 되는 동영상(이하 질의 동영상; query video)으로부터 특징정보를 추출한다. 질의 동영상에서 추출된 특징정보와 데이터베이스에 저장된 참조 동영상의 특징정보를 비교해서 동일한 특징정보가 존재하는지 검사하여, 만약 동일한 특징정보가 데이터베이스로부터 검출된다면 질의 동영상은 불법전송된 저작권 침해물로 간주하고 온라인 전송(업로드)를 차단한다. 반면에 동일한 특징정보가 데이터베이스 내에서 검출되지 않는다면 적절한 동영상으로 판단하고 업로드를 허용한다<sup>2)</sup>.

본 논문은 특히, 대표적인 저작권 침해 대상인 TV 방송영상의 불법복제물 검출에 중점을 둔다. 온라인 사이트에서 불법으로 유통되는 대부분의 TV 방송영상물은, 30분에서 1시간 정도의 시간 길이를 갖는 HD급 화질의 원본 방송영상을 원활한 온라인 재전송을 위하여 DVD급 화질 정도로 포맷 변환된 대용량의 동영상 파일이다. 그러므로 본 논문에서 제안한 비디오 복사본 검출 알고리즘은 온라인 유통시 보편적으로 적용되는 동영상 편집효과에 대한 강인성 향상과 동영상 간의 특징정보를 비교하는데 소요되는 계산처리 속도를 향상시키는데 그 주안점을 두었다. 특히, 제안하는 알고리즘은 동영상의 급격한 휘도 변화가 나타나는 키프레임(keyframe)을 이용하여 간단하면서도 편집효과에 강인한 시·공간적(spatio-temporal) 특징정보에 기반하며, 또한 다양한 동영상 편집 효과를 고려하여 특징정보의 비교과정에서 발생할 수 있는 오차를 최소화하는 방법을 포함한다.

1) 콘텐츠의 고유한 특징정보(feature)는 DNA, 지문(fingerprint), signature 등으로 다양하게 지칭된다. 디지털 워터마킹(watermarking) 기술은 이러한 특징정보를 비가시적으로 콘텐츠 자체에 삽입하여 유통시킨 후, 저작권 확인을 위하여 다시 워터마크를 추출하는 반면, 본 논문에서 다루는 비디오 복사본 검출은 콘텐츠에 어떠한 정보의 삽입 과정 없이, 필요시 콘텐츠 자체로부터 직접 특징정보를 계산해내기 때문에, 워터마크 삽입에 따른 화질열화나 변형(공격)에 따른 워터마크의 훼손과 같은 문제점에서 자유로울 수 있고, 기존에 이미 배포·유통되고 있는 모든 비디오 콘텐츠에도 적용될 수 있다는 점에서, 워터마크 기술과 구분된다.

본 논문의 구성은 다음과 같다. II장에서는 기존의 비디오 복사본 검출 알고리즘을 살펴보고, III장에서는 비디오 복사본 검출을 위한 시·공간적 특징정보의 추출과 비교 알고리즘을 제안한다. IV장에서는 제안한 알고리즘의 성능을 평가한 실험결과를 분석 제시하고, V장에서 결론을 맺으며 본 논문을 마친다.

## II. 기존 알고리즘

동영상의 특징정보 추출을 통한 내용기반 비디오 검색<sup>2)</sup>과 복사본 검출 알고리즘은 각 비디오 프레임의 화면 전체(global) 정보를 이용하는 방법<sup>3)[4][5]</sup>, 공간 및 시간(spatio-temporal) 정보를 혼합하여 이용하는 방법<sup>7)</sup>, 화면의 일부(local)만을 고려하는 방법<sup>11)</sup> 등으로 분류할 수 있다.

컬러 히스토그램(histogram)을 이용한 방법은 비디오 프레임마다 분포되어 있는 화면 전체 색상의 히스토그램을 특징정보로 사용한다<sup>5)</sup>. 하지만 컬러 히스토그램을 이용한 방법은 영상 내에 있는 색상들의 공간적(spatial) 위치 정보를 고려하지 않기 때문에, 동영상의 포맷(예, MPEG, AVI, WMV 등) 또는 화질의 변환 등으로 인한 색상의 작은 변화에도 민감할 수 있다. Naphade<sup>3)</sup>는 MPEG 비디오에서 DC 영상을 이용한 비디오 검색 알고리즘을 제안하였지만 다른 비디오 코딩 방식에는 적용할 수 없는 단점이 있다. Hampapur<sup>8)</sup>는 윤곽선(edge) 기반의 공간적 정보를 이용한 비디오 검색 방법을 제안하였다. 그러나 윤곽선은 화질의 변화나 편집 효과에 따라 민감하게 영향을 받을 수 있다.

Bhat<sup>11)</sup>는 정지영상 검색을 위해 오디널(ordinal) 비교 방법을 제안하였고, Mohan<sup>4)</sup>은 이를 내용기반 비디오 검색에 적용하였다. 오디널 특징정보는 색상 정보와 공간 정보를 모두 가지고 있기 때문에 미세한 색상 변화에 강한 특성을 갖는다. Hoad<sup>6)</sup>은 비디오 샷(shot)의 길이, 화면 내의 컬러 이동 및 중심점(centroid)의 움직임 정보를 이용한 비디오 복사본 검출 방법을 제안하였다.

Hampapur<sup>5)</sup>은 포맷 변환과 해상도 변환 등 동영상 복제

에 적용되는 편집효과에 대한 오디널 방법의 강인함을 다른 알고리즘과의 비교실험을 통하여 증명했다. Kim<sup>7)</sup>은 보다 강인성을 높이기 위해서 오디널 방법에 시간적(temporal) 특징정보를 추가한 알고리즘을 제안하였는데, 특히 영상비(aspect ratio) 변환 편집에 강한 2x2 분할 방식을 사용함으로써 알고리즘의 성능을 향상시켰다.

오디널 방법의 강인성을 개선한 또다른 기법<sup>10)</sup>은 기존의 오디널 방법과 달리 순서값(rank)의 오차를 허용하는 2차원 벡터의 특징정보를 추출함으로써, 비디오의 잡음과 프레임율의 변화에 덜 민감하도록 개선하였다. 또한 Joly<sup>11)</sup>는 해리스(Harris) 검출기<sup>12)</sup>를 이용하여 화면 내의 모서리(corner)를 특징점(interest point)으로 찾은 후, 특징점 주변 영역을 분석하여 특징정보를 추출하였다. 그러나 화면 전체가 아닌 특징점에 의존하는 방법이기 때문에, 대비(contrast) 변형과 같은 편집효과로 인해 특징점을 검출하지 못하는 경우 그 성능이 감소될 수 있다.

기존의 내용기반 비디오 검색 및 복사본 검출 알고리즘은 대부분 비디오 프레임 단위로 특징정보를 추출하고 비교하였기 때문에 실제 대용량의 비디오 데이터에 적용할 경우 많은 계산처리 시간이 소요된다. 최근 영화·방송영상물과 같은 비디오 데이터베이스의 규모와 그 분량은 급격히 증가하고 있으므로, 계산처리 시간은 비디오 복사본 검출 알고리즘의 성능에 있어서 매우 중요한 요소가 아닐 수 없다. 또한 기존 알고리즘은 실제로 불법복제 동영상에 적용되는 일부 편집효과만을 고려하였는데, 최근 동영상 편집 소프트웨어의 발달과 보급으로 UCC 편집과 같이 일반 사용자들도 손쉽게 다양한 동영상 편집효과를 적용할 수 있다. 제안하는 알고리즘은 이러한 상황들을 고려하여 동영상 편집효과에 대한 강인성과 비디오 데이터의 고속처리를 위한 계산속도의 향상에 주안점을 두었다.

## III. 제안하는 알고리즘

본 논문은 화면 전체의 급격한 휘도 변화가 나타나는 비

2) 내용기반 검색은, 원본의 일부 또는 전체와 정확하게 동일한 복사본(exact copy) 이외에도 원본의 내용에 기반한(content-based) 의미적으로 유사한(semantically similar) 콘텐츠도 그 검색 대상에 포함된다는 점에서, 복사본 검출과 구분된다.



