

특집논문-02-07-2-01

디지털 텔레비전 수신환경에서의 실시간 비디오 인덱싱과 비선형적 비디오 브라우징

윤 경 로*, 전 성 배*

Real-Time Video Indexing and Non-Linear Video Browsing for DTV Receivers

Kyoungro Yoon* and Sung-Bae Jun*

요 약

지난 10여년간의 디지털 비디오와 관련 멀티미디어처리 기술의 빠른 발전은 비디오 내용의 이해에 기반한 키프레임 기반의 내용 요약과 같은 여러 가지의 비선형적 비디오 브라우징 방법들을 가능하게 하였다. 그러나 스토리보드나 목차와 같은 키프레임 기반의 사용자 인터페이스들은 텔레비전 환경에 익숙한 대다수의 일반 사용자들에게는 여전히 사용하기 어렵다는 단점이 있으며, 이러한 사용자 인터페이스의 구성 또한 실시간 처리를 지원하지 않고 있어 서비스 제공자를 배제한 일반 방송 환경에서 사용하기 어렵다는 단점을 가지고 있다. 본 논문에서는 이러한 단점들을 극복하며 동시에 사용자들이 사용하기 편리한 비선형적 비디오 브라우징 기술들을 제안하며, 이를 지원하기 위한 실시간 비디오 인덱싱 기술을 제안하고 있다. 즉 구조기반의 건너뛰기와 요약보기 기능은 실시간 비디오 인덱싱 기술에 기반하여, 사용자들이 기존의 비디오 레코더의 리모콘과 같은 간단한 사용으로 원하는 장면을 쉽게 찾아갈 수 있도록 하여 주며, 간단한 요약 기능을 제공하여 짧은 시간에 전체의 내용을 이해할 수 있도록 하여 준다.

Abstract

The fast advances in digital video processing and multimedia processing technology over the last decade enabled various non-linear video browsing techniques. Based on the machine-understanding of the video content, non-linear video browsing interfaces such as key-frame based content summarization have been introduced. The key-frame based user interfaces, such as storyboard or table of content, however, are still very hard for conventional TV users to use, and are very hard to implement without the service providers providing additional information for the construction of the key-frame based interfaces. In this paper, non-linear video browsing techniques, which not only overcome previously described drawbacks but also are easy-to-use, and real-time video indexing technology to support the proposed browsing techniques are proposed. The structure-based skipping and skimming help users easily find interesting scene and understand the content in a very short time, using real-time video indexing technology.

I. 개 요

비선형 비디오 브라우징이란 일반적으로 모든 비 연속적 접근을 통한 비디오 브라우징 방법들을 통칭한다. 이러한 비선형적 비디오 브라우징 도구들은 사용자들이 수신한 비디오의 관심있는 부분만을 선택하여 시청할 수 있도록 한다.

PVR(Personal Video Recorder)과 같은 시간 왜곡을 가능하게 하는 하드디스크기반의 디지털 비디오 녹화 장치들은 최근 들어 많은 관심을 끌고 있다. 그러나 기존의 PVR 시스템들은 의미있고, 사용하기 편리한 비선형적 비디오 브라우징 방법을 사용자들에게 제공하지 못하고 있다. TV와 같은 환경에서 비선형 비디오 브라우징 도구가 실질적으로 적용되고 사용되기 위하여는 반드시 고려되어야 할 몇 가지 제한 요소들이 있다. 그 첫째는 사용의 편이를 위하여 최소한의 사용자 입력을 요구하여야 한다. 두 번째는 사용

* LG 전자기술원
LG Electronics Institute of Technology

자의 입력에 대한 반응 지연 시간이 매우 짧아야 한다. 즉, 사용자의 요구를 처리하기 위하여 최소한의 연산 및 자원만을 요구하여야 하며, 사용자가 요구한 액션은 사용자가 인지할 수 없는 지연시간 내에 취하여 져야 한다. 이러한 제한 요소들을 만족시켜 주기 위하여, 실질적으로 적용하는 비선형 비디오 브라우징 도구는 명령의 입력이 간단/단순하면서도 사용자에게 효과적인 기능을 제공하여야 하며, 이러한 기능을 제공하기 위한 인덱싱의 추출은 실시간 연산에 기반하거나 사용자가 인식할 수 없는 최소한의 지연시간만을 요구 하여야 한다.

본 논문에서 제안하는 비선형 비디오 브라우징 도구들은 의미있는 기능 들을 제공하기 위하여 실시간 비디오 인덱싱 기술에 기반하였다. 즉, 제안하는 비선형 비디오 브라우징 기능들은 비디오 인덱싱 분야 연구 결과를 실용화한 것이라 할 수 있다. 비디오 인덱싱의 연구는 장면분할^[1]과 키 프레임 추출에서 시작 하였으며, 비디오 내용의 간략한 표현과 브라우징을 위하여 샷 클러스터링 기술의 개발로 이어졌다^{[2]-[5]}. 그러나 이러한 기존의 비디오 인덱싱의 연구는 실시간 시스템을 지원하기 위한 고려가 되어 있지 않다. 즉 정확성의 향상을 위하여 많은 계산을 필요로 하는 알고리즘을 제안하거나, 오프라인 인덱싱을 전제로, 한 시점에서 과거와 미래의 프레임 정보를 사용하기도 한다. 또한 비디오 내용의 요약이나 하이라이트를 자동으로 추출하는 방법에 최근들어 많은 관심이 모이고 있다^[6].

