

동영상에서 배경프레임을 이용한 차량 프레임 검출

남 석 우[†] · 오 해 석^{††}

요 약

본 연구는 동영상으로부터 내용기반 검색을 위하여 동영상의 연속된 프레임간의 영상의 내용 변화를 찾아내어 프레임의 시간정보와 번호판 프레임 영상을 통하여 얻어진 정보를 데이터베이스화하여 검색하는 시스템을 제안한다. 얻어진 동영상을 배경프레임과 처리프레임의 비교영역의 영상의 특징정보를 비교하여 원하는 프레임을 찾는다. 차량의 통과 시간과 차량의 번호판프레임을 자동으로 추출하여 동영상을 내용과 함께 저장하여 원하는 차량의 동영상 부분을 보여주는 웹에서의 검색시스템이다. 이는 교통정보를 구축 동영상에 포함하고 있는 내용 즉 통과 차량의 정보를 제공할 수 있게 된다.

Car Frame Extraction using Background Frame in Video

Seok-Woo Nam[†] · Hea-Seok Oh^{††}

ABSTRACT

Recent years, as a rapid development of multimedia technology, video database system to retrieve video data efficiently seems to core technology in the in the oriented society. This thesis describes an efficient automatic frame detection and location method for content based retrieval of video. Frame extraction part is consist of incoming / outgoing car frame extraction and car number frame extraction stage. We gain start/end time of car video also car number frames. Frames are selected at fixed time interval from video and key frames are selected by color scale histogram and edge operation method. Car frame recognized can be searched by content based retrieval method.

키워드 : 내용기반검색(Content-based Retrieval), 영상처리(Image processing)

1. 서 론

컴퓨터 기술의 발달과 함께 변하는 다양한 매체의 출현으로 인하여 컴퓨터에 의해서 표현되고 처리되는 정보가 기존의 전통적인 문자정보 위주에서 정지영상, 동영상, 음성 데이터 등과 같이 보다 이해하기 쉬운 멀티미디어 데이터 세계로 변해가고 있다.

이러한 다양한 정보들을 처리하기 위한 여러 가지 연구들이 다양한 분야에서 연구 응용되고 있다. 방대한 양의 데이터를 저장·검색하기 위한 연구들이 활발히 이루어지고 있다.

본 연구에서는 멀티미디어 영상 중에서 동영상 데이터를 이용하여 도로상에서 주행하는 차량을 담아 내용기반에 의한 동영상을 분류하고 분류된 동영상을 웹상에서 검색하기 위한 시스템에 관한 연구이다.

내용기반 검색을 통하여 이루어지는 대상이 정지영상과 동영상에서 다루어지고 있는데 동영상에서 오디오 부분을

생략하면 정지영상처리와 유사한 관계가 있다. 동영상은 연속된 정지영상들의 모임이라고 보면 연속된 영상의 비교처리에 의하여 처리한다면 정지영상과 같이 처리할 수 있다.

영상정보를 효율적으로 질의·검색하기 위해선 문자기반 검색방법과 의미론적 기반검색방법, 내용기반 검색방법이 있다[1]. 이 중에서 내용기반에 관한 연구는 영상데이터의 색상(color), 모양(shape), 질감(texture) 등의 내용 표현요소들을 자동으로 추출하여, 이를 검색에 이용하는 방법이다[5].

내용기반 검색을 위하여 차량의 진입정보를 구하기 위하여 배경프레임과 획득프레임을 비교하여 동영상의 내용을 구한다. 2장에서는 내용기반 검색을 위한 차량의 배경프레임과 획득프레임을 비교하는 방법을 알아보고 3장에서는 진입차량의 번호판 프레임을 획득하여 저장하는 시스템에 관하여 제안하고 4장에서 구현 평가하고 5장에서 결론을 기술한다.

2. 관련 연구

다양한 분야에서의 동영상과 정지영상의 관한 연구가 이루어지고 있다.

* 이 논문은 2003년 숭실대학교 교내연구비에 의해 작성이 되었음.