그림 1. 자막 및 로고 삽입과 영상비 변환(16:9→4:3): (a) 원본, (b) 상하단 테두리 삽입, (c) 좌우 잘라내기  
 Fig. 1. Caption, logo insertion and aspect-ratio change(16:9→4:3): (a) original, (b) top & bottom box insertion, (c) side cropping

디오 프레임을 기반으로 공간적 휘도 변화와 시간적 길이 변화를 결합하여 비디오 복사본 검출을 위한 시·공간적 (spatio-temporal) 특징정보로 활용한다. 본 절에서는 제안하는 알고리즘의 동영상 특징정보의 추출과 판별 방법을 상세히 설명한다.

1. 급격한 휘도 변화를 이용한 시·공간적 특징정보의 추출

제안한 알고리즘은 영상의 휘도(luminance) 성분을 이용하여 시·공간적 특징정보를 추출한다. 영상 정보의 분석은 휘도뿐만 아니라 색상, 질감, 형태, 움직임, 객체 등 다양한 영상 특징을 통하여 수행될 수 있지만, 대용량 비디오 데이터베이스를 대상으로 제안한 본 알고리즘은 계산 처리의 효율성을 고려하여 가장 간단하게 동영상을 특징화 할 수 있으면서 동시에 강인성을 제공하는 휘도(luminance) 평균값만을 이용한다. 우선, 동영상의 휘도 변화를 측정하기 위해 모든 비디오 프레임을 3×3 블록으로 분할하고 각 블록마다 휘도 평균값을 구한다. 블록별 휘도 평균값을 사용하면 포맷 변환과 국지적인 변형에 강인한 특성을 갖는다. 또한 3×3 블록으로 분할함으로써 각 블록마다 공간적 가중치(spatial weight)를 부여할 수 있다.

그림 1은 자막 및 로고 삽입과 영상비(aspect ratio) 변환에 따른 동영상 편집효과를 나타낸다. 편집 위치가 대부분 화면 중앙을 제외한 모서리나 상하단 영역임을 알 수 있다. 이는 화면의 가운데 영역이 편집에 의하여 훼손될 경우 영상의 중요한 정보를 전달하는 데 있어서 큰 장애가 발생하므로 주로 가운데 영역을 제외하고 편집하기 때문이다. 이



0.2	0.5	0.2
0.5	1.0	0.5
0.2	0.5	0.2

65	43	78
68	92	81
121	117	105

그림 2. 공간적 가중치: (a) 3×3 블록화, (b) 블록 휘도 평균, (c) 블록 가중치, (d) (b)에 (c)의 가중치 적용 결과  
 Fig. 2. Spatial weight: (a) 3x3 block, (b) blocks of averaged luminance, (c) blocks of spatial weight, (d) result of applying (c) to (b)

러한 영상편집의 특성을 고려하여 제안한 알고리즘은 그림 2(c)와 같이 3×3 블록의 화면에서 중앙 1.0, 상하좌우 0.5, 모서리 0.2의 공간적 가중치를 적용한다.

본 논문에서는 특히 급격한 휘도 변화가 나타나는 시간적 영역에서의 비디오 프레임을 키프레임(keyframe)으로 정의하고, 이러한 키프레임에 기반하여 시·공간적 특징정보를 추출한다. 그림 3은 1분짜리 동영상의 시간적 변화에 따른 세 가지 형태의 휘도 정보를 도시하였다. 그림 3(a)은 식(1)에서 정의된 각 프레임의 휘도가중평균(weighted averaged luminance)  $F_w(t)$ 의 시간적 변화이고, 그림 3(b)은 식(2)에서 표현한 가중치가 적용된 프레임 간의 휘도차이(weighted luminance difference)  $\Delta_w(t)$ 의 시간적 변화를 나타낸다. 마지막으로 그림 3(c)은 식(3)을 통해서 계산된 휘도차이변화(differential luminance difference)  $\Delta'(t)$ 의 시간적 변화를 도시한다.

$$F_w(t) = \sum_{i=1}^9 w(i)F(t, i), \tag{1}$$

$$\Delta_w(t) = \sum_{i=1}^9 w(i)\Delta(t, i), \tag{2}$$

$$\Delta'(t) = \Delta_w(t) - \Delta_w(t-1). \tag{3}$$

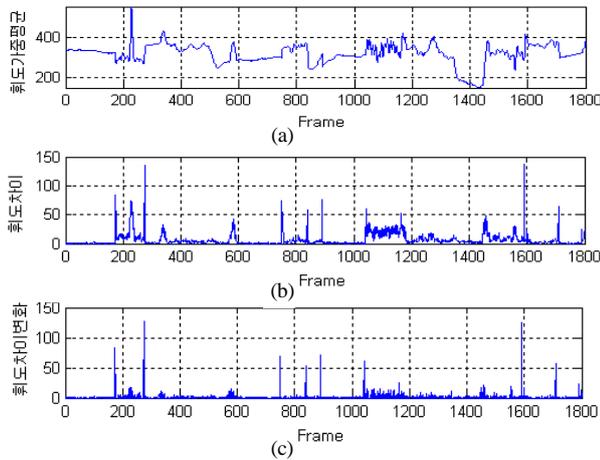


그림 3. 휘도 정보의 시간적 변화: (a) 휘도가중평균  $F_w(t)$ , (b) 휘도차이  $\Delta_w(t)$ , (c) 휘도차이변화  $\Delta'(t)$

Fig. 3. Temporal change of luminance: (a) weighted averaged luminance  $F_w(t)$ , (b) weighted luminance difference  $\Delta_w(t)$ , (c) differential luminance difference  $\Delta'(t)$

식 (1)에서  $F(t, i)$ 는  $t$ 번째 프레임의  $i$ 번째 블록의 휘도 평균값이고,  $w(i)$ 는  $i$ 번째 블록의 가중치 값이다. 식 (2)의  $\Delta(t, i)$ 는  $t$ 번째 프레임의  $i$ 번째 블록의 휘도차이로서  $\Delta(t, i) = |F(t, i) - F(t-1, i)|$ 로 정의되고,  $\Delta_w(t)$ 는 가중치  $w(i)$ 가 적용된  $t$ 번째 프레임의 휘도차이다. 식 (3)의  $\Delta'(t)$ 는 시간에 따른  $\Delta_w(t)$ 의 변화, 즉 휘도차이변화를 나타낸다. 휘도차이변화  $\Delta'(t)$ 는 실제 양수뿐만 아니라 음수도 나타나지만 본 알고리즘에서는 양수의 휘도차이변화만을 고려하고 음수는 계산에서 제외한다.

그림 3(b)과 그림 3(c)을 비교하면 둘 다 비슷한 프레임 위치에서 급격한 휘도 변화를 나타내는 피크(peak)가 표시되지만, 그림 3(c)은 피크와 피크가 아닌 프레임 간의 차이가 더 극명하게 표현된다. 휘도차이  $\Delta_w(t)$ 는 이전 프레임의 휘도 변화와 상관없이 현재의 변화만을 나타내는 반면, 휘도차이변화  $\Delta'(t)$ 는 이전 프레임의 휘도 변화가 반영되므로 급격한 휘도 변화를 찾는데 보다 효율적이다. 예를 들어, 이전에 휘도가 크게 변했다면 현재 프레임의 큰 휘도변화는 급격한 변화로서는 의미가 없다. 반면에 이전 프레임에서 휘도 변화가 작았다면 현재 프레임은 급격한 휘도 변화를 나타내었다고 할 수 있다. 그러므로 휘도차이변화를 이용한 방법은 일정한 임계치(threshold)의 적용에 의해 키

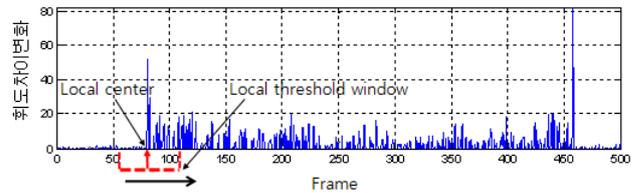


그림 4. 지역임계치 적용  
Fig. 4. Local thresholding

프레임을 선택하는 과정에서, 다양한 동영상 편집효과에 따라 휘도 성분이 변화했을 때 원본 동영상에서와는 다른 키프레임이 선택될 확률을 감소시킨다. 즉, 비디오 복사본 검출 알고리즘의 강인성(robustness) 향상에 기여한다.

