

퍼지 클러스터링을 이용한 이동 차량 추적

Moving Vehicle Tracking using Fuzzy Clustering

양 상 규*, 이 정 재*, 소 영 성**

Sangkyu Yang*, Jungjae Lee*, Youngsung Soh**

※본 논문은 1996년도 정보통신연구관리단 대학기초 연구비지원에 의해 연구되었음.

요 약

차량의 증가와 도로 신장율의 저조로 교통체증은 날로 심해지고 있다. 이 문제를 풀기 위한 한 방법은 교통류에 관한 실시간 정보를 얻어 교통신호를 동적으로 제어하는 것이다. 현재까지 교통류 정보를 얻기 위해 가장 많이 쓰인 것은 루프코일 인데 이것은 통과 차량의 평균속도를 측정하기 위해 필요한 차량 추적을 할 수 없는 단점이 있다. 따라서 이 문제를 해결하기 위해 영상 검지기가 출현하게 되었고 농담영상 혹은 칼라 영상열을 가정하고 있다. 본 논문에서는 칼라영상의 형태로 주어진 영상열에서 퍼지 클러스터링에 기반을 둔 이동 차량 추적 방법을 제안한다.

ABSTRACT

Due to the rapid increase of vehicles and poor availability of roads, traffic congestion problem is about to explode. To solve this problem, we need real time information about traffic flow to control traffic signals dynamically. Until now loop coil is the most prevalent sensor used for obtaining traffic flow information. However, it is not able to track individual vehicles which is essential in estimating the average vehicle speed. As a result, image sensors started to find their role in this problem domain. Several systems based on image sensors were proposed which assumes either gray level or color image sequence. In this paper, we propose moving vehicle tracking method based on fuzzy clustering assuming a color image sequenc.

I. 서 론

도로상의 교통류 측정은 유동적인 교통류 제어를 위한 적응적인 신호 주기, 우회도로 설정, 여행자 운행 예정 시간 제공 등에 응용되어 질 수 있고 향후 자동차 자동 항법 장치, 인공 위성의 GPS시스템과 연관

된 차량위치 정보제공 등 전반적인 도로의 효율성 증진을 위한 정보로 사용이 가능하다. 기존의 교통류 측정은 loop coil을 주로 이용하고 있는데 이러한 방법에서는 시설물을 설치하거나 유지 및 보수 시에 토목공사 등의 번거로움과 기타 주변 환경의 영향으로 성능 열화의 문제점을 가지고 있다. 도로에 매설된 loop coil은 통과하는 차량에 의해서 발생된 자장의 변화에 의해 교통류를 측정하게 되는데 이 방법은 도로상에서 달리는 차량의 속도, 밀집도 등에 의해 오

*명지대학교 정보통신공학과 석사과정

**명지대학교 정보통신공학과 교수

동작 할 수 있으며 다양한 정보를 측정하기 위한 기능도 포함하지 않고 있다. 따라서 최근엔 영상처리 기술을 이용하여 근본적인 문제점을 해결하고 도로상의 다양한 정보를 효과적으로 제공하기 위한 기술개발이 산업화 측면에서 요구되고 있다. 영상 취득을 위해서는 카메라가 필요하고 카메라는 노변에 설치 가능하며 이동성이 보장되기 때문에 loop coil의 단점을 해결 가능하다. 교통류 정보에는 많은 것이 포함될 수 있으나 가장 중요한 것이 통과차량대수, 평균속도 등이다. 이중 차량의 평균운행속도를 알기 위해서는 차량의 추적이 이루어져야 하는데 기존의 방법은 대부분 농담영상을 가정하고 이동점 혹은 이동영역을 추적하였는데 본 논문에서는 칼라 영상을 가정하고 칼라정보에 바탕을 둔 퍼지 클러스터링을 통해 동질의 칼라를 갖는 클래스들로 분류하고 각 클래스 내의 차량들에 대해 영역상응을 행함으로써 이동차량을 추적하는 방법에 제안한다.