† 정 회 원 : 해천대학 컴퓨터통신계열 교수

†† 종신회원 : 숭실대학교 컴퓨터학과 교수

논문접수 : 2002년 7월 30일, 심사완료 : 2003년 6월 18일

2.1 동영상 키프레임의 추출

동영상에서의 키프레임 추출은 동영상의 내용을 구분하는 중요한 정보임으로 매우 중요하다. 키프레임 즉 원하는 비디오의 샷을 구분하기 위한 방법으로는 연속된 프레임과 프레임을 비교하여 변화를 찾아 구분한다. 구분 방법으로는 픽셀간의 비교, 통계적 차이, 히스토그램, DCT 계수를 비교하는 방법, 에지를 추적하는 방법, 움직임을 이용하는 방법 등이 있다[2, 3, 6, 7]. 본 연구에서는 연속된 프레임이 아닌 배경프레임과 획득 프레임을 처리한다.

차량을 인지하기 위해서 사용하는 방식으로는 프레임간의 영상차를 이용하는 방법과 프레임간의 정보 없이 주어진 프레임만을 처리하는 방법이 있다[4].

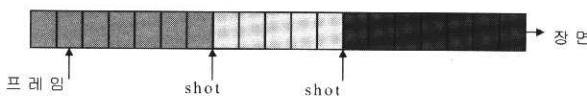
영상처리 기법을 이용하여 이동중인 차량을 추적하고 계수하는 시스템은 많은 분야에 응용될 수 있다. 기존의 차량 계수 시스템에서 사용한 방법들은 도로의 구조정보와 시간 흐름에 따라 변화하는 차량의 이동 속도를 시스템에 반영하지 않기 때문에 효율적인 차량 계수가 이루어지지 않고 있다. 차량의 이동상황을 추적하는 이동점 추적 방법에 기술하고, 추출된 도로 구조정보와 차량의 이동 속도 정보를 이용하여 가변적으로 계수 범위를 조정함으로써 불필요한 처리시간을 절약하는 효율적인 차량 계수 방법을 제안하였다[9].

색상변화 분석을 차량의 속도와 수량을 측정하고[10] 동영상이 압축상태에서의 처리는 주로 MPEG 동영상에서 전체를 복원하지 않고 일부 프레임만을 가지고 처리하고 있다[8].

2.2 동영상에서의 내용기반 검색

일반적으로 비디오는 프레임(frame), 샷(shot), 장면(scene)의 세 가지 구성 요소로 이루어진다. 프레임이란 영상을 분할하는 최소 단위로써, 필름 한 장에 해당하는 하나의 영상을 나타낸다. 샷은 연속된 프레임으로 구성되며, 하나의 카메라로 촬영한 영상들이다. 샷 내에서는 필름이 끊기지 않고 연속적으로 연결되어 있으며 일반적으로 샷은 장면 전환 검출의 기본 단위로 사용된다.

장면은 일련의 샷으로 구성되며, 보통 하나의 대상에 초점을 맞추어 촬영된 영상들을 나타낸다. (그림 1)은 비디오 데이터의 구성을 보여준다. 각각의 직사각형은 하나의 프레임임을 보여주며 프레임들이 모여서 샷을 구성하고 샷이 모여서 장면을 이룬다.



(그림 1) 동영상(비디오) 데이터의 구성

영상의 전처리로 영상개선, 동영상에서의 프레임간의 변화를 비교하기 위해 히스토그램과 에지를 이용하게 되고, 일정한 영상의 형태를 갖추려고 영상의 정규화, 계산상의 복

합함을 줄임과 문자 인식을 영상의 양자화(흑,백), 그리고 원하는 영역을 얻기 위한 영역분할 등이 이용된다.

2.2.1 동영상 장면전환 검출 방법

동영상에서 장면전환 처리를 위하여 이전프레임과 현재의 프레임을 비교하여 변화된 프레임을 검출하고 있다. 다음에 설명하는 방법들은 장면 전환 검출 방법이지만 배경 화면 획득에도 응용된다.

① 히스토그램 비교에 의한 방법

두 프레임간의 히스토그램의 차이를 비교함으로써 카메라와 물체의 움직임을 감지할 수 있다. 사실 두 프레임간의 차이는 배경이 많은 부분을 차지하고 있고 움직임이 적기 때문에 그리 크지 않다. 프레임의 각 화소의 값을 얻어 히스토그램을 처리한다.

그러나 비교 영상이 서로 비슷한 특성을 가지고 있을 때는 구별하기가 힘들어 다음과 같은 에지 처리로 보완한다.

히스토그램의 처리는 칼라와 그레이 영상의 구분으로 나눌 수 있고, 칼라 공간은 RGB에서 HSI 변환하여 색상, 명도, 채도로 색을 정의하여, 직관적인 처리를 할 수 있다.

