

論文99-36C-8-7

# 영상 기반의 실시간 교통 감시 시스템

## (Vision-based Real-time Traffic Surveillance System)

朴世鉉\*, 鄭基哲\*, 許俊九\*, 金恒俊\*

(Se Hyun Park, Keechul Jung, Jun Koo Hea, and Hang Joon Kim)

### 요 약

본 논문에서는 영상에 기반한 실시간 교통 감시 시스템을 구현한다. 영상 기반의 교통 감시 시스템은 루프 감지기 등의 센서를 이용한 방법에 비해 비용과 설치, 유지, 보수 면에서의 장점으로 인하여 많이 연구되고 있다. 제안한 시스템은 인터넷상에서 FPA (Field Processing Agent)와 TSM (Traffic Surveillance Manager)으로 구성되며, FPA는 TSM에게 도로 영상과 차량의 속도, 도로 점유율과 같은 교통 정보를 제공한다. 차량의 평균 속도와 도로 점유율은, 도로색 영상과 연속된 입력 영상간의 샘플링 지점의 색상 차이 변화를 이용하여 추출한다. 제안한 방법은 근사적인 교통정보를 추출해 주며, 입력 영상 전체에 대한 처리 과정 없이 제한된 영역만을 처리하기 때문에, 실시간 감시 시스템을 구축하는데 용이하다.

### Abstract

This paper presents a vision-based real-time traffic surveillance system. Current research in machine vision applied to traffic is due to its potential for more powerful process, its flexibility and its lower cost. Our traffic surveillance system consists of FPA and TSM on the Internet. The FPA provides image data, vehicle speed and vehicle density for the TSM. Traffic parameters, the vehicle speed and density, are extracted over many frames using differences between the sampling points of a background image and those of incoming frames. The FPA works by just processing groups of pixels without any understanding of the image. Although this system has an accuracy limitation, it can be used in applications requiring an approximate vehicle density and vehicle speed in real-time.

### I. 서 론

최근 도로상의 차량으로부터 도로 상황 파악, 교통량 분석, 교통 관제 등의 작업을 실시간으로 수행할 수 있는 자동화 시스템에 대한 요구가 급격히 증가되고 있다<sup>[1-8]</sup>. 이를 위한 일반적인 방법에는 자기 루프(magnetic loop detector)를 이용하는 방법과, 영상 처리를 기반으로 하는 방법이 있다. 최근에는 영상 처

리를 기반으로 한 교통 감시 시스템에 대한 연구가 활발히 진행되고 있는데<sup>[2,9]</sup>, 이는 센서를 이용한 시스템에 비해 여러 가지 장점을 가지고 있기 때문이다. 이러한 장점에는 기존의 방법과는 달리 보다 다양한 정보(차량의 모양, 속도, 점유율, 교통량, 이동 경로 등)를 추출할 수 있다는 점, 설치와 유지에 따른 비용 면에서 저렴하다는 점 등이 있다. 영상 처리를 이용한 교통 감시 시스템 중에는 차량의 모양과 도로 환경에 대한 3차원 모델링을 이용한 방법이 있다<sup>[10]</sup>. 이러한 시스템은 차량의 모양과 진행 방향, 이동 경로 등의 정보를 추출할 수 있지만, 수행 속도가 느리다는 단점이 있다. 그와 달리 프레임 차이나 특징 벡터를 기반으로 한 방법은, 상대적으로 수행 속도가 빠른 반면에

\* 正會員, 慶北大學校 컴퓨터工學科  
(Dept. of Computer Engineering, Kyungpook National University)

接受日字:1999年1月26日, 수정완료일:1999年7月19日

제공하는 정보가 차량의 속도, 차량의 개수, 도로의 점유율 등에 제한된다<sup>[4, 11, 12]</sup>.