II. 관련 연구

Dubuisson et al. [1]은 일정거리 떨어져 있는 두대의 카메라가 잡은 칼라 영상에 나타나는 차량의 추적을 위해 칼라 정보를 이용한 방법을 제시하였는데 우선 영상 분할을 위해 차영상을 다음과 같이 구하였다.

$$d(i, j) = \max_{k=R, G, B} \{|f_k^1(i, j) - f_k^2(i, j)| \times |f_k^2(i, j) - f_k^3(i, j)|\}$$

여기서 세장의 칼라 이미지 프레임 f^1, f^2 , 그리고 f^3 는 각각 시간 t_1, t_2, t_3 에서 표본 추출되었다.

각 프레임내 split-and-merge방법으로 나누어진 균일한 영역은 Canny에지 검출기를 이용해 얻은 정보와 결합하여 다시 합쳐지며, 이 영역으로부터 차영상을 이용해 얻은 분할 마스크를 이용해 배경으로부터 이동물체를 분리한다. Occlusion이 존재할 경우 맨 앞의 차량이 분리되어야 하는데 이를 위해서는 차량의 일반적인 특징을 이용해 구성된 모양에 대한 일반적인 모델을 사용한다. t_1 의 이미지 프레임에서 차량들의 문 부근에서 추출된 균일한 칼라영역과 차량들의 에지 정보는 데이터베이스에 저장되고 이 정보는

t_2 에서 추출된 칼라영역의 칼라 히스토그램과 데이터베이스 내에 저장된 정보와의 정합값을 색인값으로 사용하는 color indexing방법[6]을 통해 후보 차량의 수를 줄이고 최종적으로 에지 정합값을 결과인 유사도 값으로 결정하게 된다. 이 방법에서는 차량들의 순시속도 보다는 일정시간 동안의 평균속도를 구하고 영상내 차량 비교를 위해 차량속도에 대한 사전지식이 없기 때문에 두 대의 카메라로 부터 잡은 영상 중 관련없는 많은 프레임들을 비교하게 되는 단점이 있다.

Hoose et al [2]는 농담영상을 가정하고 통과 차량대수 계산과 이동물체 추적을 위해 여러 방법을 제안하였다. 이동물체가 전혀 없는 기준 프레임이 있는 것으로 가정하고 이후의 프레임들을 기준프레임에 비교하여 차량대수를 계산하는 background referencing방법, 이동물체가 전혀 없는 기준프레임의 존재를 가정하지 않는 image subtraction방법을 제시하였고 이동차량 추적을 위해서는 이웃한 프레임간의 차량의 위치는 상호 중첩되어 나타나는 것으로 가정하였다. S/W적으로 emulate하는 가상의 감지기를 표시해 놓고 그 감지기와 차량으로 생각되는 영역과의 중첩정보를 이용해 교통정보를 계산해 내었다. 이 방법에서의 차량추적은 연속된 프레임에서 동일차량이 중복될 수 있을 정도로 차량속도가 빠르지 않거나 혹은 표본주기가 충분히 짧은 것으로 가정하였고 감지기의 폭은 차량속도에 관련없이 고정된 넓이를 갖도록 되어 있어 차량대수 계수시 문제가 된다. 본 논문에서 제안하는 방법은 차량추적을 위해 연속된 프레임에서 차량중복을 가정하지 않고 또한 부수적으로 이동하지 않는 차량의 존재도 추출이 가능하다.