이 방법은 가장 일반적인 방법으로 연속되고 인접한 프레임간의 히스토그램 차이를 계산해 주어진 임계치와 비교해 임계치를 초과할 경우 샷으로 판단하는 방법이다. 히스토그램의 차를 이용한 방법은 카메라의 움직임에 강인하며, 연산속도가 빠르다는 장점이 있다.

$$\sum_{j=1}^G |H_i(j) - H_{i+1}(j)| > T \tag{1}$$

식 (1)의 G는 칼라 레벨의 수이며, H_i 은 i 번째 프레임의 히스토그램을 의미한다.

② 픽셀간의 차를 이용하는 방법

인접한 프레임에 대응하는 픽셀간 차를 계산함으로써 두 프레임간의 샷 전환을 감지하는 방법으로 임계치(Threshold)를 계산해 계산된 임계치를 초과하는 프레임을 컷(샷 경계)으로 판단하는 방법이다.

$$DP_i(x, y) = |F_i(x, y) - F_{i+1}(x, y)| \tag{2}$$

$$\frac{\sum_{x,y=1}^{X,Y} DP_i(x, y)}{X \times Y} > T \tag{3}$$

식 (2)에서 $F_i(x, y)$ 는 i 번째 프레임 (x, y) 의 밝기 값을 나타내며, $DP_i(x, y)$ 는 인접하는 프레임의 픽셀차를 의미한다. 식 (3)의 $X \times Y$ 는 프레임의 크기를 나타내며, T는 컷을 판별하기 위한 임계치이다. 이 방법은 구현이 아주 간단한 반면에 카메라 움직임에 민감하며, 영상 각각의 픽셀에 대해 비교를 해야하기 때문에 시간이 많이 걸린다는 단점이 있다.

③ 통계적인 차를 이용한 방법

인접하는 프레임간의 통계적인 차를 이용한다. 즉 두 프레임간의 평균, 분산, 표준편차 등을 구한 후 임계치를 설정해 샷을 검출하는 방법이다.

$$\frac{[(\frac{\sigma_i + \sigma_{i+1}}{2}) + (\frac{\mu_i - \mu_{i+1}}{2})^2]^2}{\sigma_i \times \sigma_{i+1}} > T \quad (4)$$

식 (4)의 σ_i 는 i 번째 프레임의 분산이며, μ_i 는 프레임의 평균을 의미한다.

④ 변환 계수의 차를 이용하는 방법

FFT, Wavelet Transform 그리고 DCT와 같은 변환영역의 계수들을 이용하여 샷 경계를 검출하는 방법들이다. 특히, 최근 들어 MPEG 동영상과 같은 압축된 영상에서 압축을 해제할 필요 없이 비디오 데이터로부터 DC계수를 사용하여 샷 경계를 검출하는 방법이 연구 중에 있다. 이 방법은 프레임을 다수의 서블록으로 나눠 각각의 DC계수를 추출해 추출된 DC계수들을 인접한 프레임과 비교함으로써 샷 경계를 검출하는 방법이다.

⑤ 에지의 차를 이용하는 방법

장면전환 검출을 위하여 주로 프레임이 가지는 색상정보를 이용하고 있으나 비슷한 값들로 인한 분별이 어려울 경우 프레임에 포함된 에지 들을 비교하여 구분할 수 있다. 가장 간단한 에지 모델 중 하나는 기울기(gradient) 계수이다. 디지털 영상을 표현하는 3차원 공간을 정의할 수 있다. 여기서 x 와 y 는 화소의 좌표이고, 광학 매개변수(optical parameter)는 밝기를 나타낸다. 그리고 기울기는 광학 매개변수에 의해 표현되는 스칼라 영역에서 가파른 상승을 나타낸다.

$$G = i \frac{\partial I}{\partial x} + j \frac{\partial I}{\partial y} \quad (6)$$

여기서 I 는 광학 파라미터이다. 기울기 벡터의 절댓값(또는 계수)는 다음과 같다.

$$|G| = \sqrt{[\frac{\partial I}{\partial x}]^2 + [\frac{\partial I}{\partial y}]^2} \quad (7)$$

기울기 계수는 광학 매개변수의 급격한 변화의 궤적을 따라 자신의 지역(local) maxima를 갖는다. 디지털 영상에서 Δx 와 Δy 요소들은 작은 윈도우 상에서 한정된 차 연산자를 통해 얻어지는데, 대부분 2×2 또는 3×3 화소를 이용한다.