디지털 비디오 분야의 연구 결과에 기반하여 가진 분야에서도 많은 발전이 있었으며, PVR(Personal Video Recorder)과 같은 시간 왜곡 시스템들이 등장하였다^{[7]-[9]}. 이러한 시간 왜곡 기능을 갖는 셋탑박스 들은 수신된 텔레비전 프로그램에 대하여 인스턴트 리플레이(Instant Replay), 건너뛰기(Forward Skipping), 빠른 배속 보기(Fast Forward) 등과 같은 기초적인 비선형 비디오 브라우징 도구들을 제공한다. 그러나 이러한 종래의 PVR 들은 실시간 비디오 인덱싱 기술에 기반하고 있지 않기 때문에 시간차에 기반한 비선형 기능만을 제공하게 된다. 이러한 종래의 PVR의 빠른 배속 보기 기능은 적절한 오디오 정보를 동반하지 않기 때문에 사용자가 짧은 시간에 비디오의 내용을 이해 할 수 없으며, 또한 배속이 높은 경우 빠른 화면의 전환으로 사용자의 집중도를 떨어뜨리며 동시에 비디오 정보의 이해를 어렵게 한다. 또한 인스턴트 리플레이나 건너뛰기 같은 기능은 정해진 시간차를 기준으로 구현되기 때문에 원하는 장면을 찾아가거나, 원하지 않는 장면을 건너뛰고자 하는 사용자에게 충분히 편리한 기능을 제공하지 못한다.

의미있는 비선형 브라우징 기능을 제공하기 위하여 비디오 내용의 이해는 필수적이다. 그러나 자동적인 기계에 의한 비디오 내용의 완전한 이해는 아직 요원하며, 반면에 장면 분할과 샷클러스터링 기술에 기반한 이야기 구조 추출 기술은 완전하지 않으나 제한된 분야에서 유사한 효과를 줄 수 있다.

본 논문에서는 이러한 기존의 비선형 비디오 브라우징 도구들의 단점을 극복하며 동시에 빠른 비디오 내의 이동과 내용의 이해를 위한 구조 기반의 건너뛰기와 요약보기 기능을 제안하며, 디지털 방송환경에서 제안된 구조 기반의 비선형 브라우징 기능을 가능케 하기 위한 실시간 비디오 인덱싱 기술을 제안한다.

II. 비선형 비디오 브라우징 시스템의 구조

제안하는 비선형 비디오 브라우징 시스템은 그림 1에 보여진 바와 같이 비디오 구조화 엔진, 인덱싱 엔진, 비선형 비디오 브라우저, 저장장치의 크게 4개의 부분으로 나뉘어져 있다.

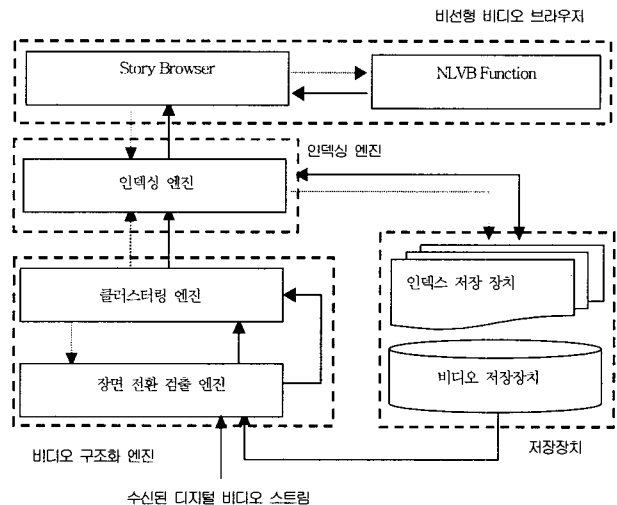


그림 1. 비선형 비디오 브라우징 시스템 구조도
Fig. 1. Architecture of NLVB System

비디오 구조화 엔진은 장면 전환 검출 엔진(SCDE)과 클러스터링 엔진(CE)으로 구성되어 있다. 비디오 구조화 엔진은 디지털 TV 방송 표준에 적용된 바와 동일한MPEG 표준에 맞추어 압축된 비디오 스트림을 입력으로 받아 들여 압축도메인에서 샷정보와 씬정보를 실시간으로 자동 생성하게 된다. 새로운 장면 전환이 검출 되면 이전 장면 전

환 지점과 새로운 장면 전환 지점 사이의 비디오 스트림이 하나의 샷으로 등록되며 등록된 샷의 특징소들이 클러스터링 엔진으로 넘겨지게 된다. 클러스터링 엔진은 입력된 특징소들에 기반하여 썸의 전환이 이루어 졌는지 판단한다. 구현된 시스템에서의 클러스터링 엔진은, 하나의 이야기 단위에는 시각적으로 유사한 여러 샷들이 존재한다는 특징에 기반하여, 논리적 이야기 단위를 추출하게 된다^[4]. 이러한 알고리즘은 영화나 드라마와 같은 장르에 특히 적합하나, 스포츠나 뉴스와 같은 특정한 장르의 비디오의 경우는 또 다른 특징을 갖게 되므로 여러 장르의 비디오 들을 위한 인덱스 정보의 추출/제공을 위하여 각각의 특징적인 다른 장르를 위한 장르 의존적인 클러스터링 엔진이 필요하게 된다. 현재 영화/드라마를 위한 클러스터링 엔진은 본 시스템에 구현되었으며, 뉴스, 스포츠, 뮤직쇼를 위한 클러스터링엔진을 구현할 예정이다. 본 시스템에서 사용된 비디오 구조화 엔진은 다음 장에서 상세히 설명되었다. 본 시스템에서 장면 전환검출 엔진과 클러스터링엔진은 동시에 구동되어, 인식할 수 없는 정도의 지연 시간 내에서, 녹화 중인 비디오의 모든 녹화된 부분에 대하여 구조 정보가 제공될 수 있도록 한다.