본 논문에서는 영상 기반의 실시간 교통 감시 시스템을 구현한다. 본 시스템은 동영상의 제한된 영역에서의 정보를 이용하여 차량의 평균 속도와 점유율을 실시간에 측정한다. 제안한 시스템은 인터넷상에 FPA와 TSM으로 구성되며, FPA는 TSM에게 도로 영상과 차량의 속도, 도로 점유율을 제공한다. 차량의 평균 속도와 점유율 등의 교통 정보는 차량을 추출하는 과정 없이 도로색 영상과 입력 영상의 샘플링 지점에서의 색상 차이의 변화에 의해 추출된다. 도로의 점유율은 구해진 색상 변화 그래프 상에서 경계값을 설정하여, 총 측정 시간당 차량에 의해 가려진 시간으로 정한다. 본 연구에서 FPA는 Apache 웹 서버를 사용하며, 웹상의 사용자 인터페이스, 영상 처리 프로그램 등으로 구성되어 있으며, TSM은 보편적인 웹 브라우저를 사용한다. 본 논문에서 제안한 방법은 차량 추출이나 영상 분할과 같은 입력 영상 전체에 대한 처리 과정 없이, 제한된 영역만을 이용함으로써 실시간에 교통 정보를 추출할 수 있다.

본 논문의 구성은 다음과 같다. 먼저 II장에서는 교통정보 시스템에 대한 관련 연구를 살펴보고, III장에서는 교통 정보 추출 방법에 대한 설명을 한다. IV장에서는 구현된 시스템에 대한 설명과 실험 결과를 보이고, V장에서 결론을 내린다.

## II. 관련 연구

도로 영상에서 차량의 위치와 형태 정보를 추출하기 위한 여러 시스템이 있다. Reading대학의 CARRS(CAR Recognition System)<sup>[5]</sup>와 VIEWS(Visual Interpretation and Evolution of Wide-area Scenes)<sup>[6]</sup>가 가장 대표적인 시스템이다. CARRS는 Reading대학의 초기 연구 결과로써, 차량 모델과 이미지 사이의 매핑을 통하여 차량의 위치를 찾는 시스템이다. VIEWS는 차량의 추적을 통하여 차량의 형태, 크기, 위치, 속도 등을 추출하는 시스템이다. 또한 Koller는 recursive estimation 방법을 사용하여 차량을 추적하였고<sup>[10]</sup>, 조명모델을 도입하여 차량에 대한 좀 더 정확한 정보를 추출하였다. 반면, 영상에 대한 이해 없이 단순히 motion detection이나 프레임 차이를 이용한 시스템들이 있다. London대학의 Hoose에

의한 IMPACTS는 이미지를 차량과 레인의 길이에 따라서 Cell로 나누고 각 Cell의 상태를 이용하여 차량을 추출하였다<sup>[7]</sup>. Jet Propulsion Laboratory의 WADS(Wide Area Detection System)은 이미지의 작은 부분을 처리하여 실시간으로 차량의 속도와 개수를 추출하였다<sup>[8]</sup>. 이러한 시스템은 차량이 아닌 물체에 대해서도 반응하는 단점이 있으나 차량의 속도나 개수의 추정에 값싸고 유용하다. 본 논문에서 제안한 방식은 차량 추출이나 모델에 관한 정보를 사용하지 않고 연속된 영상에서 제한된 영역내의 정보만을 이용하므로, 근사적인 교통정보를 실시간으로 제공해 주는 시스템에 용이하다.

## III. 교통 정보 추출

본 시스템은 도로영상의 동적인 정보를 이용하여 차량의 평균속도와 점유율을 추출하고, 이를 인터넷망을 통해 제어한다. 본 시스템은 FPA와 TSM으로 구성되어 있다. TSM으로부터 교통 정보 추출을 요구받은 FPA는 TSM에게 도로 영상, 차량속도, 차량 점유율 등의 교통 정보를 제공한다. 본 장에서는 실시간 도로 감시 시스템에 사용될 수 있는 교통 정보 추출 기법을 기술한다.

### 1. 도로색 추정

본 시스템에서 제안한 도로 정보 추출 방법은 제한된 영역에서의 색상 변화 정보를 이용한다. 따라서 도로영상에서 차량속도와 점유율을 구하기 위해서는 추정된 도로색 영상이 필요하다. 본 연구에서는 정해진 시간의 입력 영상에서 연속된 프레임들을 관찰하여 도로색을 구한다. 어느 한 프레임에서 도로의 일부는 차량에 의해서 가려진다. 시간이 지남에 따라 차량은 지나가고 가려던 도로는 나타난다. 연속된 영상이 충분히 길다면, 차량이 없는 깨끗한 도로 영상을 발견할 수 있다. 도로를 나타내는 픽셀의 색상은 변함이 없는데 반하여 차량을 나타내는 화소는 다른 색상을 가진다. 도로색은 픽셀 단위로 생성되며, 가장 많이 나타나는 색상값을 그 픽셀의 도로색으로 추정한다. 도로색 추정 단계는 도로의, 일반적인 조건과 조명의 변화에 부합하기 위해서 주기적으로 반복된다. 이로써 도로색 추정은 영상의 조건에 적응성을 가진다. 다음의 그림 1은 위의 방법에 의해 추정된 도로색 영상이다. 좌측

의 그림 1(a)가 입력 영상열중 한 프레임이고, 우측의 그림 1(b)가 추정된 도로색이다.