윤곽선을 추출하기 위한 수학적 모델로 라플라시안(Laplacian) 연산자가 있다.

$$L = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \quad (8)$$

라플라시안은 스칼라이며 선형 연산자이다. 결과적으로, 이것은 디지털 영상에 대한 유한-차 회선(Convolution)에 의한 계산을 수행한다.

⑥ 임계치를 이용하는 방법(Thresholding)

영상 분할에 쓰이는 일반적인 기술이 바로 임계치(thresholding)이다. N 을 자연수 (x, y) 로 정의하자. (x, y) 는 디지털 영상의 공간좌표이고, $G = \{0, 1, \dots, l-1\}$ 은 그레이레벨을 나타내는 양의 정수이다. 그리고, 영상 함수는 다음과 같은 매핑으로 정의된다.

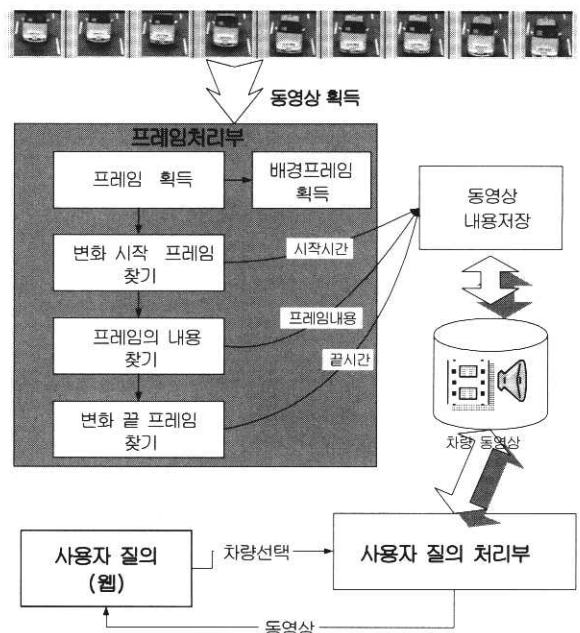
$$f : N \times N \rightarrow G$$

(x, y) 좌표상의 한 화소의 밝기값(예를 들어 graylevel) $f(x, y)$ 로 표시한다. $t \in G$ 은 임계치를, $b = \{b_0, b_1\}$ 은 이진 값의 쌍을 나타낸다. 그레이레벨 t 에서의 영상 함수에 대한 임계치의 결과는 $f_t : N \times N \rightarrow B$ 로 표시되는 이진 영상 함수이며, 정의는 다음과 같다.

$$f_t(x, y) = \begin{cases} b_0 & \text{if } f(x, y) < t \\ b_1 & \text{if } f(x, y) \geq t \end{cases} \quad (9)$$

3. 시스템의 구성

입력된 비디오의 프레임을 분석하여 비디오에 포함된 내용데이터(차량번호)와 시간 데이터를 찾아서 데이터베이스에 입력하고 웹상에서 원하는 데이터를 찾아볼 수 있는 내용기반 시스템이다. 시스템은 (그림 2)과 같이 프레임처리부, 사용자 질의 처리부로 구성되어진다.



(그림 2) 시스템 구성도

3.1 프레임 처리부

획득된 동영상상을 프레임 단위로 처리하게 되는데 우선 배경프레임을 추출하고 이 배경프레임과 연속되는 프레임들을 비교하여 차량진입/진출프레임을 찾아서 차량의 진입/진출시간을 얻는다. 또한 번호판이 들어있는 프레임을 찾아서 내용인 차량의 영상을 저장하게 된다. 이러한 정보는 질의를 할 때 사용된다.

3.1.1 프레임 획득

도로 또는 특정 구역의 진입로에 카메라를 설치하고 들어오는 차량의 영상이 담긴 동영상상을 획득·저장하여 프레임 단위로 처리하여 차량의 진입과 번호판이 들어있는 프레임 그리고 진출하는 프레임을 찾아서 프레임의 위치정보를 구한다.

실제로 촬영한 영상은 도로의 육교 위에서 달려오는 차량을 비디오카메라로 촬영하여 그 동영상상을 AVI 파일 형태로 저장하여 프레임을 획득하고 있다.



(그림 3) 동영상의 연속된 프레임

연속된 프레임들은 초당 30프레임으로 저장되어 있다. (그림 3)에서는 매 다섯번째 프레임을 획득한 연속된 프레임들의 일부분이다.