인덱싱 엔진(IE)은 비디오 구조화 엔진이 생성한 구조 정보들을 받아 구조 정보 저장장치에 저장하고, 비선형 브라우저가 비선형 비디오 브라우징 기능을 요청하는 경우 관련된 인덱스 정보를 검색하여 브라우저로 전달하는 역할을 한다.

비선형 비디오 브라우저는 사용자가 실제로 사용하는 부분으로 텔레비전과 같은 환경에서 일반적인 비디오 레코더의 사용자 인터페이스와 결합, 연동되게 된다. 따라서 사용자는 비선형 브라우징 기능(NLVB)을 기존의 비디오 레코더의 기본 기능들의 한 연장된 기능으로 인식하며, 간단하게 사용할 수 있도록 하여 준다.

Ⅲ. 실시간 비디오 인덱싱

비디오 인덱싱에는 여러 가지 분야가 있으나, 본 논문에서는 제안하는 비선형적 비디오 브라우징 기능을 구현하기 위하여 필요한 비디오 구조 정보의 추출 방법, 즉 실시간 샷 경계 추출 방법과 실시간 샷 클러스터링 방법을 제안하였다. 실시간 영화/드라마 샷클러스터링 엔진을 구현하기 위하여 다음의 몇 가지 개념들을 정의하였다. 썸이란 의미적 지속성을 갖는 연속적인 샷들의 집합으로 정의한다. 레이블이란 하나의 썸 안에서 시각적 유사성을 갖는 하나 또는 그 이상의 샷들의 집합으로 정의한다. 따라서 하나의 썸은 다시 레이블

들의 연속으로 정의 될 수 있다. 어떠한 썸에 속하여 있는 레이블 들의 특성에 따라서 썸 들은 연결된 썸과 단절된 썸의 두 가지 타입으로 나뉠 수 있다. 연결된 썸이란 두개 또는 그 이상의 샷들을 포함하는 레이블을 하나 이상 가지고 있는 썸으로 정의며, 단절된 썸이란 두개 이상의 샷들을 포함하는 레이블이 하나도 없는 경우, 즉, 썸 안에 포함되어 있는 모든 레이블들이 하나의 샷만을 가지고 있는 썸으로 정의 된다. 연결된 썸들은 대화 장면에서 흔히 발견된다.

1. 실시간 장면 전환 검출

장면 전환 검출 엔진은 MPEG 으로 압축된 비디오 스트림을 입력으로 받아 압축도메인에서 실시간으로 장면 전환 지점을 검출하게 된다. 장면 전환 지점이 검출되면, 이를 기준으로 샷을 정의하게 되고, 정의된 샷은 대표 키프레임의 칼라 히스토그램 정보 등의 부가 정보와 함께 샷 정보로 정의되어 클러스터링 엔진에 제공 된다.

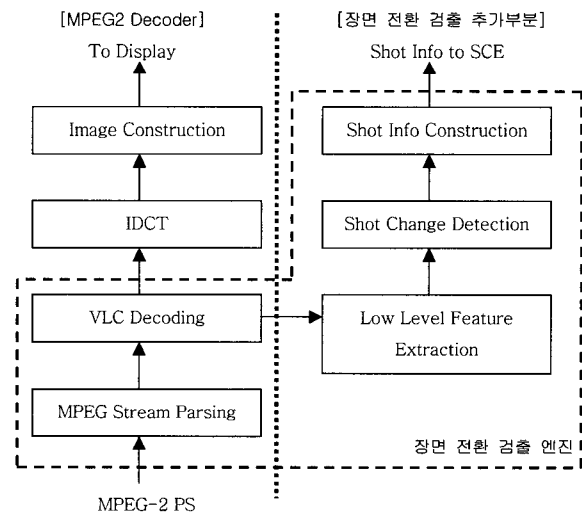


그림 2. 실시간 장면 전환 검출 엔진
Fig. 2. Block Diagram of Real-Time Shot Change Detection Engine

그림 2는 실시간 장면 전환 검출 엔진의 구조를 나타내고 있다. 디지털 방송에서 비디오 콘텐츠는 MPEG-2 스트림으로 전송되게 되며, 수신기는 튜너를 통하여 원하는 채널의 신호를 분리시키고 다시 분리된 채널에서 MPEG-2 디코더를 통하여 이미지를 구성하여 영상 디스플레이를 통하여 사용자에게 전달하게 된다. 이때 실시간 장면 전환 검출 엔진은 가변장 부호화(Variable Length Code)의 디코딩

된 신호를 디코더의 초기 단계에서 취득하여 장면 검출에 필요한 Low Level 특징소들을 실시간으로 추출하게 된다.