III. 제안된 방법

클러스터링 방법은 크게 Hierachical한 방법, Graph theoretic방법, Objective function방법으로 나누어 질 수 있으며 각각의 방법에는 Crisp분류법과 Fuzzy분류법이 있는데 본 논문에서는 fuzzy objective function 방법 중 fuzzy c-means알고리즘을 이동차량 추적에 이용하기 위한 방법을 제안한다. 같은 클러스터 내의 원소들은 유사한 성질을 가지며 각각의 클러스터에는 상대적으로 큰 성질차를 갖는다. Fuzzy c-me

(FCM) 알고리즘은 클러스터 중심값과 원소 상호간의 가중 유사치 값을 기반으로 반복적 수행을 통해 objective function의 최적화를 추구하는데 objective function은 다음과 같이 정의된다.[5]

$$J_m(U, v) = \sum_{k=1}^n \sum_{i=1}^c (\mu_{ik})^m (d_{ik})^2,$$

여기서, μ_{ik} 는 i^{th} 클러스터 중심과 k^{th} 입력 데이터의 퍼지 소속값을 나타내며, 그 값은 고전적인 분류법에서는 0 혹은 1의 값을 갖는 반면 fuzzy 분류법에서는 0과 1사이의 실수를 갖는다. d_{ik} 는 norm metric을 나타내며 EUCLIDEAN, DIAGONAL, MAHALANOBIS norm 등이 될 수 있다. m 값은 지수 가중치이고, 수렴된 클러스터의 fuzzy한 정도를 나타낸다. v 는 벡터값 형태의 클러스터 중심을 나타내는 행렬이고 U 는 fuzzy c-partition이다. FCM알고리즘은 다음과 같은 수행 과정으로 구성되어져 있다.

1. c ($2 \leq c \leq n$) 값을 결정한다.
2. m ($1 \leq m < \infty$) 값을 결정하고 metric norm을 결정한다.
3. fuzzy c-partition $U^{(0)}$ 행렬을 초기화한다.
 - (1) $\mu_{ik} \in [0, 1]; 1 \leq i \leq c, 1 \leq k \leq n$
 - (2) $\sum_{i=1}^c \mu_{ik} = 1; 1 \leq k \leq n$
 - (3) $0 < \sum_{k=1}^n < n; 1 \leq i \leq c$
4. 클러스터 중심 $U^{(l)}$ 값을 이용하여 $v_i^{(l)}$ 구한다. 각 클러스터 i 에 대한 클러스터 중심은 다음과 같다.

$$v_i = \frac{\sum_{k=1}^n (\mu_{ik})^m x_k}{\sum_{k=1}^n (\mu_{ik})^m}$$

5. $U^{(l)}$ 값을 갱신한다.

if $x_k \neq v_i^{(l)}$

$$\mu_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{m-1}}}$$

$$\text{else, } \mu_{ik} = \begin{cases} 1 & \text{for } (j=i) \\ 1 & \text{for } (j \neq i) \end{cases}$$

6. if $\|U^{(l)} - U^{(l+1)}\| \leq \epsilon$

else $l = l + 1$, and return to step 4.

여기서 c : 클러스터의 수
 m : 지수 가중치
 ϵ : 허용 오차 임계값이다.