제안한 알고리즘은 급격한 휘도 변화를 찾아서 키프레임을 선택할 때, 지역임계치(local threshold)를 적용하므로 동영상의 휘도 특성 변화에 덜 민감하게 키프레임을 선택할 수 있다. 그림 4는 휘도차이변화  $\Delta'(t)$ 에 지역임계치를 적용하는 예를 나타낸다. 일정한 시간길이(지역범위)의 지역임계치 창(local threshold window)을 시간축에 따라 지역중심(local center)을 한 프레임씩 이동하면서, 지역중심을 기준으로 전후로 나누어 지역임계치를 적용한다. 즉, 검사하려는 현재 프레임을 기준으로 지역임계치 창을 이전과 이후로 나눠서 지역임계치를 계산한 후, 어느 한 쪽이라도 임계치를 넘으면 급격한 휘도 변화로 간주한다. 그림 4의 경우, 80번째 프레임의 급격한 휘도 변화는 이전 영역에서 임계치를 넘게 되고, 460번째 프레임에서 나타나는 급격한 휘도 변화는 이후 영역에서 임계치를 넘게 된다. 식 (4)은 휘도차이변화  $\Delta'(t)$ 에 지역임계치를 적용하여 키프레임  $\hat{F}_n$ 을 찾는 과정을 수식으로 표현하였다.

$$\hat{F}_n = \{F(t) \mid \Delta'(t) > \alpha T_L(t), 1 \leq t \leq U\},$$

$$T_L(t) = \min \left[ \frac{1}{L} \sum_{i=1}^L \Delta'(t-i), \frac{1}{L} \sum_{i=1}^L \Delta'(t+i) \right], \tag{4}$$

식 (4)에서  $L$ 은 지역임계치 창 내에서 지역범위로 사용할 비디오 프레임의 개수이고,  $\alpha$ 는 키프레임으로 선정하기 위한 휘도차이변화와 지역임계치 간의 상대적 크기 비교를 위한 파라미터이며,  $U$ 는 동영상의 전체 프레임 개수이다.

지역임계치  $T_L(t)$ 는, 지역임계치 창 내에서 지역중심을 기준으로 이전 또는 이후의 휘도차이변화의 평균값 중에서 작은 값으로 선택된다. 본 알고리즘에서는 실험과정을 통하여 최적의 성능을 보인  $L=30, \alpha=20$ 을 사용했다.

지역임계치 적용을 통하여 키프레임  $\hat{F}_n$ 을 구한 후, 식 (5)을 이용하여 바로 이전에 선택된 키프레임과의 시간적 간격  $D_n$ 을 구한다.

$$D_n = I[\hat{F}_n] - I[\hat{F}_{n-1}] \quad (5)$$

식(5)에서  $I[\cdot]$ 는 프레임 순서 번호를 나타낸 함수이다. 예를 들어, 동영상의 10번째 프레임( $t=10$ )이 첫 번째 키프레임  $\hat{F}_0$ 이고, 15번째 프레임( $t=15$ )이 두 번째 키프레임  $\hat{F}_1$ 인 경우, 두 키프레임 간의 간격  $D_1$ 은 5이다. 본 실험은 초당 30프레임의 비디오 데이터를 기준으로 적용되기 때문에 간격 10은 약 0.3초의 시간간격을 의미한다. 만약 질의 동영상과 참조 동영상의 초당 프레임 비율이 다르다면 초당 30프레임을 기준으로 간격값을 변환한다.

최종적으로, 비디오 복사본 검출을 위한 시·공간적 특징정보  $S_n$ 는 각 키프레임당 9개 블록의 휘도차이값  $\hat{\Delta}_w(n, m)$ 과 바로 이전 키프레임과의 간격  $D_n$ 으로 정의되며, 식 (6)과 같이 표현된다.

$$S_n = \{ \hat{\Delta}_w(n, 1), \hat{\Delta}_w(n, 2), \hat{\Delta}_w(n, 3), \dots, \hat{\Delta}_w(n, 9), D_n \} \quad (6)$$

for  $n = 1, \dots, N$ .

식 (6)에서  $\hat{\Delta}_w(n, m)$ 는 식(2)의 일반 프레임 블록 휘도차이  $\Delta(t, i)$ 와는 다른 표현으로서,  $n$ 번째 키프레임  $\hat{F}_n$ 의  $m$ 번째 블록의 가중치가 적용된 휘도차이(luminance difference)이며,  $N$ 은 키프레임의 개수이다. 본 논문의 실험과정에서 구성된 동영상 데이터베이스를 통하여 50분 길이의 TV 드라마 영상의 경우 평균적으로 610개 내외의 키프레임이 추출됨을 확인하였다.

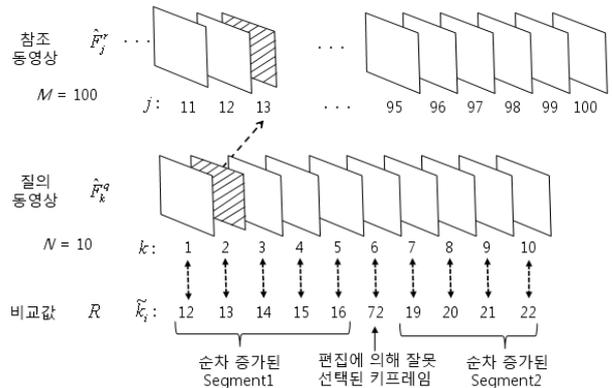


그림 5. 질의 동영상의 키프레임  $\hat{F}_k^q$ 과 참조 동영상의 키프레임  $\hat{F}_j^r$  간의 비교  
Fig. 5. Comparison between keyframe of query video  $\hat{F}_k^q$  and keyframe of reference video  $\hat{F}_j^r$

## 2. 특징정보 간의 비교

질의 동영상이 참조 동영상의 복사본일 경우, 두 동영상 간의 시·공간적 특징정보  $S_n$ 는 서로 동일하므로, 질의 동영상과 참조 동영상 간에 특징정보를 비교해서 복제 여부를 판별할 수 있다<sup>3)</sup>.

그림 5는 질의 동영상과 참조 동영상 간의 특징정보를 비교하는 과정을 도시하였다.  $\hat{F}_k^q$ 와  $\hat{F}_j^r$ 는 각각 질의(query) 동영상과 참조(reference) 동영상의 키프레임이고,  $k$ 와  $j$ 는 키프레임의 순서번호다.  $R$ 은 두 키프레임 간의 특징비교를 통해서 얻은 비교값이다. 그림 5는 질의 동영상이 10개의 키프레임을 가지고 있고 참조 동영상이 100개의 키프레임을 가지고 있을 때, 질의 동영상의 두 번째 키프레임과 참조 동영상의 모든 키프레임 간의 특징정보를 순차적으로 비교하여 그 가운데 가장 유사한 특징정보를 갖는 키프레임을 찾는 과정의 예를 보인다. 식(7)과 식(8)은 두 동영상 간의 특징정보 비교과정을 수식으로 표현하였다.

$$\hat{F}_k^q(k) \approx \hat{F}_j^r(j) : \min_j \{ Diff(k, j) \mid |D_k^q - D_j^r| < T_d \}, \text{ for } \forall j, \forall k \quad (7)$$

3) 전체 복제물일 경우 질의 동영상과 참조 동영상의 키프레임은 완전히 동일하고, 부분 복제물일 경우 질의 동영상 키프레임이 참조 동영상 키프레임의 일부분과 동일하다.

```

begin initialize  $i=0, N=\#keyframe\ of\ Query, SegLen=0, TotalSegLen=0, FailCount=0$ 
do  $i \leftarrow i + 1$ 
  if  $SegLen \leq 3$ 
    then if  $\tilde{k}_i = \tilde{k}_{i-1} + 1$  then  $SegLen \leftarrow SegLen + 1$ 
      else then  $TotalSegLen \leftarrow TotalSegLen + SegLen, SegLen \leftarrow 0$ 
    else if  $SegLen > 3$ 
      then if  $\tilde{k}_{i-1} + 1 \leq \tilde{k}_i \leq \tilde{k}_{i-1} + \gamma$ 
        then  $FailCount \leftarrow 0, SegLen \leftarrow SegLen + 1$ 
        else if  $FailCount < 3$ 
          then  $FailCount \leftarrow FailCount + 1$ 
          else  $TotalSegLen \leftarrow TotalSegLen + \beta SegLen, SegLen \leftarrow 0$ 
      until  $i < N$ 
     $Corr \leftarrow \frac{TotalSegLen}{\beta N}$ 
    if  $Corr > T_c$  then return Copy
    else return No Copy
end