본 시스템에서는 장면 전환 검출을 위한 연산 복잡도를 최소화하고 실시간 처리를 위하여 필요로 하는 프로세서의 요구사항을 최소화하기 위하여, 추출/사용되는 특징소 들을 디코더의 초기단계에서 취득이 가능한 I-Type Picture의 DC 이미지의 color histogram과 Macroblock Type 정보로 제한하였다. 본 시스템은 일차적으로 I-Picture의 DC Color Histogram의 차이를 통하여 장면전환 후보를 검출하며, 비교 대상 I-Picture 사이에 존재하는 이미지들의 Macroblock Type 정보만을 버퍼에 저장하게 된다. 비교되는 두 I-Picture의 DC Color Histogram의 차이가 임계치 이하인 경우 버퍼에 존재하는 두 Picture 사이에 존재하는 이미지들의 Macroblock Type 정보는 폐기되며, 임계치 이상인 경우 버퍼에 존재하는 정보를 이용하여 장면전환 지점을 검출하게 된다^[10]. 따라서 본 시스템은 최대한 GOP의 지연시간이내에서 제한된 메모리만을 이용하여 각 프레임간의 장면 전환 여부를 검사하며 장면 전환 검출이 필요한 시점에서 한 GOP 이내의 정보만을 사용한다. 장면 전환이 검출 되는 경우 샷 정보, 즉 검출된 장면의 시작 정보, 끝 정보, 장면 대표 이미지, 장면 Color Histogram을 구성하여 샷 클러스터링 엔진에 제공하게 된다.

2. 실시간 샷 클러스터링

샷 클러스터링은 진술한 장면 전환 검출 엔진의 출력으로 샷 정보를 받아 제한된 수의 샷 정보만을 유지하며, 실시간으로 씬 검출을 하게 된다. 그림 3에 보인 바와 같이, 제안된 클러스터링 알고리즘은 샷메모리라고 표현된 제한된 수의 샷정보만을 유지함으로써 제한된 메모리만을 사용하도록 되어 있다.

그림 3에서 Shot_k 는 장면 전환 검출 엔진에서 k번째로 검출한 샷을 나타낸다. 그림 3에서 k+6 번째로 새로운 샷 (Shot_{cur})이 검출되어 클러스터링 엔진으로 입력되면 CLMS (Check Link and Merge Scene) 과정이 수행된다. Check Link 과정에서는 새로 입력된 샷과 샷메모리의 첫 번째 샷을 제외한 다른 샷들, 즉 매치윈도우 내에 있는 샷들과 샷정보, 특히 장면 전환 검출엔진에서 제공된 장면 Color Histogram을 비교한다. 이때 매치윈도우 내에 있는 어떤 샷 (Shot_n 이라 하자)과 Shot_{cur}의 장면 Color Histogram의 차가 임계치 이하인 경우 동일한 레이블에 속하게 되며, 이 경우 Merge Scene 과정이 수행되게 된다. Merge Scene 과정에

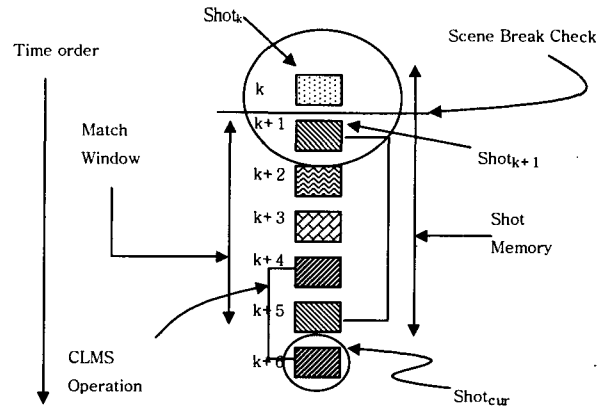


그림 3. 샷 클러스터링 과정
Fig. 3. Shot Clustering Process

서는 Shot_n과 Shot_{cur} 사이의 모든 샷들을 동일한 씬으로 등록하게 되고, 따라서 Shot_n과 Shot_{cur} 사이의 샷들이 속해있는 모든 레이블들이 하나의 씬으로 등록되게 된다. CLMS 과정 후 SBC (Scene Break Check) 과정이 수행되게 되는데, SBC 과정에서는 Shot_k와 Shot_{k+1}사이, 즉 샷메모리내의 첫 번째 샷과 두 번째 샷 사이에 씬의 경계가 발생하는 지 확인하게 된다. 샷 경계의 결정은 경계 후보인 두개의 샷, 즉 Shot_k와 Shot_{k+1}이 각각 연결된 씬에 속하는지, 단절된 씬에 속하는지에 따라 달라지게 된다.

표 1. 샷 경계 판단 방법
Table 1. Scene Boundary Decision Process

Shot _k 소속 씬 타입	Shot _{k+1} 소속 씬 타입	동작	
단절	단절	Shot _{k+1} 을 Shot _k 가 속한 씬에 통합한다	
단절	연결	Shot _k 과 Shot _{k+1} 사이의 씬 경계를 선언하고, Shot _k 가 속한 씬 구조 정보를 디스크에 쓴 후, 관련된 씬 정보를 메모리에서 삭제한다.	
연결	단절	Shot _k 과 Shot _{k+1} 사이의 씬 경계를 선언하고, Shot _k 가 속한 씬 구조 정보를 디스크에 쓴 후, 관련된 씬 정보를 메모리에서 삭제한다.	
연결	연결	동일 씬	다음 샷 경계로 이동
		다른 씬	다른 씬 Shot _k 과 Shot _{k+1} 사이의 씬 경계를 선언하고, Shot _k 가 속한 씬 구조 정보를 디스크에 쓴 후, 관련된 씬 정보를 메모리에서 삭제한다.