본 논문에서는 상기 FCM을 이용하여 카메라가 잡은 연속된 영상 프레임들에 대한 칼라 영역 분할을 하고 영역분할 후 같은 칼라 클래스에 속하는 차량에 대해서만 추적관계를 규명한다. FCM알고리즘을 실제 적용하기 위해서는 몇 가지 파라메타를 설정해야 하는데 그중 하나가 클러스터수(c)이다. 대상 영상이 몇 개의 칼라 클래스를 가지고 있는지 미리 알고 있다면 문제가 없으나 실제 모르는 경우가 대부분이기 때문에 이를 위해 Huntsberger et al.[3]이 제안한 클러스터 collapse방법을 사용한다. 이 방법에서는 매 반복마다 일정갯수의 c 를 가정하는데 i 번째 반복에서 얻은 c 개 이하의 클러스터 중심과 $i+1$ 번째 반복에서 얻은 c 개 이하의 클러스터 중심간 클러스터 분리도를 계산하여 분리도가 낮은 클러스터는 서로 융합해 한 클러스터를 만들고 이를 영상내 모든 화소들에 대한 분류가 끝날 때까지 적용하여 만들어진 클러스터들이 최종적인 클러스터가 되게 하는 방법으로써 클러스터 갯수에 대한 사전지식 없이도 FCM알고리즘을 적용할 수 있는 토대를 제공해 준다. 이와 같은 과정을 거쳐 영상열이 칼라 분할되면 이제 추적은 같은 칼라 클러스터 내에서만 이루어지면 되는데 한 칼라 클래스내 연속된 각 프레임상에 한 개의 영역만이 있다면 그 영역들은 같은 물체를 뜻하며 다수의 영역이 있다면 그 영역들간 상응의 필요성이 생긴다. 영역상응은 아래 그림 1과 같이 NEARNESS를 정의하여 사용한다. 여기서 F^1, F^2 는 연속적인 두 이미지 프레임을 나타내며 프레임내에 존재하는 이동물체 후보영역은 외접사각형 정보를 이용 $A(sx^1, sy^1, ex^1, ey^1), B(sx^2, sy^2, ex^2, ey^2)$ 와 같은 형태로 표현한다. 여기서 s 는 start, e 는 end, x 와 y 는 x, y 좌표, 위첨자는 프레임 번호, 아래 첨자는 프레임내 영역번호를 나타낸

다. 그림 1의 NEARNESS를 모든 후보영역에 적용할 경우 동일 방향의 인접차량과 반대 방향의 인접차량까지 후보로 삼아 잘못된 상용결과를 보일 수 있는데 이의 방지를 위해 후보영역 외접사각형이 Table 1의 조건을 만족하는 영역만을 최종적인 후보영역으로 선택한다. 이러한 조건을 만족하는 후보영역이 둘 이상일 경우 영역간 NEARNESS를 비교하여 상용을 하는데 예를 들어 추적대상 차량의 진행방향은 아는 것으로 가정할 수 있기 때문에 만약 아래에서 위 방향으로 진행하는 차량을 추적한다면 두 프레임간 sy와 ey의 차는 반드시 - 또는 0이 되어야 하며 두 프

레이미간 차량의 위치가 거의 동일 차선에 있기 위해서는 sx와 ex의 차는 작아야 한다.

현 프레임 내의 영역 A에 대해 다음 프레임에 상기 조건을 만족하는 후보영역 B, C가 있을때 NEARNESS(A, B)와 NEARNESS(A, C)중 그 값이 큰 후보영역과 상용하게 된다. 위와 같이 NEARNESS가 정의된 배경은 차량의 운행속도와 운행방향은 영상 표본 간격에 비해 급격히 변하지 않는다고 가정할 수 있기 때문이다.

IV. 실험 결과

표 1. 차량의 진행 방향 및 차선인접 여부에 따른 조건
Table 1. Conditions needed according to the direction of vehicle and juxtaposition of car lanes

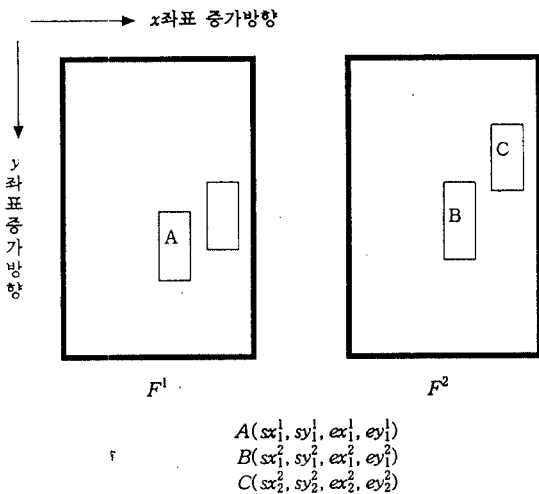
	진행 방향		차 선	
	아래→위	위→아래	동 일	인 접
$sx_1^n - sx_k^{n+1}$.	.	<<	>
$sy_1^n - sy_k^{n+1}$	- 또는 0	+ 또는 0	.	.
$ex_1^n - ex_k^{n+1}$.	.	<<	>
$ey_1^n - ey_k^{n+1}$	- 또는 0	+ 또는 0	.	.