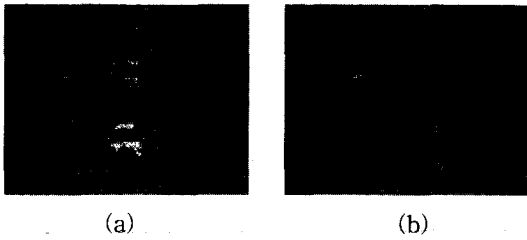


그림 1. 도로색 영상 추출의 예 (a) 도로 영상 (b) 도로색 영상  
Fig. 1. A sample of background image. (a) Road image (b) Background image.

2. 교통정보 추출

본 장에서는 차량의 평균 속도와 도로 점유율 추출 방법을 설명한다. 그림 2는 교통 정보 추출 단계에 대한 블록 다이어그램으로써 도로색과 동영상을 입력받아 교통 정보를 추출하는 과정을 보인다. 어떤 화소의 색상과 도로색과의 차이를 분석하기 위해서는 먼저 기준이 되는 도로색과의 색상 차이가 정의되어야 한다. 색상 차이는 두 화소의 RGB 값의 유클리디안 거리(Euclidean distance)로 정의한다. 입력 화소  $p$ 와 추정 도로색  $q$ 의 색상 차이  $d(p,q)$ 는 다음과 같이 나타낸다.

$$d(p,q) = \sqrt{(p_R - q_R)^2 + (p_G - q_G)^2 + (p_B - q_B)^2} \quad (1)$$

임의의 화소  $x$ 에 대해  $x_R, x_G, x_B$ 는 각각 화소  $x$ 의 R,G,B 값을 나타낸다.

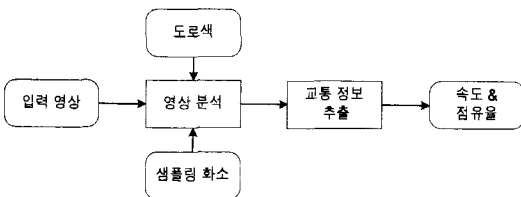


그림 2. 교통 정보 추출  
Fig. 2. Traffic parameters estimation.

이러한 도로색과의 색상 차이를 바탕으로, 화소에 대한 시간 흐름에 따른 색상 차이를 그림 3과 같이 구할 수 있다. 그림 3은 중앙의 두 화소의 시간차에 따른 색상 차이의 변화를 나타낸다. 그림 3의 (a)와 (b)는 각각 낮과 밤에 촬영한 동영상에 대한 샘플링

화소 한 점에서의 바탕색과의 색상차(식 1)를 나타낸 그림이다. 각 그림의 좌측의 영상은 동영상중의 한 프레임을 나타낸다.

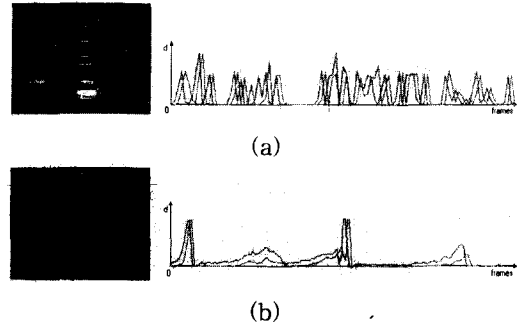


그림 3. 색상 변화 (a) 낮 (b) 밤  
Fig. 3. Color difference of traffic scenes. (a) Day (b)Night.