이때 판단 방법은 표 1에 상세히 나타나 있다. 이렇게 SBC 과정까지 수행한 후, 클러스터링 엔진은 장면 전환 검출 엔진으로부터 다음 샷 정보가 입력 될 때까지 대기하며, 새로운 샷 정보가 입력되면 진술한 CLMS 과정부터 다시 수행하게 된다. 이러한 과정을 통하여 샷 클러스터링은 최

대 샷 메모리내의 샷의 개수의 지연 시간을 갖으며 수행된다. 즉 썸 검출이 필요한 시점에 제한된 시간 내의 미래의 정보를 사용하게 된다. 그러나 사용자의 입력으로 샷 정보가 필요한 경우 샷 메모리내의 샷들을 모두 처리하여 지연 없는 썸 정보를 제공하게 되어, 사용자 측면에서는 지연 시간 없는 실시간 정보를 제공 받게 된다.

IV. 구조기반 건너뛰기

구조기반 건너뛰기(Structure based Skipping) 기능은 사용자가 앞쪽이나 뒤쪽으로 건너 뛰고자 할때, 많은 경우 이동하는 이야기 단위나 샷의 처음 시작 장면으로 이동하고자 한다는 가정을 기반 한 것이다. 종래의 시간 왜곡이 가능한 셋탑박스 등에서 사용되는 단순한 시간 기반의 건너뛰기 기능을 이용하는 경우, 원하는 지점으로 이동하기 위하여 앞 또는 뒤로 건너뛰기를 여러번 반복하여도 결국 원하는 위치로 이동하지 못하는 경우가 발생하게 된다.

본 논문에서는, 앞쪽으로 건너뛰기는 뒤쪽으로 건너뛰기와 유사한 개념이며, 뒤쪽으로 건너뛰기에 기반하여 쉽게 확장이 가능한 개념이므로, 뒤쪽으로 건너뛰기를 기반으로 설명하도록 하겠다.

그림 4은 구조기반 건너뛰기의 개념을 나타내고 있다. 현재 시청중인 위치를 i 번째 썸의 j 번째 샷의 중간이라고 가정하자. 이 경우 사용자는 샷 또는 썸 기반의 건너뛰기를 이용하여 현재 샷의 시작 지점인 'c'나, 이전 샷의 시작 지점인 'd'나, 현재 썸의 시작 지점인 'a'나, 이전 썸의 시작 지점인 'b'로 한번 또는 두 번의 리모콘의 버튼 동작과 같은 명령으로 쉽게 이동할 수 있다. 이러한 구조 기반의 건너뛰기가 제공되지 않는 경우, 사용자는 시간 기반의 순간 재생이나 앞으로 건너뛰기 기능을 이용하여 몇 번씩 명령을 내리고, 원하는 지점을 지나쳐 이동한 후 정속 재생을 이용하여 원하는 지점으로 이동될 때까지 기다려야 한다.

구조기반 건너뛰기의 실용적인 이용 방법은 그림 5에 보인 뉴스 브라우징의 예를 들 수 있다. 일반적으로 뉴스는 앵커 샷 (또는 앵커 썸)과 기사 설명 썸 (에피소드 썸)으로

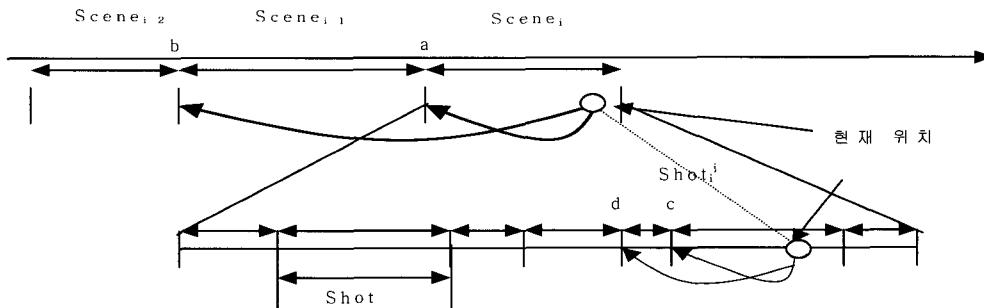


그림 4. 구조 기반의 뒤로 건너뛰기
Fig. 4. Structure-Based Backward Skipping

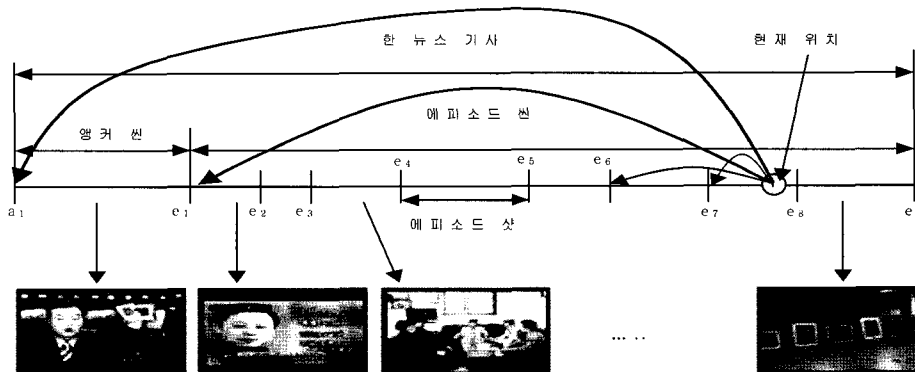


그림 5. 구조 기반 건너뛰기를 이용한 뉴스 시청 방법
Fig. 5. Example of News Browsing Method Using Structure-Based Skipping