<< : 두 지점간 차가 작음. > : 두 지점간 차가 큼.
 . : 후보 영역간 관계에 영향이 없음.

데이터를 얻기 위해서 실험실내에 모의 실험장치를 구성하여 수동으로 차량을 옮기면서 4개의 연속된 프레임을 획득했다.(그림 2참조) 카메라 입력으로부터 표본 추출된 이미지의 크기는 128×120이고 화소당 24bit 칼라 깊이를 갖는다. 원래의 영상프레임들은 칼라이나 본 논문에서는 농담영상으로 출력되어 있어 칼라가 나타나질 않았는데 차량의 칼라와 진행방향은 그림 3과 같다. 차선 1에는 진행 차량이 없고 차선 2에는 위에서 아래로 진행하는 두대의 적색 차량이 있고 차선 3에는 아래에서 위로 진행하는 흰색 차량이 있고 차선 4에는 아래에서 위로 진행하는 청록색 차량이 있다. 중앙선과 좌우 경계는 노란색이며 점선 형태의 분리선은 흰색이다.

FCM알고리즘은 사용하기 전 시스템에 초기화해야 할 상수값이 있는데 본 실험에서는 클러스터의 개수 $c=4$ 개, 가중치 지수 $m=2$, 최대 허용 오차 임계값은 $1.0e^{-3}$, norm metric은 EUCLIDEAN distance를 사용하였다. 그림 3을 보면 모두 5개의 칼라가 존재하므로(흰색-차량, 점선 분리선, 빨간색-차량, 청록색-차량, 노란색-중앙선, 좌우경계, 흑색-도로 표면) 클러스터 갯수 c 를 5로 해야하지만 클러스터 갯수에 대한 사전지식이 없는 것으로 가정하기 때문에 $c=4$ 로 두고 클러스터 collapse방법을 적용하였다. FCM알고리즘의 결과는 최종 수렴한 퍼지 membership matrix에 255값을 곱한 256단계의 농담영상으로 표현하였으며 이를 이치화한 것이 그림 4에 나타나 있다.

동일 column에 있는 것이 같은 클러스터를 나타내며 모두 5개의 클러스터가 나타나 있다. 첫 번째 클러스터는 도로중앙선과 보도와의 경계선에 해당하는



$$NEARNESS(A, B) = (|sx_1 - sx_2| + |sy_1 - sy_2| + |ex_1 - ex_2| + |ey_1 - ey_2|)^{-1}$$