그림 3에서 볼 수 있듯이 지정한 화소에 차량이 있는 시점에서는 높은 값을 가지고 차량이 존재하지 않는 시점에서는 거의 도로색과 같음을 볼 수 있다. 또한 두 그래프는 일정한 프레임 차를 가지고 비슷한 모양으로 변한다. 한 그래프를 일정한 양만큼 이동하면 두 그래프는 거의 일치하게 된다. 이 이동한 양으로부터 차량의 속도를 구할 수 있는데, 이는 두 개의 일련의 색상 변화 값 사이의 상관계수(correlation)를 계산함으로써 이루어진다. 비교할 프레임 수를  $n$ , 프레임  $t$ 에서 첫 번째 화소의 색상 차이값을  $g_t$ , 두 번째 화소의 색상 차이값을  $f_t$ , 첫 번째 화소의 평균 색상 차이를  $\bar{f}$ , 두 번째 화소의 평균 색상차이를  $\bar{g}$ , 두 그래프 중 하나를 프레임 축으로 평행 이동하여 가장 비슷하게 겹치게 하는데 필요한 이동량을  $k$ 라 하면, 상관계수  $C_k$ 는 다음과 같다.

$$C_k = \frac{\sum_{t=0}^n (f_t - \bar{f}) \cdot \sum_{t=0}^n (g_{t+k} - \bar{g})}{\sqrt{\sum_{t=0}^n (f_t - \bar{f})^2} \cdot \sqrt{\sum_{t=0}^n (g_{t+k} - \bar{g})^2}} \quad (2)$$

이를 적용하여 상관계수를 구하면 그림 4(b)와 같은 상관그램(Correlogram)을 얻을 수 있다. 상관그램에서 가장 큰 값을 보이는  $k$ 값을 이동량으로 결정한다. 그림 4는 그림 2의 각 동영상에 대한 색상 변화 그래프(a)에 대해 상관계수를 구하고(b), 가장 높은 상관계수에 따라 프레임을 이동시켜 일치시킨 그래프(c)이다. 그림 4는 오후 5시 경의 영상을 가지고 실험한 결과이고, 그림 5는 같은 방법을 오후 8시경의 영상에 적

용한 결과이다.

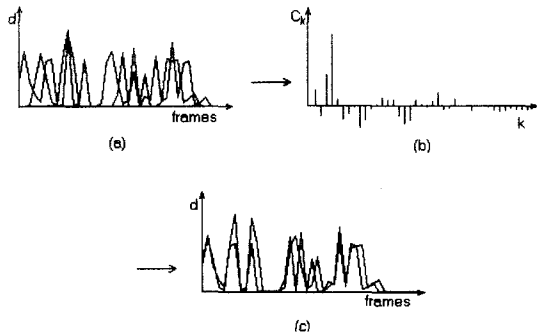


그림 4. 색상 변화 그래프: (a) 색상 차이 (b) 상관그림 (c) 이동된 그래프 (오후 5시경)  
 Fig. 4. Color difference graph: (a)Color Difference (b)Correlogram (c)Shifted Graph (5 p. m.).

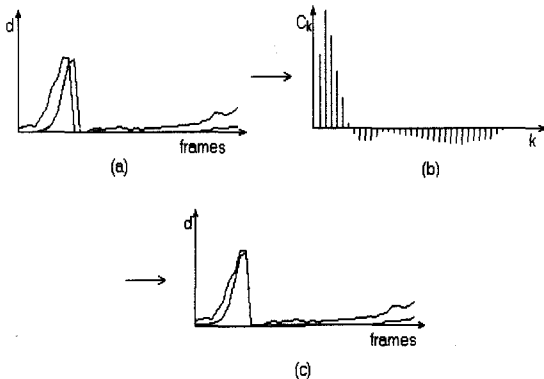


그림 5. 색상 변화 그래프: (a) 색상 차이 (b) 상관그림 (c) 이동된 그래프 (오후 8시경)  
 Fig. 5. Color difference graph: (a)Color Difference (b)Correlogram (c)Shifted Graph (8 p. m.).

샘플링한 두 화소의 도로에서의 실제 거리를  $L$ , 동영상에서 프레임간의 시간을  $T$ 라고 하면 속도  $s$ 는 다음과 같이 구할 수 있다.

$$s = L / (T \cdot k) \tag{3}$$

차량 진행 방향에 있는  $M$ 개의 화소를 선택하기 위하여 도로상의 한 차로와 직각 방향의 가상의 선  $M$ 개를 설정하고, 이  $M$ 개의 선 각각을  $N(>0)$ 개의 균일한 간격으로 나누어  $N$ 개의 쌍을 만들 수 있다. 이렇게 얻어진  $N$ 개의 쌍을 각각 속도를 구하기 위한  $M \times N$ 개의 화소로 선택한다.