로 구성되어 있다. 에피소드 썸은 다시 여러 개의 에피소드 샷들로 나누어진다. 뉴스 기사에 대하여 간략하게 설명하여 주는 앵커샷을 놓쳤을 경우, 사용자는 샷기반의 뒤로 건너뛰기를 이용하여 앵커샷의 처음으로 돌아갈 수 있으며, 현재의 뉴스 기사에 관심이 없는 경우 썸 기반의 앞으로 건너뛰기 기능을 이용하여 다음 뉴스 기사의 처음으로 갈 수 있다. 이러한 기능들은 뉴스뿐만 아니라 연속극이나 운동경기 중계와 같은 다른 장르의 비디오에서도 편리하게 사용될 수 있다.

V. 구조기반 요약보기

일반적인 기존의 디지털 비디오 시스템에서의 빨리보기(또는 배속보기)(앞으로/뒤로) 기능은 재생 프레임 레이트를 고정 시킨채 재생 배속에 비례하여 프레임들을 건너뛰는 방식으로 재생된다. 예를 들어 10배속 보기인 경우 초당 30프레임을 유지 하며 10 프레임 마다 한 프레임을 재생하는 방식으로 구현되게 된다. 이러한 배속보기에서는 일정한 비율을 유지하며 재생하는 프레임들을 건너뛰기 때문에 짧은 샷 또는 씬들을 놓치는 경우가 발생할 수도 있으며, 화면의 빠른 변화로 인하여 사용자가 화면에 집중할 수 없는 경우가 발생한다. 또한 이러한 배속보기에서는 적절한 오디오가 재생되지 않기 때문에 더욱 사용자가 내용을 이해하기 어렵게 된다. 따라서 기존의 배속보기 기능은 비디오의 요약이나 비선형 브라우징을 위한 도구로는 부적합하게 된다.

제안하는 구조기반 요약보기는 비디오의 일부와 이와 동

기화된 오디오를 재생하여, 원시적 형태의 요약을 제공한다. 이러한 구조기반 요약보기는 짧은 시간에 내용의 빠른 이해를 도와주며, 원하는 장면으로 빨리 이동할 수 있도록 하여 줌으로써, 배속 보기를 대체할 수 있는 기능이다. 이러한 구조기반 비디오 요약보기의 개념은 그림 6에 나타내었다. 그림 6에서 각각의 박스는 하나의 샷을 나타내며, 각 박스의 문자는 각 샷이 해당되는 레이블을 나타낸다. 각 박스 아래의 숫자는 각 샷이 나타난 시간적 순서를 나타낸다. 구조기반 비디오 요약보기에서는 전체의 일부만을 재생하여 재생 시간의 단축을 가져오는데, 재생할 구간의 선정 방법은 두 가지 단계를 거친다. 첫 번째 단계는 재생할 샷을 선정하는 단계이며, 두 번째 단계는 선정된 샷 내에서 재생할 일부 구간(세그먼트)을 선정하는 단계이다. 실시간 구조정보 추출 단계에서 샷, 레이블, 썸의 정보가 실시간으로 모두 추출되어 인덱스 저장장치에 저장되므로, 요약 보기를 위한 다양한 세그먼트의 실시간 선택이 가능하게 된다. 그림 6에서는 각 레이블에서 하나의 샷을 선택하여 각 선택된 샷의 뒤쪽 일부 세그먼트를 선택하여 재생하는 구조적 요약보기의 예를 보였다.

뉴스 프로그램의 경우는 다른 방법의 구조적 요약보기가 요구된다. 뉴스 프로그램은 앵커 샷과 에피소드 샷으로 구성되며, 앵커샷에서 해당 뉴스 기사의 요약정보를 제공하는 그 고유의 구조를 가지고 있기 때문에 앵커샷을 건너뛰는 것은 바람직하지 못하다. 실험에 의하면, 앵커샷의 시작부분의 일정한 양의 세그먼트와 앵커샷 이후에 나오는 일부 에피소드 샷의 시작 부분을 재생하는 것이 가장 효과적이었다. 제공하고자 하는 요약 정도에 따라 선택되는 에피소드 샷의 개수를 조정할 수도 있다.