```

그림 6. 참조 동영상과 질의 동영상 간의 키프레임 일치도 계산 과정

Fig. 6. Process of computing a keyframe similarity between reference video and query video

$$Diff(k, j) = \sum_{i=0}^1 \sum_{m=1}^9 |\hat{\Delta}^q(k+i-1, m) - \hat{\Delta}^r(j+i-1, m)| \quad (8)$$

$\hat{F}_k^q$ 는 질의 동영상의  $k$ 번째 키프레임이고  $\hat{F}_j^r$ 는 참조 동영상의  $j$ 번째 키프레임이다.  $D_k^q$ 는 식(5)에서와 같이 질의 동영상의  $k$ 번째와  $k-1$ 번째 키프레임의 시간적 간격이고,  $D_j^r$ 는 참조 동영상의  $j$ 번째와  $j-1$ 번째 키프레임의 시간적 간격이다.  $T_d$ 는 키프레임 간격의 오차 범위로 본 실험에서는 시간적으로 30 frame/sec의 프레임율을 갖는 동영상에서 1개의 프레임 간격에 해당하는 0.03초를 사용하였다. 식(8)의  $Diff()$ 에서  $\hat{\Delta}^q(k, m)$ 는 식(6)에서와 같이 질의 동영상의  $k$ 번째 키프레임의  $m$ 번째 블록의 휘도차이이고,  $\hat{\Delta}^r(j, m)$ 은 참조 동영상의  $j$ 번째 키프레임의  $m$ 번째 블록의 휘도차이다.

식(7)을 살펴보면, 우선 시간적 특징정보인 바로 전 키프레임과의 간격을 검사해서 그 차이가 오차범위  $T_d$  이내 일 경우, 공간적 특징정보인 블록 휘도차이  $\hat{\Delta}(n, m)$ 를 이용하여 유사도를 측정함으로써 시·공간적 특징정보가 함

게 적용되는 비교측정이 수행됨을 알 수 있다. 이때 정확도를 높이기 위해서, 바로 전 키프레임의 블록 휘도차이  $\hat{\Delta}(n-1, m)$ 도 함께 비교하고 그 결과, 가장 차이가 작은 참조 동영상의 키프레임  $\hat{F}_j^r$ 이 질의 동영상의 키프레임  $\hat{F}_k^q$ 과 일치한다고 가정한다. 그림 5에서는 참조 동영상의 13번째 키프레임  $\hat{F}_{13}^r$ 이 질의 동영상의 2번째 키프레임  $\hat{F}_2^q$ 과 시·공간적 특징정보의 비교 측정에서 가장 유사한 것으로 판명되었다. 이와 같은 방법으로 질의 동영상의 모든 키프레임에 대해서 각각 가장 유사한 참조 동영상의 키프레임을 찾으면, 식(9)의 비교값  $R$ 을 구할 수 있다.

$$R = \{ \tilde{k}_1, \tilde{k}_2, \tilde{k}_3, \tilde{k}_4, \dots, \tilde{k}_N \}, \quad (9)$$

$$\tilde{k}_i \in \{ 1, 2, 3, \dots, M \} \text{ for } i = 1, \dots, N.$$

식(9)의  $N$ 은 질의 동영상의 키프레임 개수,  $M$ 은 참조 동영상의 키프레임 개수이다.  $\tilde{k}_i$ 는 시·공간적 특징정보를 비교 측정된 결과, 일치된 것으로 판정된 참조 동영상의 키프레임 번호이다. 여기서 질의 동영상이 참조 동영상의 일부(부분 복제물) 또는 동일(전체 복제물) 하다면  $\tilde{k}_i$ 는 순차

적으로 증가할 것이다. 이와 같은 원리를 이용해서 그림 6에 두 동영상 간의 일치도를 계산하는 과정을 정리, 제시하였다.

그림 5의 비교값  $R$ 에서 보여지는  $Segment$ 는  $\tilde{k}_i$ 가 순차적으로 증가하는 구간을 의미한다. 예를 들어, 그림 5의  $R$ 에서  $i$ 가 1~5 일 때,  $\tilde{k}_i$ 가 순차적으로 증가하기 때문에  $Segment1$ 가 되며, 마찬가지로  $i$ 가 7~10인 구간은  $Segment2$ 를 형성한다. 그림 6의 계산 과정에서  $SegLen$ 은  $Segment$ 의 길이이고,  $TotalSegLen$ 은 모든  $Segment$ 들의 길이를 합한 값이다. 질의 동영상과 참조 동영상이 동일하더라도 질의 동영상이 편집 효과에 의해 변형되면 그림 5에서와 같이 중간에 잘못된 키프레임이 선택될 수 있다. 이러한 경우 비교값  $R$ 이 여러  $Segment$ 로 나뉘게 되고, 검출률의 저하를 야기할 수 있다. 그러므로 본 알고리즘에서는  $\tilde{k}_i$ 가 3번 연속으로 순차 증가한다면(즉,  $SegLen > 3$ ), 두 동영상 간의 일치도가 높을 것이라고 예측하고, 그림 6에서의  $SegLen > 3$  과정으로 진입하여 오차보정의 기회를 부여한다. 오차보정이란,  $\tilde{k}_i$ 의 값이 순차적으로 하나씩 증가하지 않더라도 일정범위( $1 \sim \gamma$ ) 안에서 증가하는 것과 3번까지 잘못된 키프레임이 선택되는 것을 허용하는 것이다. 또한  $Segment$  길이인  $SegLen$ 에  $\beta$ 배의 가중치도 부여한다. 최종적으로 두 동영상 간의 일치도  $Corr$ 는 식(10)을 통해 얻는다.

$$Corr = \frac{TotalSegLen}{\beta N} > T_c \tag{10}$$

식(10)에서  $N$ 은 질의 동영상의 총 키프레임 개수이고, 본 실험에서  $Corr$ 이  $T_c=0.4$ 를 넘으면 질의 동영상은 참조 동영상과 일치 또는 참조 동영상의 일부, 즉 부분 또는 전체 동영상 복제물로 판단하였다. 본 논문에서는 실험을 통하여 최적의 성능을 나타낸  $\beta=1.5$ ,  $\gamma=5$ 를 적용하였다. 그림 6에서  $FailCount$ 는 잘못된 키프레임이 선택될 때 증가하는 값이며, 마지막으로  $Copy$ 와  $No Copy$ 는 각각 질의 동영상의 불법복제 여부를 반환하는 최종 결과이다.

표 1. 데이터베이스 구성

Table 1. Organization of database

장르	시간	편
드라마	735시간	998
연예오락	531시간	499
시사고양	132시간	180
뉴스	40시간	77
광고	1시간 20분	200
총 계	1,439시간 20분	1,954

#### IV. 성능평가실험

##### 1. 데이터베이스 및 질의 동영상 구성

본 논문의 성능평가 실험과정에서 참조 동영상으로 사용된 데이터베이스는 2007년 11월부터 2008년 1월까지 국내에서 방영된 주요 지상파방송의 TV 프로그램으로 구성되었다. 방송프로그램의 구성은 드라마, 연예오락, 시사교양, 뉴스, 광고의 5가지 프로그램 장르, 총 96종류의 프로그램, 1,954편으로서, 데이터베이스 전체는 총 1,439시간 20분에 해당하는 분량이다. 실험영상의 비디오 해상도는 모두 624x352 크기이며, 표 1은 참조 동영상으로 사용되는 데이터베이스의 구성을 나타낸다.

질의 동영상은, 표 1의 데이터베이스로부터 85편의 동영상을 임의로 선정하여 각 편마다 1분, 3분, 5분, 10분 길이의 동영상 일부를 발췌하여 구성하였다. 실제 저작권 침해가 발생하고 있는 대부분의 방송영상물은 전체 프로그램 길이 단위로 온라인 불법전송 되고 있지만, 실험에 사용된 질의 동영상의 최소 길이는 1분으로 정하였다.