차량의 점유율은 총 측정된 시간당 차량에 의해서 가려진 시간으로 정의한다<sup>[13]</sup>. 차량에 의해서 가려진 시간은 그림 3과 같은 그래프에서 도로색으로 설정한

경계값을 초과하는 시간이다. 정적인 경계값 설정은 조명과 노이즈의 영향을 충분히 극복 못하는 문제를 가진다. 그래서 경계값은 조명의 변화, 즉 도로색에 따라서 동적으로 바꾼다. 이는 실험적으로 설정하였으며, 표 1은 시간에 따라 동적으로 변화되는 경계값을 보인다. 표 안에서의 각 시각에서의 평균 밝기는 샘플링 지점에서의 평균 밝기으로써, RGB값의 평균값을 의미하며, 0-255까지의 숫자 값을 가질 수 있다. 경계값 또한 0-255 값 중, 한 값을 가지며 각 시간(평균 밝기) 별로 실험에 의해 구해진 값을 사용한다.

표 1. 시간에 따른 도로색 경계값

Table 1. Threshold value for road color.						
시각	5시	6시	7시	8시	9시	우천시(17시)
평균 밝기	165	150	120	90	85	116
경계값	3	4	7	30	35	17

#### IV. 구현 및 실험결과

본 장에서는 제안한 교통 감시 시스템의 구현 및 성능에 관해 기술한다. 전체적인 교통감시 시스템의 구성은 그림 6과 같다. 교통 감시 시스템은 도로상에서 정보를 수집하는 FPA와 중앙에서 이를 관리하는 TSM으로 구성된다. FPA와 TSM은 인터넷을 통해 연결된다. FPA는 Linux OS 상에서 C++과 Java를 이용해 구현되었다. FPA는 삼성 SV-N25 비디오 카메라를 영상입력장치로 사용하고 IBM 팜탑 PC를 프로세서로 사용한다. TSM는 범용 컴퓨터와 일반적인 웹 브라우저로 구성된다. FPA와 TSM은 인터넷을 통해 연결되며 HTTP 프로토콜상에서 동작한다.

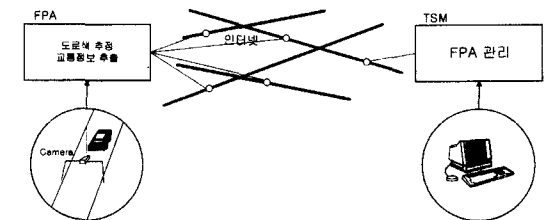


그림 6. 교통 감시 시스템 구성도  
 Fig. 6. TSS configuration.

표 2는 TSM과 FPA 사이의 가능한 메시지, 즉 질의와 응답을 보여준다. 그림 7은 표2의 메시지의 순서를 나타내는 다이어그램이다. (a)는 정지영상을 요구하

는 메시지이다. (b)는 동영상에 관한 메시지인데 초당 몇 프레임을 처리할 것인지를 결정한 후 영상을 보낸다. (c)는 수동으로 레인과 라인 등의 매개변수를 설정하는 메시지이다. (d)는 교통정보, 즉 속도와 점유율을 요청, 응답한다.

표 2. TSM과 FPA간의 메시지

Table 2. Lists of messages passed between the TSM and the FPA.

Type	TSM	FPA
a Image messages	Get image	Send image
b Video messages	Set rates	Acknowledge
	Start video	Send frames
	Stop video	Acknowledge
c Region setting messages	Set region	Acknowledge
d Traffic data messages	Start Extractor	Send reports

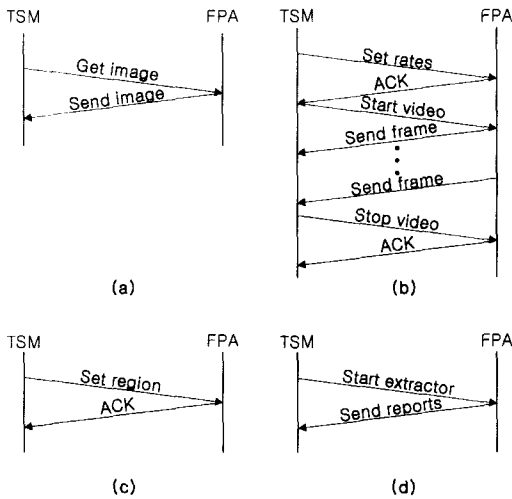


그림 7. FPA와 TSM간의 순서 다이어그램  
Fig. 7. Time sequence diagram between the TSM and the FPA.

