

장면전환 지점 검출을 이용한 동영상 검색 시스템 설계에 관한 연구

임영숙, 김형균*, 정기봉*, 고석만*, 오무송*
조선대학교 컴퓨터공학과

A Study On The Design Of Video Retrieving System Using Cut Detection

Young Sook Lim, Hyeng Gyun Kim*, Ki Bong Joung*, SukMan Go*, Moo Song Oh*
Dept. of Computer Eng. Chosun Univ.*

요 약

멀티미디어 기술이 발전함에 따라 방대한 양의 동영상 데이터로부터 원하는 장면을 빠르고 손쉽게 검색하기 위한 연구는 동영상을 포함한 멀티미디어 서비스가 제공되는 현 시점에서 대단히 시급한 문제로 인식되고 있다. 본 연구에서는 동영상 데이터를 대상으로 장면전환 지점을 검출하기 위하여 검색을 원하는 영상을 4개의 구역으로 구분하여 각각의 구역에 경계위치를 설정하며, 설정된 경계위치에 해당하는 칼라값을 분석하여 비디오 메모리에 저장한 후 이미지 데이터 베이스에 저장된 동영상 데이터의 각 Frame도 공간별 칼라값을 추출하여 이를 비디오 메모리 값과 비교해서 연속된 두 Frame 간의 칼라값 차이를 구하여 그 차이가 임계값 이상이 되면 장면전환 지점으로 검출하는 방법으로 사용자가 원하는 정보를 빠르게 검색하는 시스템을 설계하였다.

1. 서 론

최근 멀티미디어 데이터를 여러 분야의 컴퓨터 응용에 사용하려는 연구와 개발이 매우 활발한 추세이다. 멀티미디어 기술이 발전됨에 따라 방대한 양의 데이터로부터 원하는 장면을 빠르고 손쉽게 검색하기 위한 연구는 동영상을 포함한 멀티미디어 서비스가 제공되는 현 시점에서 대단히 시급한 문제로 인식되고 있다. 그러나 기존의 시스템에서는 주석 및 그 텍스트 정보에 의한 검색은 수작업에 의한 주석 작업의 부담이 과중하고 영상이 가지는 애매모호성으로 인하여 텍스트 정보만으로는 동영상 데이터가 가지는 정보를 충분히 표현하기 어려워 사용자의 필요에 유연성 있게 대응할 수 없다는 한계를 가지고 있다.

이와 같은 문제점을 해결하기 위하여 동영상이 가지는 내용정보 즉, 영상 자체로부터 추출된 정보나 움직임에 관한 정보등을 이용하여 검색함으로써 사용자가 검색하고자 하는 대상에 대한 자신의 의도를 보다 자유롭게 표현할 수 있는 연구가 필요하다.

그러나 영상 자체로부터 정보를 추출하고자 할 경우, 정지영상의 경우와 달리 동영상은 유사한 영상이 연속적으로 나타나는 특성을 가지고 있다. 따라서 영상에 관련된 정보를 그 각각에 대해 추출한다는 것은 처리시간과 저장용량의 낭비에 문제가 있다.

시간적인 배열로 구성되어 있는 비디오 샷들에서 장면변화를 검출한 뒤 관련성 있는 정보들을 글러스트링 하기 위해 본 연구에서는 동영상 데이터를 대

상으로 장면전환 지점을 검출하기 위하여 먼저 검색을 원하는 정지영상을 4개의 영역으로 구분하며, 구분된 각각의 영역에 경계위치를 설정하고, 설정된 경계위치의 해당하는 칼라값을 분석하여 비디오 메모리에 저장한 후 이미지 데이터 베이스에 저장된 동영상 데이터의 각 Frame도 공간별 칼라값을 추출하여 이를 비디오 메모리 값과 비교해서 연속된 두 Frame 간의 칼라값 차이를 구하여 그 차이가 임계값 이상이 되면 장면전환 지점으로 검출하는 방법으로 사용자가 원하는 정보를 빠르게 검색하는 시스템을 설계하였다.