그림 1. NEARNESS 정의
Fig. 1 Definition of NEARNESS

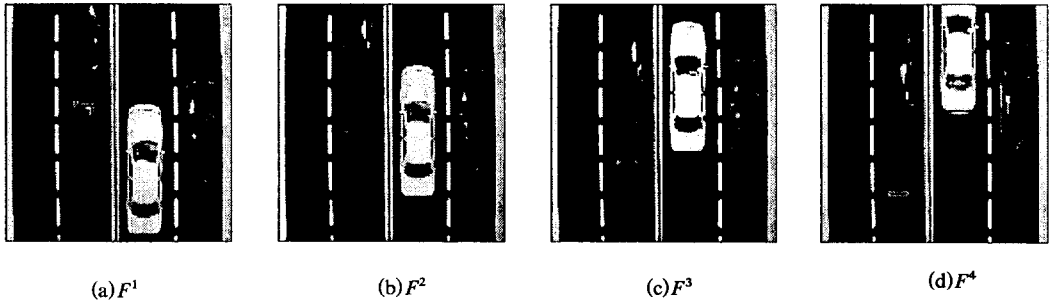


그림 2. 입력 영상열
Fig. 2 Input image sequence

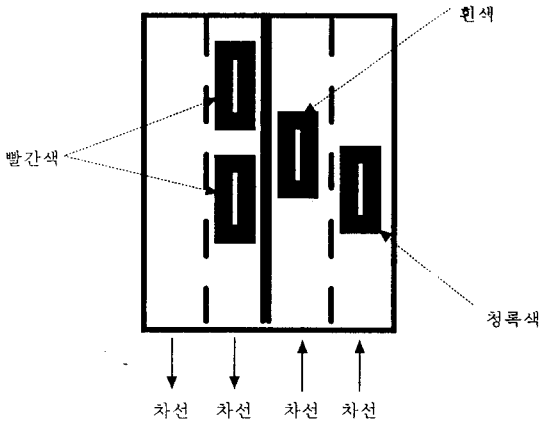


그림 3. 차량 칼라와 이동 방향
Fig. 3 Color and moving direction of vehicles

노란색부분, 두 번째 클러스터는 도로 표면, 세 번째는 흰색차량과 흰색분리선, 네 번째는 빨간색차량, 다섯 번째는 청록색차량을 나타내고 있다. 그림 4의 각 클러스터를 보면 차량에 해당하는 영역 외에 도로분리선, 중앙선, 차량내 유리창과 조명반사에 의한 군소 영역들이 존재해 있는데 이들로부터 차량에 해당하는 영역만을 취해야 한다. 이를 위해 우선 morphological operation을 통해 군소 영역을 없애고 영역 경계선을 smooth하게 만든다.

Morphology를 통해 어느 정도 이상크기의 경계선이 smooth하게 된 영역들만 남으면 각 영역을 감싸는 외접사각형을 구해 차량의 일반적인 모습과 유사한지를 조사하고 그것을 만족하는 영역들만을 상용의

대상으로 삼는다. 여기서 중앙선, 분리선 등은 사라지게 된다.(그림 5참조)

그림 5에서 첫 번째와 두 번째 클러스터의 경우는 차량으로 간주할 만한 영역이 전혀 없어 추적에 불필요한 부분이며 세 번째와 네 번째 클러스터의 경우는 매 프레임당 한 개의 영역만이 존재 그대로 상용관계가 형성되며 다섯 번째 클러스터의 경우만 상용과정을 필요로 하는데 그 결과가 Table 2에 나타나 있다. 여기서 영역번호는 scanline order대로 붙여졌으며 프레임 경계선과 이동차량간 occlusion과 disocclusion이 병존하는 상황에서도 프레임간 정확히 상용된 결과를 보여주고 있다. 상용과정을 필요로 하는 2개 이상의 클러스터를 얻기위해 그림 6과 같이 새로운 영상 시퀀스를 잡았다. 차량의 색깔 및 운행방향은 그림 7에 설명되어 있다. 여기에는 노란색차량과 빨간색차량이 각각 2대씩 있고 청록색 차량이 1대 있다. 앞에서 한 바와 같이 이치화, 형태학처리, 모양검사를 거친후의 클러스터모양이 그림 8에 나타나 있다. 세 번째 클러스터는 노란색 차량, 네 번째 클러스터는 빨간색 차량, 다섯 번째 클러스터는 청록색 차량에 해당하며 첫 번째와 두 번째 클러스터는 각각 도로 바닥과 차선 분리선에 해당하는데 모양검사를 통해 해당하는 것이 없으므로 아무것도 나타나지 않는 상태이다. 세 번째와 네 번째 클러스터에 대한 상용결과가 테이블 3에 나타나 있다. 첫 번째 영상 시퀀스에서와 마찬가지로 완벽한 상용결과를 보여주고 있다.