구현된 시스템은 단일 레인(한 차로)만 처리하고 영상의 부분(라인당 5 화소)만 처리한다. 그림 8은 도로의 환경 설정을 나타낸다. 그림에서는 5개의 가상선만을 나타내었다. 각각의 선마다 5개의 샘플링 화소를 설정한다. 레인은 10개의 라인으로 구성되어 있고 라인과 라인의 거리는 2m이다. 실제도로에서 이 거리는 사람이 직접 측정하였으며, 2m는 아래 표 3에 나와 있듯이 적절한 평균속도를 측정할 수 있는 실험치이다.

본 시스템과 같은 환경 설정으로 인해 시속 70km/h까지의 속도만이 실용적인 측정이 가능하다. 이는 입

력 영상의 프레임간의 시간차, 즉 시간당 캡춰되는 프레임의 수와 샘플링 선간의 거리에 의존한다. 표 3에서 (a)는 초당 5프레임을 캡춰하였을 때 각 샘플링 선간에 측정할 수 있는 속도를 나타내고, (b)는 초당 10프레임을 캡춰할 때이다. 표에서 가로축의 거리는 2m마다 있는 가상 추출선을 나타낸다. 세로축의 프레임차는 가상 추출선에서 차량이 추출되었을 때, 프레임차가 얼마나 나는가를 나타낸다. (a)에서 2m의 가상 선에서 추출되고, 3프레임 차가 나면 24km/h의 속도가 된다.

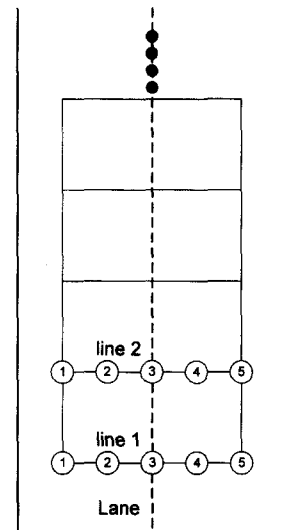


그림 8. 시스템에서 사용한 도로의 가상 추출선  
Fig. 8. Virtual Detection Line of the System.

표 3. 본 시스템의 환경 하에서 측정 가능한 속도 (a) 5 프레임/초 (b) 10 프레임/초

Table 3. Possible velocity of the environment (a) 5 frames/sec. (b) 10 frames/sec.

프레임차	거리	2m	4m	6m	8m	10m
	1		72	144	216	288
2		36	72	108	144	180
3		24	48	72	96	120
4		18	36	54	72	90
5		14	28	43	57	72
6		12	24	36	48	60
7		10	20	30	41	51
8		9	18	27	36	45
9		8	16	24	32	40
10		7	14	21	28	36

(a)

프레임차	거리				
	2m	4m	6m	8m	10m
1	36	72	108	144	180
2	18	36	54	72	90
3	12	24	36	48	60
4	9	18	27	36	45
5	7	14	21	28	36
6	6	12	18	24	30
7	5	10	15	20	25
8	4	9	13	18	22
9	4	8	12	16	20
10	3	7	10	14	18

(b)

구현된 시스템을 검증하기 위해서 1 레인에 대해 6 시간의 비디오 데이터에 관해서 실험하였다. 시스템의 실제 수행을 평가하기 위해서 오후 5시부터 오후 10시 까지 비디오 카메라로 촬영한 영상을 초당 10프레임으로 샘플링하여 실험하였다. 이 도로영상은 50개의 연속된 프레임으로 구성된 2초 길이의 영상으로 나뉘어진다. 본 시스템에서 계산된 속도와 점유율은 이 2초에 대한 속도와 점유율이다. 시스템의 카메라는 지상에서 6.6 m 위에 설치되어 있고 첫 검출 라인까지는 30m거리이다. 또 대상 차선을 정면으로 관측한다. 표 4는 제안된 시스템의 성능평가 결과이다. 표 4에서 볼 수 있듯이 차량의 평균속도에 대한 평균 에러율은 9.2%를 보이고 있다. 또 점유율에 대한 평균 에러율은 20.01%를 보인다. 표에서 보면 19시와 우천시에 에러율이 높은데 이것은 빛 변화가 심한 환경 - 해질 때, 비올 때 - 일 때 나타나는 현상인 것 같다. 본 시스템에서 사용한 방법은 차를 트래킹(tracking)하지 않고 교통 정보를 추출하므로 정확도에는 다소 문제가 있지만, 본 시스템의 적용 환경이 실시간 교통 통제<sup>[14]</sup> 시스템인 점을 고려할 때 유용할 것으로 보인다.