2. 관련연구

2.1. 동영상 데이터의 특성

동영상 자료를 저장하고 검색하는 방법은 내용을 기초로 한 검색과 동영상에 대한 캡션(주석)을 이용한 검색으로 나눌 수 있다. 내용을 기초로한 동영상 정보 검색은 동영상 각각의 Frame에서 모양이나 색깔, 움직임등을 이용하여 동영상에 대한 의미 정보를 추출하고 이를 기반으로 영상 정보를 검색하는 방법이다. 캡션(주석)을 기초로 한 동영상 정보 검색은 동영상에 대한 캡션 정보 및 그림 설명 정보를 주석 형태로 삽입하여 이를 기반으로 영상 정보를 검색하는 방법이다. 최근 MPEG으로 압축된 동영상 자료에 대한 효율적인 인덱스 생성 기술에 대해 활발한 연구가 되고 있으며 이들은 주로 압축된 동영상에서 내용을 분석하여 지동으로 키 Frame을 선택하여 인

덱스를 구성하는 방법이다

2.2. 장면전환 지정 검색 방법

장면전환 지정을 검출하기 위한 방법으로는 히스토그램의 차이를 비교하는 방법, 장면을 구성하는 화소간의 차이를 비교하는 방법, 움직임 벡터의 비교방법등의 알고리즘등이 있다 [1,2,3]

3. 장면전환 지정 검출을 위한 알고리즘

본 연구에서는 칼라값에 기반한 두 영상의 유사성을 측정하기 위해 동영상의 Frame 들을 한 Frame 씩 불러 영상을 4 개의 영역으로 구분하며, 구분된 각각의 구역에 경계위치를 설정하고 여기에 해당하는 칼라값들을 비디오 메모리에 저장한 후 각각 연속된 두 Frame 간의 칼라값 차이를 구하여 그 차이가 임계값 이상이 되면 컷 지점으로 검출하게 된다

동영상에서 장면전환 지정을 검출하고 컷 단위로 분할하였다. 분할된 컷 들은 시작 Frame 번호와 끝 Frame 번호로 컷의 시작과 끝을 알 수 있다 분할된 컷은 검색시에 컷 단위 검색과 임의의 Frame 이 어떤 컷에 존재하는가를 알기 위하여 사용된다

3.1 장면전환 지정 검출을 위한 동영상 비교 검색

동영상 화일자료의 Pixel 별 비교를 위하여 해당 Frame 의 Pixel 위치를 다음과 같이 구한다

Frame 의 X 축 값인 mmpInfo dwMoviesExtentX 와 Frame 의 Y 축값인 mmpInfo dwMoviesExtentY 와 픽셀당 비트수인 mmpInfo dwPixelDepath 의 곱에 의하여 위치를 구한다 해당 Frame 으로부터 구해진 각각의 값의 연산 처리와 각 함수간의 전송을 위하여 Comp_Pixel_Dat 라는 구조체형으로 Define 하였으며, 검색화상과 비교 검색된 자료를 보관하기 위하여 Comp_Frame_Data 라는 구조체형을 Define 하였다

Comp_Frame_Data 의 구조체 내에는 해당 Frame 의 번호인 Frame_Position 와 첫째, 중간, 마지막 자료 비교대상의 평균값을 보관하기 위하여 Comp_Value 인 요소로 구성 되어 있고, 작업 대상이 되는 Class 의 생성자 부분에 비교 대상이 되는 동영상 화일에 Frame 수 만큼에 해당하는 비교 자료를 보관 하기 위한 Map 이 필요 함에 따라 아래와 같이 선언 하였다

```
Comp_Frame_Data
Comp_Data_Map[mmpInfo dwTotalFrames],
위의 항목에서 mmpInfo.dwTotalFrames 는 동영상 화일의 총 Frame 수. 또한 자료 비교를 위한 Pixel 의 자료 교환을 위한 구조체를 아래와 같이 선언 하였다
```

```
Comp_Pixel_Dat Comp_Position_Start,
Frame 의 진행에 따라 비교 자료가 저장 되는 위치의 Counter 를 0 으로 Clear 하였으며, 함수 LoadingInformation (MMPID idMovie)는 검색 대상이 되는 화일의 id 를 받아 기본 정보를 취득한 후 검색 처리를 위하여 설정 되는 함수 이다
```