3.1.2 배경프레임의 획득

배경프레임의 초기획득은 동영상의 처음 시작부분의 연속적으로 변화가 없는 프레임을 배경프레임으로 결정하고, 배경프레임 찾기에서는 실제적으로 입력되는 영상이 환경에 따라 변화되고 있는 것에 대해 적응하기 위한 고려로, 시간, 날씨 주행도로의 여건 등의 변화에 적응하기 위한 것이다. 이를 위하여 처음에 사용한 배경프레임을 처리 중간 중간에 변화가 없는 프레임들의 연속을 이용하여 배경프레임을 교체하고 있다. 급격한 환경의 변화에는 아직 문제가 있다.

```
// 배경화면획득
// AreaA(비교영역1)와 AreaB(비교영역2)가 가장 많이 나오는 프레임을 배경 frame으로
// 프레임 획득(초당 처리 프레임을 capture), time stamp도 함께
frame = capture_frame();
// 배경 frame과 획득 frame 비교
thA = compare_AreaA();
thB = compare_AreaB();
// 진입차량 검사
// thH는 변화값기준 상위(다르다기준) thL는 변화값기준 하한(같다기준)
if (thA > thH) {
```

```
in_car = 1; 차량진입 OK
reset_countA = 0;
set_countA++; // 처음에만(1일때) 진입시작 나머지값은 진입 중
if (set_countA == 1) then { // 차량진입 정보 저장
    start_time = frame_time + alpha; // 시작 시간의 저장
    alpha는 여유시간
}
}
else if (thA < thL) { // 배경과 같아짐
in_car = 0;
reset_countA++; // 새로운 배경화면 획득을 위한 카운터
set_countA=0;
}
// 번호판 프레임 찾기
if (thB > thH) {
out_car = 1;
reset_countB = 0;
set_countB++; // 처음에만(1일때) 진출시작 나머지 값은
진출 중
// 번호판이 들어 있는 후보 frame
if (set_countB == 1) then { // 차량 번호인식 호출
if (search_plate(image frame) != 1) set_countB = 0;
// 찾지 못했다면
car_number ;
}
}
else if (thB < thL) { // 배경과 같아짐
out_car = 0;
reset_countB++; // 새로운 배경화면 획득을 위한 카운터
set_countB=0;
if (reset_countB == 1) { // 차량 진출 확인
end_time = frame_time - alpha; // alpha 여유시간
}
// 배경화면의 교체
if (reset_countA > 6000 and reset_countB > 6000)
{ //6000 f/10 min
back_image = frame; //배경화면을 현재의 frame으로 변경
}
}
```

3.1.3 프레임 찾기

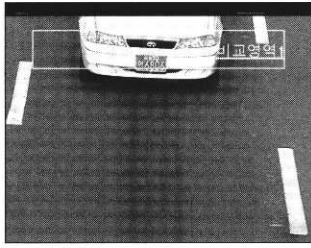
차량진입 프레임 찾기는 배경영역과 획득된 프레임의 비교영역 1부분을 처리하여 일정 임계값 이상일 경우 차량이 진입되었다고 판정한다. (그림 4)의 윗부분이 차량의 진입 여부를 조사하기 위한 영역이다.

비교영역을 비교하는 방법으로는 칼라 히스토그램과 에지를 찾아 윤곽선을 이용한 방법을 적용하고 있다. 히스토그램으로 처리시 운전석 부분이 통과할 때에 배경과 같은 결과가 나와서 이를 다시 에지를 이용해서 배경과 다시 비교하게 되었다.

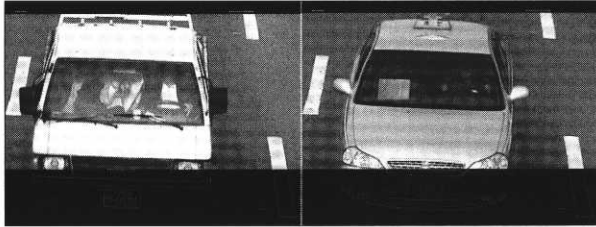
0	-1	0
-1	4	-1
0	-1	0

<표 1> 라플라시안 mask

비교영역을 히스토그램과 에지(라플라시안)에 의한 비교를 수행하여 일정한 값 이상의 변화가 있을 때 프레임을 획득하게 된다.