표 4. 실험 결과

Table 4. Experimental results.

측정시간	속도 (km/h)			점유율 (%)		
	실제	측정치	에러율(%)	실제	측정치	에러율(%)
17	64	57	10.9	20	18	10.0
18	53	54	1.8	42	36	14.3
19	42	51	21.4	62	43	30.6
20	57	54	5.3	39	34	12.8
21	63	57	9.5	23	19	17.4
우천시(17시)	63	67	6.3	20	13	35.0
평균			9.2			20.01

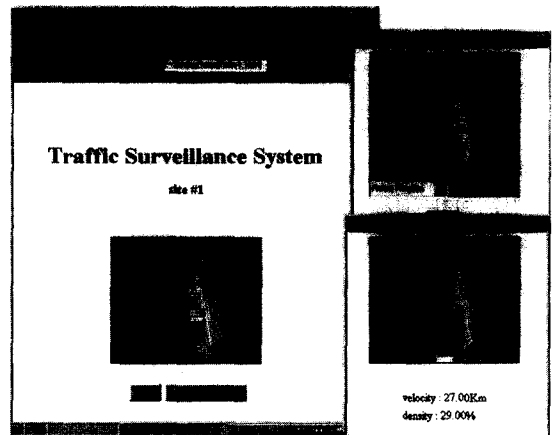


그림 9. 시스템의 사용자 인터페이스  
Fig. 9. User Interface of TSS.

TSM은 FPA에게 교통정보추출을 요구하고 관리한다. TSM은 이러한 태스크를 Internet Explorer나 Netscape 등의 일반적인 웹 브라우저를 사용해 처리한다. 그림 9는 제안된 시스템 중 TSM에서의 사용자 인터페이스이다. 좌측의 윈도우는 여러 사이트의 FPA를 관리하는 윈도우이다. 왼쪽의 [ video ] 버튼은 현재의 실시간 영상을 보내주고, 우측 아래 윈도우를 연다. 우측 아래의 창은 2초 길이의 영상에 대한 추출 파라미터, 속도와 점유율이다. 다음 [ management ] 버튼은 우측 위 윈도우를 연다. 우측 위의 윈도우는 시스템의 환경 설정에 관한 윈도우이다.

## V. 결 론

본 논문에서는 영상을 기반으로 한 실시간 교통감시 시스템을 구현한다. 본 시스템은 도로영상의 동적인 정보를 이용하여 차량의 평균속도와 점유율을 추출하고 이를 인터넷망을 통해 제어한다. 차량의 평균 속도와 점유율은 차량을 추출하는 과정 없이 도로색 영상과 입력 영상의 샘플링 지점의 색상 변화에 의해 추출된다. 점유율은 총 측정 시간당 차량이 점유한 시간으로 정의한다. 본 논문이 제시한 방법은 근사적인 교통 정보를 추출해 주며, 수행시간이 적게 드는 장점을 가진다. 또한 인터넷을 이용한 서비스의 제공으로 보다 많은 사용자들에게 서비스할 수 있다는 장점이 있다. 그러나 본 시스템이 제공하는 정보는 평균 속도와 평균 점유율로써, 특정 차에 대한 정확한 교통정보(속도) 등은 추출하기가 어렵다. 이러한 문제는 차를 트래킹

(tracking)하는 방법을 도입해야 할 것이다.

본 시스템의 응용분야로는 통행량 정보 추출 등 기존의 루프 방식의 차량감지기의 기능을 대신할 수 있고, 그밖에 현재 교통량을 실시간 비디오를 통해 원격지에서 감시할 수 있는 기능을 제공함으로써 신호등 제어를 위한 센서, 통행량 조사기, 교통 위반 감지기 등의 용도에 사용할 수 있다. 보다 다양한 도로 교통 정보를 실시간으로 제공하는 시스템 구축이 향후 연구 과제이다.