질의 동영상에 그림 7의 동영상 편집유형을 적용한 성능평가실험을 통하여 변형에 대한 알고리즘의 강인성을 확인하였다. 그림 7의 (b), (c), (e), (f)의 편집유형은 일반적으로 HD 방송영상물의 온라인 전송 시 적용되는 동영상 편집의 가장 보편적인 형태로서, 특히 상하단 테두리 삽입과 화면의 좌우 잘라내기 편집은 HD 영상의 16:9 종횡비를 4:3 종횡비로 변환할 때 빈번히 발생한다. 그림 7(a)은 16:9의 비율을 갖는 원본 영상이고, 그림 7(b), (c), (e), (f)는 4:3 화면



그림 7. 질의 동영상 편집유형: (a) 원본, (b) 상하단 테두리 삽입, (c) 좌우 잘라내기, (d) 로고 삽입, (e) 상하단 테두리 및 자막 삽입, (f) 좌우 잘라내기 및 자막 삽입

Fig. 7. Edit type of query video: (a) original, (b) top & bottom box insertion, (c) side cropping, (d) logo insertion, (e) logo, top & bottom box and caption insertion, (f) side cropping and caption insertion

비율로 편집된 영상이다. 전체 화면영역을 유지하려면 그림 7(b)처럼 화면의 상하단에 검은 테두리가 삽입되거나 그림 7(c)처럼 좌우의 화면영역이 삭제된다.

온라인으로 유통되는 동영상의 경우, 그림 7(d)와 같이 로고, 이니셜 또는 특정문구 등을 새기기도 하는데, 본 실험에서는 자막 삽입에 대한 강인성 평가를 위하여 질의 동영상 85편 중 외국 방송영상물 40편에 그림 7(e), (f)와 같이 번역 자막을 화면 내에 삽입하였다. 또한 화질 열화에 대한 강인성을 확인하기 위하여 동영상의 재부호화(re-encoding)를 통해 원본 동영상의 비트레이트(bitrate)를 50% 감소하는 편집 과정과 화면 해상도를 약 0.09 비율(1920×1080→176×99)로 축소하는 스케일링(scaling) 편집을 적용하였다. 표 2는 강인성 평가실험에 적용된 질의 동영상의 편집유형을 정리하였다.

## 2. 성능 평가

편집유형에 대한 강인성 평가는 식 (11)의 Recall 및 Precision 측정을 통하여 수행하였다. Recall은 실제 불법복제된 질의 동영상 중에서 정확하게 검출된 질의 동영상의 비율이고, Precision은 검출된 질의 동영상 중에 실제로 불법복제된 질의 동영상의 비율을 나타낸다.

표 2. 질의 동영상에 적용된 편집효과

Table 2. Edit effect on query video

편집 효과	질의 동영상 개수	비고
상하단 테두리 삽입	85	
상하단 테두리 및 자막 삽입	40	외국 방송영상
좌우 잘라내기	85	
좌우 잘라내기 및 자막 삽입	40	외국 방송영상
로고 삽입	85	
화질 열화	85	비트레이트 50%
해상도 감소	85	0.09 스케일링

$$Recall = \frac{True\ Positive}{Total\ True}, Precision = \frac{True\ Positive}{Total\ Positive} \quad (11).$$

식 (11)의 TruePositive는 불법복제된 질의 동영상 중에서 검출된 개수, TotalTrue는 실제로 불법복제된 질의 동영상의 전체 개수, TotalPositive는 검출된 질의 동영상의 개수이다.

먼저 본 알고리즘에서 제안한 3×3 블록 가중치 적용의 효과에 대한 실험결과를 표 3에 제시하였다. 질의 동영상은 3분 길이의 미국 드라마 40편에 7가지 유형의 편집효과를 적용하였고, 표 1의 전체 데이터베이스를 참조 동영상으로 실험하였다. 표 3에서는 각각의 편집유형에 대하여 블록 가중치를 적용하지 않은 경우(1:1:1), 가운데와 상하좌우는 1, 모서리는 0.5의 가중치를 적용한 경우(1:1:0.5), 그리고 제안한 가중치(1:0.5:0.2)를 적용한 경우에 대한 각각의 검출 성능을 나타내었다(공간적 가중치의 배열은 그림 2(c) 참조). 표 3의 결과를 비교해보면, 상하단 테두리 삽입 및 좌우 잘라내기의 편집효과를 적용한 동영상에는 제안한 블록 가중치를 적용하는 것이 비디오 복사본 검출의 성능을 향상시키는데 효과적임을 알 수 있다. 특히 자막 및 로고를 추가로 삽입하였을 경우에 가중치의 효과가 더욱 명확히 나타났다. 한편 변형된 가중치(1:1:0.5)를 적용하였을 때, 가중치가 없는 경우보다는 높은 검출률이 나타났지만, 좌우 잘라내기 및 자막, 로고 삽입 등 복합 편집 효과의 적용에는 제안한 가중치 적용보다 검출률이 저하됨을 확인할 수 있었다.

표 4는 질의 동영상의 시간적 길이에 따른 다양한 편집효과에 대한 실험결과로서, 제안된 비디오 복사본 검출 알고

표 3. 3x3 블록 가중치 적용에 따른 비디오 복사본 검출 성능  
Table 3. Performance of video copy detection according to block weight (단위:%)

편집 유형	성능 평가	가중치 없음 (1:1:1)	가중치 변화 (1:1:0.5)	제한한 가중치 (1:0.5:0.2)
		로그 삽입	<i>precision</i>	100.0
	<i>recall</i>	100.0	100.0	100.0
상하단 테두리 삽입	<i>precision</i>	97.5	97.5	100.0
	<i>recall</i>	100.0	100.0	100.0
상하단 테두리 및 자막 삽입	<i>precision</i>	100.0	100.0	100.0
	<i>recall</i>	80.0	100.0	100.0
좌우 잘라내기	<i>precision</i>	100.0	100.0	100.0
	<i>recall</i>	90.0	100.0	100.0
좌우 잘라내기 및 자막, 로그 삽입	<i>precision</i>	100.0	100.0	97.5
	<i>recall</i>	75.0	92.0	100.0
해상도 감소(0.09)	<i>precision</i>	100.0	100.0	100.0
	<i>recall</i>	100.0	100.0	100.0
50% 화질 열하	<i>precision</i>	100.0	100.0	100.0
	<i>recall</i>	100.0	100.0	100.0

리즘의 강인성을 확인할 수 있다. 본 알고리즘은 영상의 3x3 블록에 가중치(1:0.5:0.2)를 적용하였기 때문에 표4에서 확인한 바와 같이 상하단 테두리 삽입, 좌우 잘라내기, 로그 및 자막 삽입 등으로 야기된 휘도 성분 변화의 영향력을 감소시킬 수 있었다. 또한 50% 화질 열화와 해상도 감소의 경우, 휘도 성분이 변하더라도 급격한 휘도 변화가 나타나는 지점이 크게 달라지지 않기 때문에 높은 검출률을 보였다. 1분 길이의 질의 동영상은 3분 길이의 동영상에 비하여 일반적으로 키프레임의 수가 적게 검출되기 때문에 불법복제된 질의 동영상을 검출하지 못하는 경우가 있었다. 그러나, 전체적으로 1분, 3분 길이의 질의 동영상의 경우에도 평균적으로 98% 이상의 우수한 성능을 보였고, 5분과 10분 길이의 질의 동영상에는 모두 100%의 검출률을 나타냈다. 본 실험과정에서는 10분 이상의 길이를 갖는 질의 동영상에 대한 실험결과를 별도로 제시하지 않았지만, 실제 30분 및 50분 등 질의 동영상이 원본 영상의 길이와 비슷하거나 같은 경우 모두 100%의 검출률을 보였다.

표 4. 편집유형에 따른 비디오 복사본 검출 성능  
Table 4. Performance of video copy detection according to edit type (단위:%)

성능 평가	편집유형					
	상하단 테두리 삽입	좌우 잘라내기	로그 삽입	50% 화질 열하	해상도 감소 (0.09)	
1 분	<i>precision</i>	98.7	98.7	98.7	100.0	100.0
	<i>recall</i>	97.5	96.3	96.3	96.3	95.0
3 분	<i>precision</i>	100.0	100.0	100.0	100.0	100.0
	<i>recall</i>	98.8	98.8	98.8	98.8	98.8
5 분	<i>precision</i>	100.0	100.0	100.0	100.0	100.0
	<i>recall</i>	100.0	100.0	100.0	100.0	100.0
10 분	<i>precision</i>	100.0	100.0	100.0	100.0	100.0
	<i>recall</i>	100.0	100.0	100.0	100.0	100.0

표 5. 자막 삽입에 따른 비디오 복사본 검출 성능  
Table 5. Performance of video copy detection according to caption insertion (단위:%)

시간	성능 평가	자막 삽입 및 상하단 테두리	자막 삽입 및 좌우 잘라내기
1분	<i>precision</i>	100.0	100.0
	<i>recall</i>	100.0	100.0
3분	<i>precision</i>	100.0	100.0
	<i>recall</i>	100.0	100.0
5분	<i>precision</i>	100.0	100.0
	<i>recall</i>	100.0	100.0
10분	<i>precision</i>	100.0	100.0
	<i>recall</i>	100.0	100.0

외국 방송영상물의 경우, 자막 삽입에 대한 강인성 평가를 위해서 미국 드라마 40편에 번역자막을 삽입하였다. 표 5의 실험결과에서 보여지듯이 자막 삽입 여부는 검출률에 큰 영향을 끼치지 않음을 확인하였다. 이는 3x3 블록 가중치의 적용으로 자막 삽입에 대한 영향을 상대적으로 감소시킨 결과이다.