기본 자료의 취득 후 픽셀당 비트수와 Frame 의 X 축 픽셀수, Frame 의 Y 축 픽셀수를 검색 위치를 처리

를 위한 구조체에 각각 설정 하였다

3-2 장면전환 지정 검출을 위한 동영상 비교검색 프로그램

위에서 정의된 내용에 의하여 구현된 비교, 검색 프로그램은 다음과 같다

```
void LoadingInformation(MMPID idMovie)
{
    MMPMOVIEINFO mmpinfo,
    mmpinfo chFullMacName[0] = 0,
    mmpGetMovieInfo(idMovie, &mmpInfo),
    // 취득된 픽셀당 비트수를 저장
    Comp_Position_Start PixelD= mmpinfo dwPixelDepath,
    // 취득된 Frame 의 x 축 픽셀수 저장
    Comp_Position_Start MoviesExtentX=mmpinfo dwMoviesExtentX,
    // 취득된 Frame 의 Y 축 픽셀수 저장
    Comp_Position_Start MoviesExtentY=mmpinfo dwMoviesExtentY,
}
AVIFrameComp..CompStram(HPALETTE hhClipPalercic)
{
    HPALETTE hpalette,
    DWORD CompValue_V1,CompValue_V2,
    CompValue_V3
    // AVI 화일의 해당 Frame 의 핸들을 가져 온다
    hpalette = mmpGetPalettehandle(idMovie),
    if(hpalette == NULL)
    {
        // Frame 의 핸들을 가져 오지 못하면 Error Message 를 보여 준다
        MessageBoxWnd,
        'Could Not Access Palette Information' ,
        szAppuame, MB_OK,
        return,
    }
    // 두 화상의 비교 시작점을 0 으로 설정 한다
    Comp_Position_Start Start_Position = 0,
    // PixelComp 함수에 Load 된 Frame 의 핸들 과 비교 대상인 화상
    // 핸들, 비교 시작 지점의 Y 축 값을 보낸다
    // Y 축값*X 축값*해당픽셀당 비트수=검색 시작 위치값
    CompValue_V1 = PixelComp(hpalette, hhClipPalercic,
    Comp_Position_Start),
    Comp_Position_Start Start_Position
    =(int)(mmpInfo DwMovieExtentY/ 2),
    CompValue_V2 = PixelComp(hpalette, hhClipPalercic,
    Comp_Position_Start),
    Comp_Position_Start Start_Position=mmpInfo dwMovieExtentY,
    CompValue_V3 = PixelComp(hpalette, hhClipPalercic,
    Comp_Position_Start),
    CompValue_Ave =
    (CompValue_V1 + CompValue_V2+CompValue_V3) / 3,
    Comp_Data_Map[Map_Count].Frame_Position =
    mmpGetCurFrame(idMovie),
    Comp_Data_Map[Map_Count++] Comp_Value= CompValue_Ave,
}
```

4. 동영상 검색 시스템 설계 및 구현

본 시스템의 실험은 IBM-PC(P-200MHz) Win95 환경에서 Microsoft Visual C++(version 5.0){6,7,8,9,10,11}를

사용하여 구현하고 비디오 보드는 DVD 용 멀티미디어 통합 VGA 보드 Winx perfect IV를 사용하여 본 논문에서 제안한 프로그램을 구동하였다.

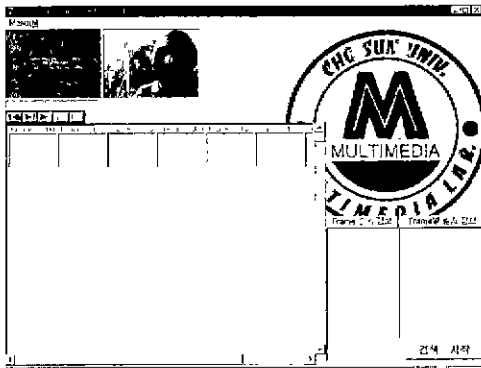


그림 1 동영상 검색시스템 시작화면

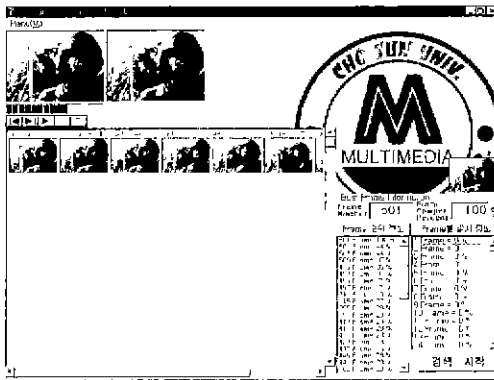


그림 2 검색 결과 화면

동영상 검색이 끝나면 Map 에 저장된 유사비에 따라 Sort 되어 [그림 3] 과 같이 Frame 정보를 출력하도록 하였으며, 여기에 나타나는 Frame 중 특정 Frame 을 선택하면 바로 해당 Frame 으르 이동 할 수 있게 하였다