### 참 고 문 헌

- [1] N. Hoose, and L. G. Willumsen, "Automatically extracting traffic data from videotape using the CLIP4 parallel image processor," *Pattern Recognition Letters*, Vol. 6, pp. 199-213, 1987.
- [2] C. Setchell, "Applications of computer vision to road-traffic monitoring," *PhD thesis, Bristol University, Bristol, UK*, September 1997.
- [3] S. H. Park, J. K. Lee, and H. J. Kim, "Evolutionary segmentation of road traffic scenes," *Proceedings of IEEE ICEC'97(International Conference on Evolutionary Computation)* pp 397-400, Indianapolis, USA, 1997.
- [4] 허준구, 박세현, 정기철, 김항준, "색상변화 분석에 의한 교통정보 추출," *한국정보과학회 '98 가을 학술발표논문집*, Vol. 25, No. 2, pp. 473-475, 1998
- [5] R. M. Bodington, G. D. Sullivan, and K. D. Baker. "The consistent labelling of image features using an ATMS," *Image and Vision Computing*, Vol. 7, No. 1, pp. 7-12, February 1989.
- [6] A. D. Worrall, G. D. Sullivan, and K. D. Baker. "Pose refinement for active models using forces in 3D," *In Proceedings of the Third European Conference on Computer Vision*, pp. 341-350, May 1994.
- [7] N. Hoose, *Computer Image Processing in Traffic Engineering*, Research Studies Press, 1991.
- [8] R. M. Inigo, "Application of machine vision to traffic monitoring and control," *IEEE Transactions on Vehicular Technology*, Vol. 38, No. 3, pp. 112-122, August 1989.
- [9] H. J. Kim, E. Y. Kim, J. W. Kim, and S. H. Park, "MRF Model based image segmentation using hierarchical distributed genetic algorithm," *IEE Electronics Letters*, Vol. 34, No. 25, pp. 2394-2395, 1998.
- [10] D. Koller, K. Daniilidis, T. Thorhallson, and H. Nagel, "Model-based object tracking in traffic scenes," *In European Conf. Computer Vision*, pp. 437-452, Italy, May 1992.
- [11] 전병태, 소정, "제한된 검색 영역을 이용한 효율적인 차량계수," *정보과학회 논문지*, 제6권, 제6호, 611-624쪽, 1996년
- [12] S. Takabu, "Measurement of traffic flow using real-time processing of moving pictures," *in 32nd. Conf. on Vehicular Technology*, pp. 488-494, San Diego, CA, 1982.
- [13] L. C. Edie, "Discussion of traffic stream measurements and definitions," *Proc. Second International Symposium on the Theory of Traffic Flow*, OECD, pp. 139-154, Paris, France, 1963.
- [14] 교통정보센터, "http://www.kortic.or.kr," 경찰청, 도로 교통 안전 관리공단

저 자 소 개



朴世鉉(正會員)

1995년 경북대학교 컴퓨터공학과 졸업(공학사). 1997년 경북대학교 대학원 컴퓨터공학과 졸업(공학석사). 1997년 ~ 현재 경북대학교 대학원 컴퓨터공학과 박사과정. 주관심분야는 인공지능, 문자인식,

Computer Vision, 영상처리



許俊九(正會員)

1998년 경북대학교 컴퓨터공학과 졸업(공학사). 1998년 ~ 현재 경북대학교 대학원 컴퓨터공학과 석사과정. 주관심분야는 게임이론, Computer Vision, Motion Detection



鄭基喆(正會員)

1994년 경북대학교 컴퓨터공학과 졸업(공학사). 1996년 경북대학교 대학원 컴퓨터공학과 졸업(공학석사). 1996년 ~ 현재 경북대학교 대학원 컴퓨터공학과 박사과정. 1999년 The German Research Center

for Artificial Intelligence GmbH(DFKI) 교환 학생, 주관심 분야는 문자인식, 비디오 인덱싱, 영상 처리



金桓俊(正會員)

1977년 서울대학교 전기공학과 학사학위 취득. 1979년 KAIST 전기 및 전자 석사학위 취득. 1997년 Shizuoka 대학원 Electronics 박사학위 취득. 1980년 ~ 현재 경북대학교 컴퓨터공학과 교수. 주관심분

야는 인공지능, 문자인식, 음성인식, 병렬처리, 컴퓨터 구조