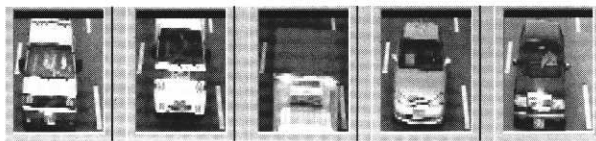
(그림 4) 차량진입 프레임 찾기 비교 영역



(그림 5) 라플라시안 비교 영역의 실행후의 프레임 들

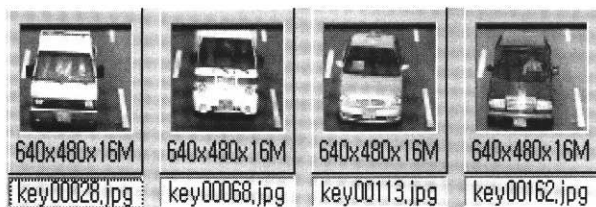
3.1.4 프레임의 내용 찾기

추출된 번호판이 들어있는 프레임은 동영상의 내용을 나타내기 위한 중요한 프레임이다. 물론 번호판을 인식하는 처리부분을 추가하여 내용을 차량번호로도 할 수 있으나 여기서는 번호판이 있는 프레임을 찾아 프레임을 저장하여 동영상과 함께 저장하게 된다. 우선 동영상으로부터 연속된 프레임을 각각의 정지영상으로 획득하여 처리하게 된다. 획득된 배경프레임과 획득한 프레임을 비교하여 원하는 프레임을 찾는 과정을 거친다.



(그림 6) 번호판이 있는 프레임 찾기 결과

(그림 6)의 3번째 프레임의 경우 히스토그램 처리만으로 잘못 인식된 프레임이다. 이를 보완하기 위하여 에지 연산을 수행함으로 (그림 7)과 같은 결과를 얻을 수 있다.



(그림 7) 번호판이 있는 프레임 찾기 옳은 결과

3.1.5 차량진출 프레임 찾기

차량진입 프레임찾기, 번호판 프레임찾기와 진출차량프레임 찾기는 동일한 처리과정을 거친다. 배경프레임과 획득한

프레임을 비교영역별로 비교하여 비교치가 일정한 값 이상의 변화이면 프레임을 결과로 얻을 수 있다.

처리의 결과로 차량진입과 진출부분에서는 프레임에 관한 시간 정보를 추출하게 된다.

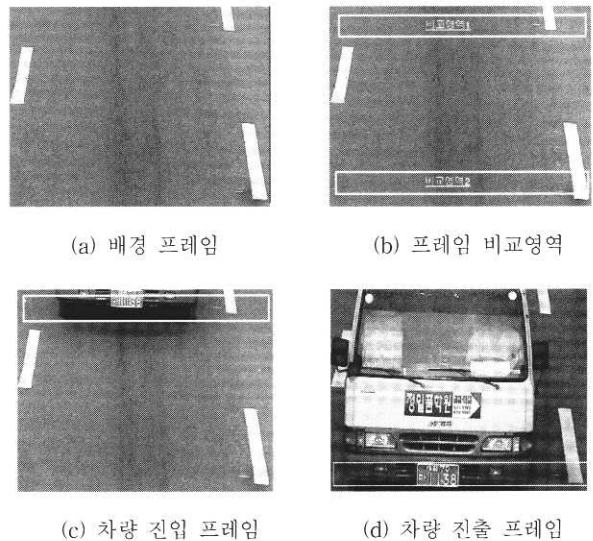
3.2 웹 질의부

처리동영상에서 처리된 결과로 나온 시간 정보와 번호판 프레임을 내용으로 저장하고 있다가 사용자가 원하는 프레임을 선택하게되면 그 프레임이 들어있는 부분을 찾아서 차량의 동영상을 보여주게 된다.

4. 구 현

구현환경은 자동차 동영상의 획득은 디지털비디오 카메라를 이용하여 일반 도로상에서 한차선에 대해 다가오는 차량을 얻었으며, 이를 AVI 파일 형태로 저장하여 실험하였다. 동영상내의 획득된 프레임의 크기는 640×480의 크기이며 초당 15개 이상의 프레임으로 구성되어 진다. 시스템은 PC에서 VC++로 프레임처리부와 구현하고, 얻어진 정보 시간정보와 번호판 프레임을 웹에 저장하여 웹부라우저로 검색할 수 있도록 하였다.