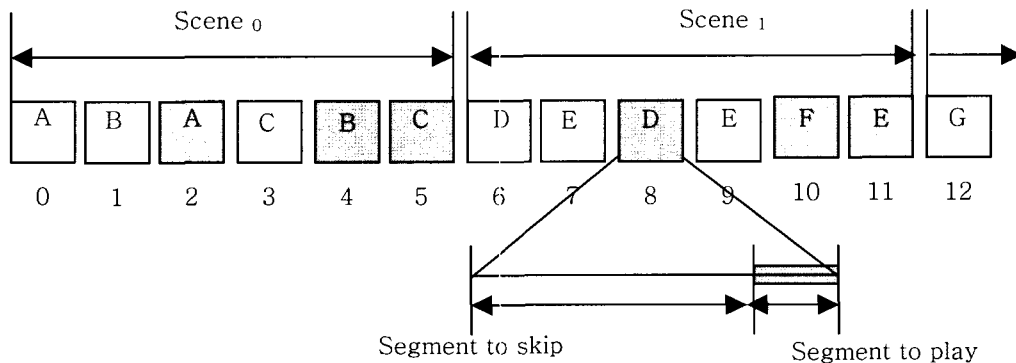


그림 6. 구조 기반 요약보기
Fig. 6. Structure-Based Video Skimming

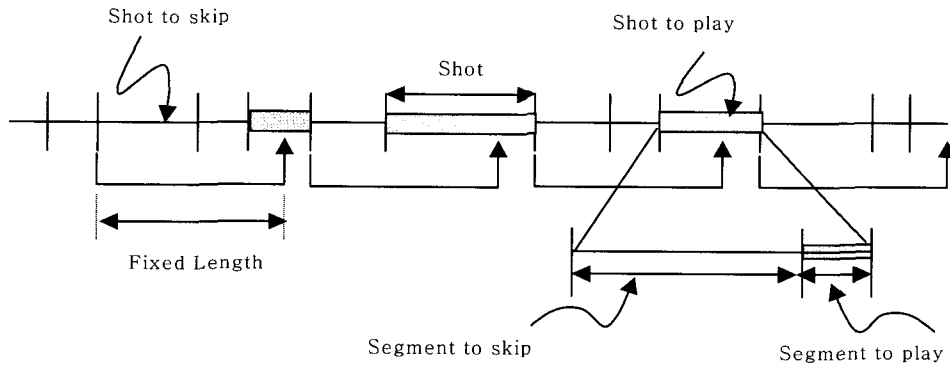


그림 7. 장르 독립적인 비디오 요약보기의 예
Fig. 7. Genre Independent Video Skimming

또 다른 변형 방법은 그림 7에 보인 장르에 독립적인 요약보기 방법이다. 장르에 독립적인 요약보기는 장르에 의존적인 썸 정보를 사용하지 않으며, 장르에 독립적인 샷 정보와 시간정보를 이용한다. 샷 선정 단계에서 시간 정보에 기반한 건너뛰기를 이용하여 선정된 샷 사이에 일정한 시간 이상의 간격이 유지되도록 하는 것이다. 본 시스템에서는 실험에 의하여 20초를 건너뛰어 샷을 선정한 후, 선정된 샷의 마지막 2초 분량을 재생하는 방법을 사용하였다. 이러한 장르 독립적인 요약 방법은 약 10 퍼센트로 전체 분량을 요약하는 효과를 얻을 수 있으며, 건너뛰는 시간 간격과 선정하는 세그먼트의 길이로 요약 정도를 조절할 수 있다. 즉 장르 독립적인 장면 외에 종래의 배속보기가 갖는 장점인 재생 시간의 예측이 어느 정도 가능하다는 장점을 갖는다.

VI. 실험결과

본 논문에서 제안된 비디오 인덱싱 엔진과 비선형 브라우징 기능은 다양한 비디오 콘텐츠를 이용하여 실험되었다. 이러한 실험 결과는 제안된 실시간 비디오 인덱싱 엔진이 제안된 비선형 브라우징 기능을 지원하기 위한 충분한 성능을 보여주고 있음을 나타내었으며, 제안된 비디오 인덱싱 엔진을 기반으로 구현된 비선형 브라우징 기능은 다양한 사용자들의 실험을 통하여 만족스러운 결과를 보여 주었다.

1. 실시간 장면 전환 검출 엔진

제안된 실시간 장면 전환 검출 엔진은 드라마, 영화, 뉴

스, 스포츠, 뮤직쇼 등 다양한 장르의 많은 양의 테스트 데이터를 이용하여 실험하였다. 다양한 장르에서 선택된 약 50시간이상의 비디오 데이터를 통하여 실험 한 바 92%의 Precision과 95%의 Recall의 성능을 보여 주었으며, 이러한 Precision/Recall은 장르와 상관없는 일관적인 성능으로 나타났다. 이러한 성능은 기존의 여러 연구결과가 보여주는 일반적인 장면 전환 검출 엔진들과 유사하다 할 수 있다.

2. 드라마 클러스터링 엔진

드라마 클러스터링 엔진은 상이한 패턴을 갖는 시트콤, 트레디드라마와 사극을 포함하는 다양한 종류의 약 150분 분량의 비디오 데이터에 대하여 실험 되었다. 드라마 클러스터링 엔진의 성능을 측정하기 위하여 수동적으로 Ground Truth가 선정되었으며, Ground Truth 선정과정에서 두 종류의 썸 경계가 나타났다. 하나는 Ground Truth 선정작업에 참여한 사람 모두가 동의하는 썸 경계이며, 다른 하나는 의견의 일치를 볼 수 없었던 썸 경계이다. 이러한 의견의 일치를 볼 수 없는 썸 경계는 트리 구조를 갖는 드라마의 구조적 특징 때문인 것으로 판단되었는데, Ground Truth 선정작업에 참여한 사람들은 다단계 구조를 갖는 드라마를 단지 두 단계로 구조화 하는데 서로 다른 기준을 갖고 작업을 행한 것이다. 성능의 분석 작업을 단순화하기 위하여 우리는 이러한 두 가지 서로 다른 썸 경계에 차이를 두지 않았으며, Open Problem으로 정의되어 있는 Descriptive Shot^[4]부분에 대하여 비선형 브라우징에 큰 영향을 미치지 않는 범위인 최대 2개의 샷 내의 썸 클러스터링을 허용하였다. 아래 실험 결과는 평균 샷 오프셋

과 함께 제안된 드라마 클러스터링 엔진의 정확도를 보여 준다.