표 6에서는 질의 동영상의 평균 키프레임 개수와 특징정보 추출 및 비교계산 시간을 나타내었다. 질의 동영상은 모두 Xvid 압축형식으로 부호화 되었으므로, 프레임 단위로 복호화 한 후 특징정보를 추출하였다. 즉, 특징정보 추출시간은 비디오 복호화 시간과 특징정보  $S_n$  을 계산하는데 소요되는 시간을 포함한다. 또한 각각의 질의 동영상 특징정

표 6. 평균 키프레임 개수 및 알고리즘 계산 시간

Table 6. Averaged number of keyframe and computation time

질의 동영상 길이	1분	3분	5분	10분
평균 키프레임 수	15.4개	45.4개	67.6개	126.4개
질의 동영상의 평균 특징정보 추출 시간	58.2초	2분 59초	4분 50초	9분 31초
참조 동영상 DB와의 평균 비교 계산 시간	6.5초	18.7초	27.8초	51.9초

표 7. 광고 동영상 실험 구성 및 결과

Table 7. Experiment organization and result of advertisement video

광고 동영상 개수	200	
평균 길이	24.1초	
평균 키프레임 수	10.2개	
검출률	<i>precision</i>	97.0%
	<i>recall</i>	95.5%

보를, 데이터베이스에 저장된 총 1,954편의 참조 동영상(표 1 참조)의 특징정보들과 비교하는 데 소요되는 평균 계산시간을 나타내었다. 실험 환경은 Pentium-4 3.0GHz CPU, 2GB RAM으로 구성하였으며, 알고리즘 구현은 Microsoft Visual C++ 6.0을 이용하였다.

위의 실험결과에서 질의 동영상이 짧은 경우 비디오 복사본 검출률이 다소 감소되는 경향이 있음을 발견하였다. 그래서 본 실험에서는 짧은 길이의 방송영상의 대표적인 예가 되는 방송광고 200편을 사용하여 비디오 복사본 검출 실험을 수행하였는데, 그 실험 구성 및 결과는 표 7에 나타내었다. 방송광고 동영상은 다른 TV프로그램 장르보다 상대적으로 짧은 시간 동안에 많은 시각정보의 변화를 가지므로 많은 키프레임을 추출할 수 있었다. 그러므로 방송광고 영상물의 짧은 시간 길이에도 불구하고 본 알고리즘은 평균적으로 96% 정도의 높은 검출률을 나타내었다.

### 3. 비교 실험

#### 3.1 편집유형에 따른 강인성 비교 평가

제안한 알고리즘의 보다 객관적인 평가를 위하여 기존 방법들과의 성능비교실험을 수행하였다. 특히 포맷 변환과 편집 효과에 강인하다고 알려진, 오디널(ordinal)<sup>[9]</sup> 기반의

대표적인 두 알고리즘인 Mohan<sup>[4]</sup>과 Kim<sup>[7]</sup>이 제안한 방법들과 성능을 비교하여 평가하였다. 그림 8에서 6가지 편집 유형에 대한 기존의 두 알고리즘<sup>[4, 7]</sup>과 제안한 알고리즘의 성능비교실험의 결과를 제시하였다.

오디널 방법<sup>[9]</sup>은 프레임을  $N \times N$  블록으로 나누고 각 블록의 평균 휘도를 이용하여 크기순서(rank)를 측정한다. 이때 각 프레임 내의  $N \times N$  개의 크기순서값이 특징정보로 사용된다. 성능비교실험의 대상이 되는 알고리즘들<sup>[4, 7]</sup>은 프레임 단위로 특징정보의 추출과 비교 과정을 수행하기 때문에 상당한 계산 시간이 소요되므로, 본 성능비교실험에서는 데이터베이스 구성을 간소화하였다. 표 1의 데이터베이스 중 드라마 40편, 연예오락 40편, 시사교양 10편, 뉴스 10편을 포함한 100편의 방송영상을 선별하였다. 이 중에서 국내 방송영상 10편과 미국 드라마 10편을 선택하여 1분, 3분, 10분, 20분 길이로 편집한 후, 상하단 테두리 삽입, 좌우 잘라내기, 로고 삽입, 해상도 감소를 적용하였다. 4.2절에서 실험하였던 화질 열하는 해상도 감소와 비슷한 공격 유형에 속하면서 화면의 왜곡이 덜하므로, 해상도 감소의 실험을 통해 그 강인성 평가를 대체할 수 있다고 판단하여 성능비교실험에서는 제외하였다. 미국 드라마 10편에는 자막 삽입을 적용하여 실험하였다. 본 성능비교실험의 조건은 4.2절의 성능평가에서와 마찬가지로 동영상의 해상도는  $624 \times 352$ 이며, 알고리즘의 파라미터는  $\alpha=20$ ,  $\beta=1.5$ ,  $\gamma=5$ ,  $L=30$  을 동일하게 적용하였다.

본 성능비교실험에서는 제안한 알고리즘의 경우 식(10)의  $T_c$ 를 0.3~0.6 범위 내에서 조절하면서 성능평가를 수행하였고, Kim<sup>[7]</sup>의 경우 해당 알고리즘에서 제시한 최적의 임계치(threshold)인 0.1을 기준으로 0.01~0.35 범위 내에서 조절하면서 실험하였다. Mohan<sup>[4]</sup>의 알고리즘은 최적의 임계치를 제시하지 않았으므로 본 비교실험을 통해 최고의 검출성능을 보인 임계치 0.06을 기준으로 0.04~0.08 범위 내에서 조절하면서 그 민감도를 평가하였다. 특히 최대한 공정하고 객관적인 성능비교평가가 이뤄질 수 있도록 해당 알고리즘마다 최고의 성능을 보이는 유효범위 내에서 임계치를 조절하면서 10번씩 실험을 수행하였다.