(그림 8)은 시스템 처리과정의 프레임을 보여 주고 있다.



(그림 8) 배경/진입/진출 프레임 들

번호판영역이 들어 있는 프레임을 키프레임으로 할 때 그 결과는 <표 2>와 같이 얻을 수 있었다.

<표 2> 번호판이 있는 프레임 획득 결과

방 법	전체 프레임수	번호판 프레임수	찾은 번호판 프레임수	시 간
히스토그램과 에지연산방법 이용	194	4	4	약7초
	511	9	9	약18초

5. 결 론

본 연구에서는 환경에 변화, 즉 동영상 획득의 위치, 시간, 장소 등에 비교적 영향을 덜 받는 시스템을 제안한다. 배경 프레임과 획득프레임의 비교에 의하여 진행되는 처리는 변화하는 배경프레임의 변경도 고려하였다. 배경프레임과 동영상의 프레임을 히스토그램과 에지연산을 수행하여 임계값이상으로 변화가 이루어지면 이를 결과로 인정하였다.

동영상에서 배경프레임 처리는 내용의 변화를 정확하게 검출하기 위하여 필요하고 배경프레임의 변화를 정확히 반영함으로써 동영상 내요 변화를 정확하게 처리할 수 있다.

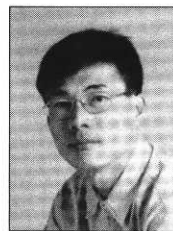
일정한 구역(학교, 회사 정문 등)에 카메라를 설치하여 드나드는 차량의 내용을 저장하여 응용할 수 있다.

참 고 문 헌

- [1] 김봉기, "멀티미디어 데이터베이스를 위한 2단계 내용기반 영상 검색 기법", 숭실대학교 박사학위논문, 1998.
- [2] J. S. Boreczky and L. A. Rowe, "A Comparison of Video Shot Boundary Detection Techniques", Proc. SPIE 2670, pp. 170-179, 1996.
- [3] 나재형, 나재형, 남석우, 오해석, "내용기반 비디오 검색을 위한 동영상에서 자막 추출", 한국정보과학회 추계학술발표논문집, Vol.24, No.2, pp.463-466, 1997.
- [4] 서창진, 김선숙, 차의영, "영상검지기를 이용한 자동차 추적시스템에 대한 연구", 한국정보과학회 '98 가을 학술발표논문집(II), 제25권 제2호, pp.423-425, 1998.
- [5] 김기옥, 김형주, "비디오 주식 시스템의 설계 및 구현", 정보과학회논문지(B), 제24권 제6호, 1997.
- [6] 이미숙, 황본우, 이성환, "내용기반 비디오 검색을 위한 장면

전환 검출 방법의 성능 분석", 인공지능학회, 1997.

- [7] 이종구, 양명섭, 이정열, 정찬근, 장옥배, "뉴스 비디오 검색을 위한 자동인덱스 모델의 설계 및 구현", Vol.24, No.1, pp. 543-546, 1997.
- [8] Boon-Lock Yeo, Bede Lie, "Visual content highlighting via automatic extraction of embedded caption on Mpeg compressed video," SPIE Vol.2668, pp.38-47, 1996.
- [9] 전병태, 소 경, "제한된 검색을 이용한 효율적인 차량 계수", 정보과학회논문지, Vol.6, No.6, pp.611-624, June, 1996.
- [10] 허준구, 박세현, 정기철, 김항준, "색상 변화 분석에 의한 교통 정보추출", 한국정보과학회 '98 가을 학술발표논문집(II), 제25권 제2호, pp.473-475, 1998.



남 석 우

e-mail : swnam@hcc.ac.kr

1988년 숭실대학교 전자계산학과(공학사)

1992년 숭실대학교 전자계산학과(공학석사)

1994년 숭실대학교 컴퓨터학과(공학박사
수료)

1988년~1993년 한국국방연구원 연구원

1993년~현재 혜천대학 컴퓨터통신계열 조교수

관심분야 : 멀티미디어, 웹어플리케이션, 영상처리, 데이터베이스



오 해 석

e-mail : oh@computing.soongsil.ac.kr

1975년 서울대학교 응용수학과(이학사)

1981년 서울대학교 계산통계학과(이학석사)

1989년 서울대학교 계산통계학과(이학박사)

1982년~현재 숭실대학교 컴퓨터학과 교수

관심분야 : Multimedia, Database, MIS