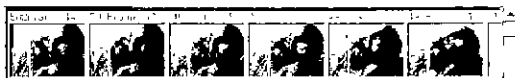


그림 3 유사 Frame 리스트

유사 Frame 중 가장 유사비가 높은 Frame 정보를 [그림 4] 와 같이 화면우측 상단에 표시하도록 하였다

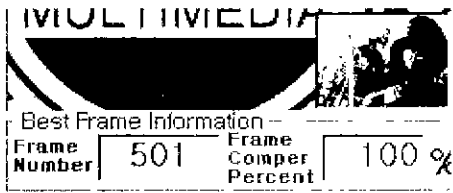


그림 4 최적 일치 Frame 정보 화면

5. 결론 및 향후과제

화면전환이 일어나는 지점 즉, 컷을 검출하기 위한 연구는 오래 전부터 수행되어 왔으며 뉴스나 드라마와 같이 장면의 구분이 비교적 확실한 데이터를 대상으로 실험할 경우 대단히 높은 컷 검출율을 보인다

본 논문에서는 매 Frame 마다 검색을 원하는 정지영상을 4 개의 영역으로 구분하며, 구분된 각각의 구역에 경계위치를 설정하고 그 위치별 칼라값을 추출하여 연속된 두 Frame 간의 칼라값 차이를 구하여 그 차이가 임계값 이상이 되면 장면전환지점으로 검출하는 방법으로 장면전환을 기반으로 사용자가 컷을 중심으로 원하는 정보를 신속히 검색할 수 있는 동영상 시스템을 설계구현 하였다 데이터 베이스 구축에 있어서 데이터들에 대한 스크립터를 정의하는데 정보를 기술할 수 있는 기반기술로써 장면 전환의 검출은 중요하게 이용 될 수 있다 향후 장면전환 뿐 아니라 영상을 구성하는 객체들의 정보를 추출하여 객체 각각에 대해서 인덱싱을 하기 위한 방법에 대한 연구와 추출된 정보를 이용하여 효율적인 데이터 베이스의 접근과 관리를 위한 인덱싱에 관한 연구가 진행되어야 할 것으로 본다

[참고 문헌]

- [1] Rajiv Mehrotra, James E Gray, "FeatureBased Retrieval of Similar Shapes", 9th International Conference on Data Engineering, pp 108-115, 1993
- [2] W Niblack et al The Qbic project Querying Images, by content using color, texture, and shape In SPIE 1908, Storage and Retrieval for Image and Video Databases, Feb 1993
- [3] William I, Grosky, Peter Neo, Rajiv Mehrotra, "A Pictorial Index Mechanism for Model Based Matching", Data and Knowledge Engineering, vol 8, pp 309-327, 1992
- [4] A Akutsu et al., ' Video Indexing Using Motion Vector', Proc SPIE Visual Communication and Image Processing 92, Vol 1818, pp 1522-1530, 1992
- [5] gutrukh Ahangei, Thomas D C Little, "A Survey of Technologies for Parsing and Indexing Digital Video" Journal of Visual Communication and Image Representation. San Jose, February, 1994
- [6] 이상연, "Visual C++ Programming Bible Ver 5 X", 영진출판사, 1997
- [7] 이정근 "Microsoft Multimedia Development Kit", 도서출판 세운, 1996
- [8] Microsoft, "Microsoft Windows software Development Kit", Microsoft, 1997
- [9] Ioannis Pitas, Digital Image Processing Algorithms, Prentice Hall, 1993
- [10] Jon Peddie, "Multimedia Graphics Controllers", Windcrest, 1994
- [11] James D Murray & William Varryper, "Graphics file format second edition" O'Reilly & Associates Inc 1996