그림 8(a)의 상하단 테두리 삽입 결과를 보면 Kim<sup>[7]</sup>과 본

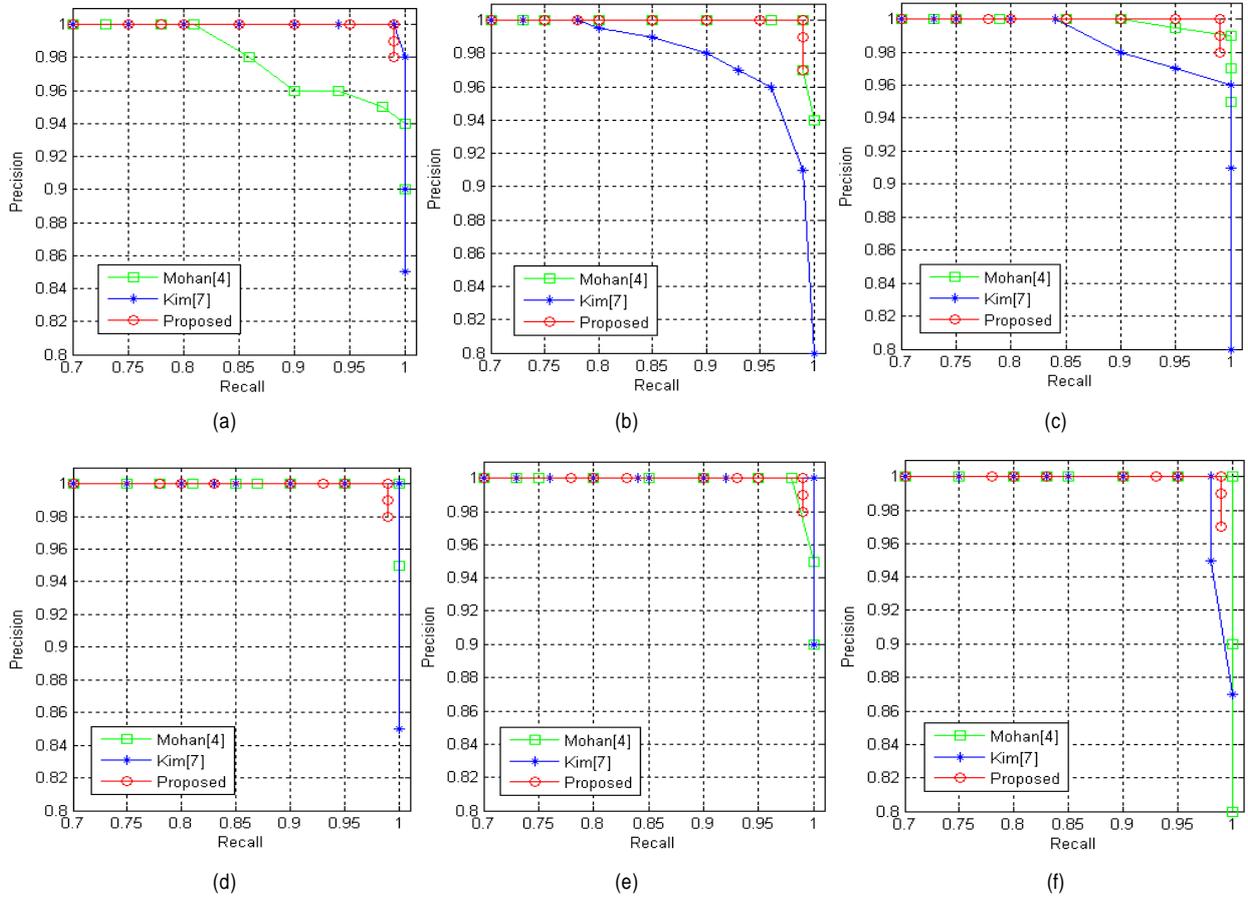


그림 8. 편집유형에 대한 성능비교평가: (a) 상하단 테두리 삽입, (b) 좌우 잘라내기, (c) 로고 삽입, (d) 해상도 감소(0.09), (e) 상하단 테두리 및 자막 삽입, (f) 좌우 잘라내기 및 자막 삽입

Fig. 8. Comparative evaluation of performance according to edit type: (a) top & bottom box insertion, (b) side cropping, (c) logo insertion, (d) reduction resolution (0.09), (e) top & bottom box and caption insertion, (f) side cropping and caption insertion

논문에서 제안하는 방법은 모두 우수한 성능을 보였으나, Mohan<sup>[4]</sup>의 경우 상대적으로 성능이 떨어짐을 확인하였다. Kim의 방법<sup>[7]</sup>은 프레임을 2x2 블록으로 나누기 때문에 화면 상하단에 검은 테두리를 삽입하여도 각 블록의 평균값은 감소하지만, 4개의 모든 블록 평균값이 같은 양만큼 감소하기 때문에 휘도 평균의 크기순서(rank)는 바뀌지 않는다. 그러나 Mohan<sup>[4]</sup>의 경우 3x3 블록으로 나누기 때문에 상하단 테두리의 삽입은 전체적으로 크기순서를 변형시키므로 검출률이 저하된다. 반면에 제안하는 방법은 블록의 휘도 크기순서에 상관없이 전체 동영상의 시간적 휘도 변화를 이용하면서 블록별 기중치를 적용하므로 상하단 테두

리의 삽입과 같은 화면 일부분의 변화에는 큰 영향을 받지 않는다.

한편 그림 8(b)의 화면 좌우 잘라내기 결과에서는 Kim의 방법<sup>[7]</sup>이 다른 방법보다 검출률이 낮았다. 상하단 테두리 삽입과는 반대로 Kim<sup>[7]</sup>은 좌우 잘라내기로 인해 2x2 모든 블록의 휘도 평균이 영향을 받는다. 그림 8(c)의 로고 삽입도 Kim<sup>[8]</sup>의 경우에는 좌우 잘라내기와 같은 이유로 성능이 상대적으로 저하되었다.

그림 8(d)는 원본의 해상도를 약 0.09배 감소하여 적용한 결과이며, 모든 알고리즘에서 높은 검출률을 보였다. 상하단 테두리 삽입과 좌우 잘라내기에 자막 삽입을 결합

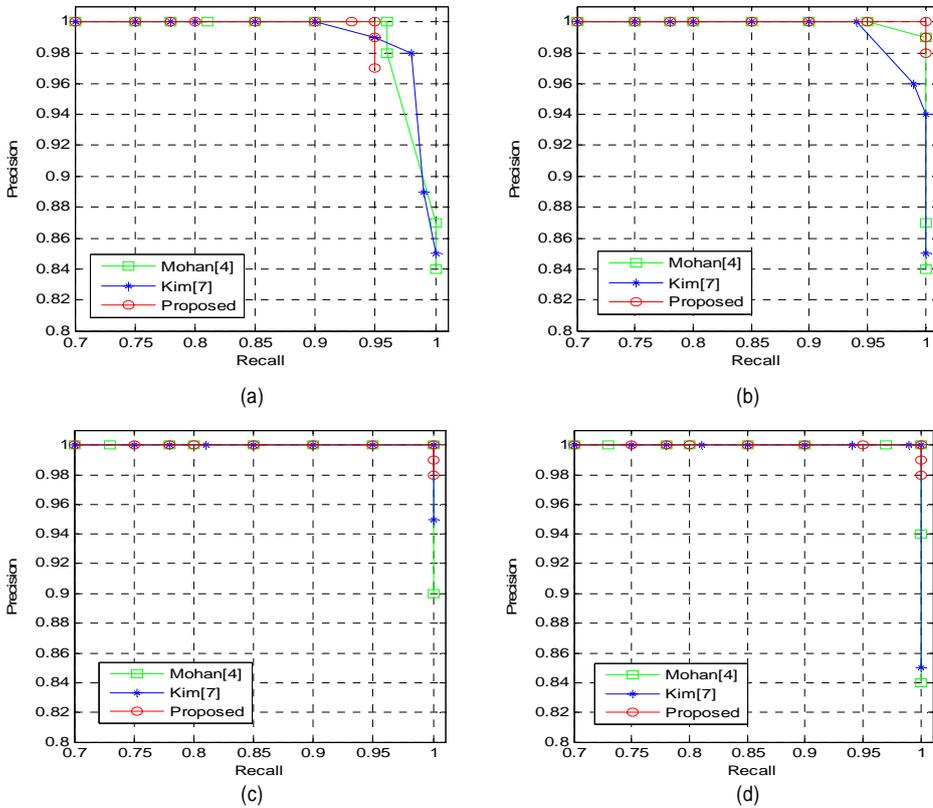


그림 9. 편집 길이에 대한 성능 평가: (a) 1분, (b) 3분, (c) 10분, (d) 20분  
 Fig. 9. Performance evaluation according to edit length: (a) 1 min, (b) 3 min, (c) 10 min, (d) 20 min

한 편집의 경우에는 그림 8(e)와 그림 8(f)의 실험결과에서와 같이, 제안한 알고리즘이 상대적으로 우수한 성능을 보였다.

3.2 질의 동영상의 시간 길이에 따른 성능 비교

그림 8의 실험결과로부터, 제안한 알고리즘이 편집효과에 대한 강인성 측면에서 기존 방법들<sup>[4,7]</sup>과 비교하여 상대적으로 모두 우수함을 확인할 수 있었다. 그러나 80개의 질의 동영상 가운데 한 개의 1분 길이 질의 동영상에서 키프레임이 두 개만 검출된 경우와 같이, 키프레임 수가 매우 적은 짧은 시간 길이의 질의 동영상에서 효과적인 복사본 검출 결과를 보이지 못하였다. 이에 질의 동영상의 길이가 알고리즘들의 성능에 어떠한 영향을 미치는지 확인하기 위해서, 상하단 테두리 삽입, 좌우 잘라내기, 로고 삽입, 해상도 감소의 편집효과를 적용한 4가지 길이의 편집 영상을

대상으로 성능비교실험을 수행하였다. 실험결과 그림 9에서 보여지듯이 질의 동영상의 시간 길이가 짧을 경우 다른 알고리즘들에 비해 본 알고리즘의 *Recall* 값이 약간 감소하였다. 그러나 3분 길이에서는 제안한 알고리즘이 다른 알고리즘보다 *Recall*과 *Precision*에서 모두 높은 성능을 보였다.