표 2. 드라마 클러스터링 엔진 실험 결과

Table 2. Experimental Result of Drama Clustering

	길이 (분)	샷 개수	정의된 씬 경계 수	Precision	Recall	평균 오프셋
1	20	225	19	0.90	0.89	0.76
2	26	400	37	0.96	0.81	0.77
3	61	508	54	0.95	0.89	0.56
4	53	650	45	0.88	0.78	9.57

위 실험의 첫 번째와 두 번째 결과는 서로 다른 시트콤 드라마에 대한 결과이며, 세 번째는 사극, 네 번째는 영화에 대한 결과이다. 이 실험에서 샷 메모리는 6개로 제한되었다. 위 표에서 볼 수 있듯이 제안된 드라마 클러스터링 엔진은 대상 콘텐츠에 무관하게 안정적인 성능을 보여 주었다.

Ⅶ. 결 론

본 논문에서는 텔레비전과 같은 환경에서 쉽게 적용될 수 있으며, 종래의 아날로그 비디오 레코더 환경에 익숙한 사용자들도 쉽게 사용할 수 있는 두 가지의 비선형적 비디오 브라우징 도구들을 소개하였으며, 이러한 비선형적 비디오 브라우징 도구 들을 이용한 사용 모델을 제시하였다. 또한 셋탑박스과 같은 실시간 처리 환경에서 제한된 메모리를 사용하여 구현될 수 있는 클러스터링 알고리즘을 포함한 비디오 구조 정보 추출 방법을 제안하였다.

현재의 샷 클러스터링 알고리즘과 이를 이용하여 추출된 씬 정보는 현재 안정된 성능을 보이고 있으나 여전히 개선의 여지가 있으며, 일반적으로 사용되기 위하여는 모든

장르를 처리할 수 있도록 다양한 장르별 알고리즘 들이 개발 되어야 한다. 또한 제안된 비선형적 비디오 브라우징 도구들의 사용 모델은 여러 가지 변형이 가능하므로, 최적의 비선형적 비디오 브라우징 모델의 확립을 위하여 많은 사용자의 연구가 동반되어야 할 것이다.

참 고 문 헌

- [1] J. Meng, Y. Juan, and S. F. Chang, "Scene Change Detection in a MPEG Compressed Video Sequence," *Proceeding of SPIE* Vol. 2419, pp. 14-25, 1995
- [2] Y. Rui, T. S. Hwang, and S. Mehrotra, "Constructing Table-of-Content for Videos," *Multimedia Systems*, Vol. 7, No. 5, Springer-Verlag, pp. 359-368, 1999
- [3] M. M. Yeung and B. L. Yeo, "Video Content Characterization and Compaction for Digital Library Applications," *Proceedings of SPIE '97 Storage and Retrieval of Image and Video Database V*, 1997
- [4] A. Hanjalic, R. Lagendijk, and J. Biemond, "Automated Segmentation of Movies into Logical Story Units," *IEEE Transactions on Circuits and Systems for Video Technology*, 1996
- [5] H. J. Zhang, S. Y. Tan, S. W. Smoliar, and G. Yihong, "Automatic Parsing and Indexing of News Video," *Multimedia Systems*, pp. 256-266, 1995
- [6] Y. Rui, A. Gupta, and A. Acero, "Automatically Extracting Highlights for TV Baseball Programs," *Proceedings of ACM Multimedia 2000*, pp. 105-115, Oct., 2000
- [7] Tivo Inc., <http://www.tivo.com>
- [8] ReplayTV, <http://replaytv.com>
- [9] E. Chamhi, and L. S. Kamhi, "Simultaneous Recording and Playback Apparatus," *United States Patent* No. 5930444, 1994
- [10] S. B. Jun, K. Yoon, and H. Y. Lee, "Dissolve Transition Detection Algorithm Using Spatio-Temporal Distribution of MPEG Macro-Block Types", *Proceedings of ACM Multimedia 2000*, pp. 391-394, Oct., 2000

저 자 소 개



윤 경 로

- 1987년 : 연세대학교 전자전산공학과 학사
- 1989년 : 미시간대학교 전기전산공학과 석사
- 1999년 : 시라큐스대학교 전산학과 박사
- 1999년~현재 : LG전자기술원 정보기술연구소 책임연구원
- 주관심분야 : 멀티미디어 정보처리



전 성 배

- 1995년 : 연세대학교 컴퓨터과학과 학사
- 1997년 : 연세대학교 컴퓨터과학과 석사
- 현재 : LG전자기술원 선임연구원
- 주관심분야 : 비디오인덱싱, PVR