나머지 10분과 20분 길이에서는 모든 알고리즘이 100%의 검출률을 나타내었다.

3.3 계산속도에 따른 성능 비교

표 8은 질의 동영상과 참조 동영상 간에 특징정보를 비교하는데 소요되는 계산시간을 나타낸다. 제안한 알고리즘의 비교계산 시간이 다른 알고리즘에 비해서 월등히 짧은 것을 알 수 있다. 이는 동영상의 모든 프레임으로부터 특징정보를 추출하는 다른 알고리즘과 달리 제안하는 알고리즘은 키프레임 기반의 특징정보를 비교 판별하기 때문에, 소요

표 8. 특징정보 비교 계산시간  
Table 8. Computation time of feature comparison

시간 \ 방법	Mohan <sup>[4]</sup>	Kim <sup>[7]</sup>	제안한 알고리즘 성능		
			제안한 알고리즘	Mohan <sup>[4]</sup> 대비	Kim <sup>[7]</sup> 대비
1분	6분 10초	3분 25초	0.1초	0.02%	0.048%
3분	17분 35초	9분 50초	0.26초	0.02%	0.043%
10분	58분 20초	33분 5초	0.79초	0.02%	0.040%
20분	116분 40초	66분 10초	1.39초	0.02%	0.040%

시간을 현격히 단축시키고 있음을 확인할 수 있다. 앞서의 강인도 평가를 위한 비교실험결과로부터 알 수 있듯이, 제안한 알고리즘은 기존 방법들과 비교하여 다양한 편집효과에도 강인하면서 정확한 검출률을 보일 뿐 아니라, 이러한 성능이 모두 상대적으로 월등히 빠른 속도로 그 계산처리가 이루어지는 것을 확인할 수 있었다.

3.4 특징정보의 정보량 비교

제안한 알고리즘은 기존 알고리즘들과 비교하여 특징정보의 저장에 요구되는 메모리량을 절약할 수 있다. Mohan<sup>[4]</sup>의 방법은 프레임마다 9개의 블록 크기순서(rank) 정보를 표현하기 위하여 블록당 최소 4bit가 필요하므로, 프레임당 36bit/frame이 요구된다. Kim의 방법<sup>[7]</sup>은 프레임을 4개의 블록 크기순서로 표현하기 위하여 블록당 2bit가 필요하므로, 프레임당 8bit/frame의 메모리가 요구된다. 그러므로 50분 길이 동영상의 특징정보를 저장하기 위해선 각각 395KB와 87KB의 메모리량이 요구된다. 반면에 제안한 알고리즘은 키프레임을 식(6)의 특징정보  $S_n$ 으로 표현하고 휘도차이  $\hat{\Delta}_w(n, m)$ 는  $\pm 255$ 를 표현하기 위해서 9bit, 간격  $D_n$ 은 최대 50분에 해당하는 프레임 거리를 표현할 수 있도록 17bit가 필요하다. 따라서 50분 길이를 갖는 방송영상물의 평균 키프레임을 610개로 가정했을 때, 50분 길이 동영상의 특징정보를 표현, 저장하기 위해선 평균 7.3KB의 정보량이 요구된다. 즉 비교 알고리즘들<sup>[4, 7]</sup>과 비교하여 대략적으로 최대 98%의 메모리량을 절약할 수 있다.

V. 결 론

본 논문은 동영상의 급격한 휘도 변화로부터 시·공간적 특징정보를 추출하여 해당 동영상의 불법복제 여부를 판별하는 비디오 복사본 검출 알고리즘을 제안하였다. 동영상마다 급격한 휘도 변화가 나타나는 시간적 위치와 공간적 변화 크기가 다르다는 점에 착안하여 키프레임 기반의 시·공간적 특징정보를 추출하였다. 또한 이러한 특징정보는 사전 등록된 방송영상 저작물들의 특징정보와 신속한 비교 처리가 가능할 수 있는 간략한 형태로 표현, 저장될 수 있음을 확인하였다. 특히 온라인에서 유통되는 동영상의 필터링 기능을 목적으로 하는 만큼, 제안된 알고리즘의 실용성을 검증하기 위해서 온라인 유통 과정에서 빈번히 적용되는 동영상 편집 효과에 대한 강인성 평가를 실험하였고, 그 실험결과로부터 기존 알고리즘보다 상대적으로 우수함을 확인하였다. 또한 제안된 알고리즘은 키프레임에 기반하여 간략한 시·공간적 특징정보를 이용하였기 때문에, 모든 비디오 프레임 단위의 특징정보 추출 및 비교 방식을 갖는 기존 알고리즘들에 비해서 특징정보의 간략한 표현과 효율적인 저장이 가능하며, 동시에 고속처리가 가능함을 성능비교 평가실험 과정을 통하여 확인하였다. 제안한 알고리즘의 이러한 장점들은 특히 TV 방송프로그램 및 영화와 같은 방대한 규모의 동영상 데이터베이스를 대상으로, 방송영상 저작물의 불법복제 여부를 신속히 판단해야 하는 온라인 동영상 필터링 서비스의 실용적 구현에 적합한 솔루션으로 활용될 수 있을 것이다.

참 고 문 헌

- [1] “개정 저작권법 시행에 따른 저작권 침해 방지 강화내용(매뉴얼),” 문화관광부, 2007년 7월.
- [2] 김주섭, 남제호, “불법 복제 콘텐츠 필터링 기술 동향 분석,” 방송공학회지, 제12권, 제4호, pp. 371-381, 2007년 12월.
- [3] M. R. Naphade and M. M. Yeung, “A novel scheme for fast and efficient video sequence matching using compact signatures,” in Proc. SPIE Conf. Storage and Retrieval for Media Databases, vol. 3972, pp. 564-572, Jan. 2000.
- [4] R. Mohan, “Video sequence matching,” in Proc. IEEE ICASSP, vol. 6, pp. 3697-3700, May 1998.

- [5] A. Hampapur and R. Bolle, "Comparison of sequence matching techniques for video copy detection," in Proc. SPIE Conf. Storage and Retrieval for Media Databases, vol. 4676, pp. 194-201, Jan. 2002.
- [6] T. C Hoad and J. Zobel, "Detection of video sequence using compact signature," ACM Trans. Information System, vol. 24, no. 1, pp. 1-50, Jan 2006.
- [7] 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.
- [8] A. Hampapur and R. Bolle, "Feature based indexing for media tracking," in Proc. IEEE ICME, vol. 3, pp. 1709-1712, July 2000.
- [9] D. N. Bhat and S. K. Nayer, "Ordinal measure for image correspondence," IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 4, pp. 414-423, Apr. 1998.
- [10] 현기호, "저작권보호를 위한 내용기반 비디오 복사검출의 비디오 정합 알고리즘," 멀티미디어학회논문지, 제11권, 제3호, pp. 315-322, 2008년 3월.
- [11] A. Joly, O. Buisson, and C. Frelicot, "Content-based copy retrieval using distortion-based probabilistic similarity search," IEEE Trans. Multimedia, vol. 9, no. 2, Feb. 2007.
- [12] C. Schmid and R. Mohr, "Local garyvalue invariants for image retrieval," IEEE Trans. Pattern Anal. Mach. Intell., vol. 19, no. 5, pp. 530-535, May 1997.

---

### 저 자 소 개

---



김 주 섭

- 2007년 2월 : 아주대학교 미디어학부(학사)
- 2007년 3월 ~ 현재 : 과학기술연합대학원대학교 이동통신 및 디지털방송공학과 석사과정
- 주관심분야 : 멀티미디어 보호, 영상처리, 디지털방송 미디어, 디지털 콘텐츠 식별체계



남 제 호

- 1992년 2월 : 홍익대학교 전기제어공학과 학사
- 1996년 12월 : University of Minnesota, Electrical Engineering 석사
- 2000년 12월 : University of Minnesota, Electrical Engineering 박사
- 2001년 2월 ~ 현재 : 한국전자통신연구원(ETRI) 방통미디어연구부 선임연구원, 융합콘텐츠보호연구팀장
- 2007년 1월 ~ 현재 : 과학기술연합대학원대학교(UST) 이동통신 및 디지털방송공학과 부교수
- 주관심분야 : 신호처리, 디지털방송기술, 멀티미디어 보호관리, MPEG