V. 결 론

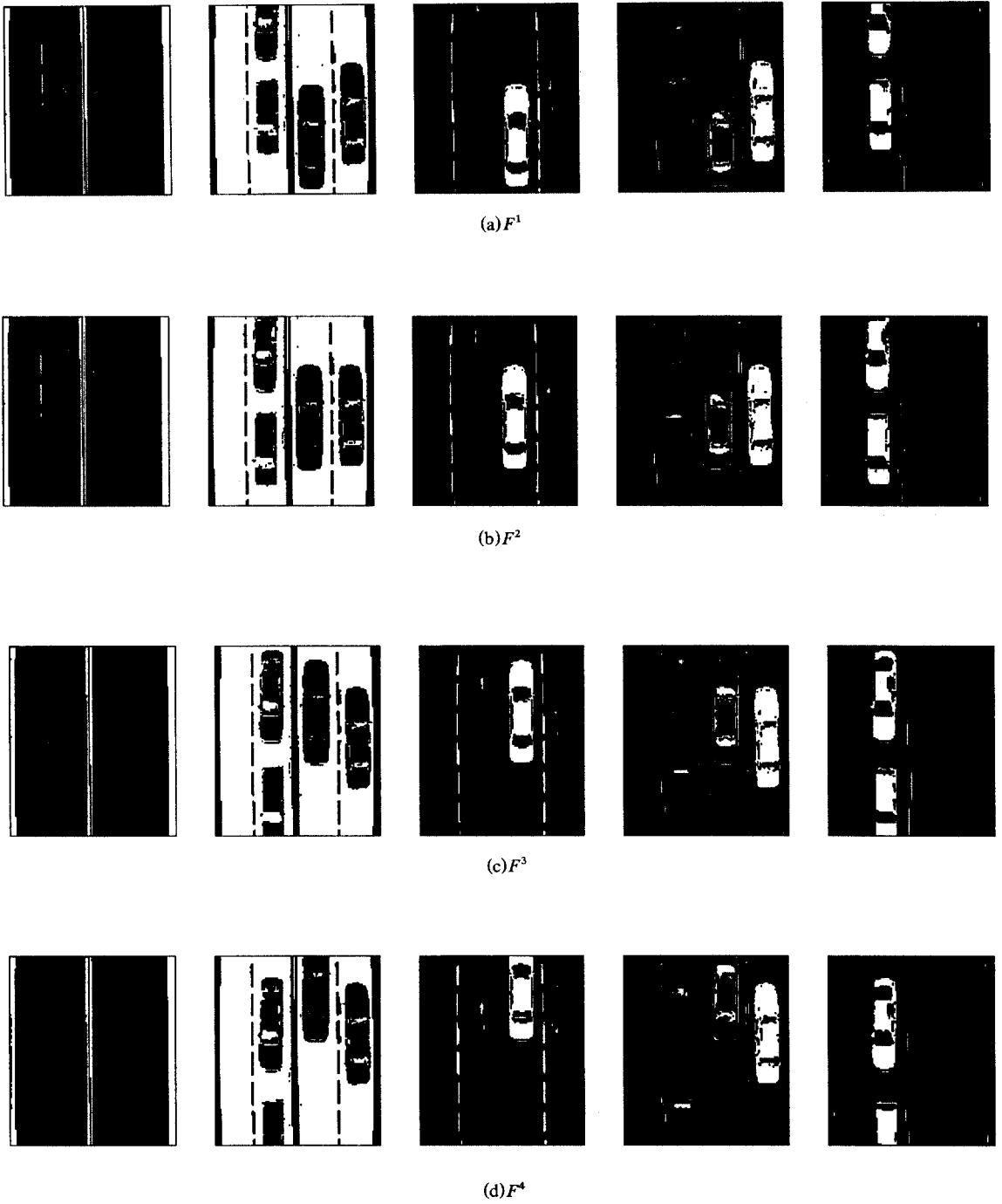
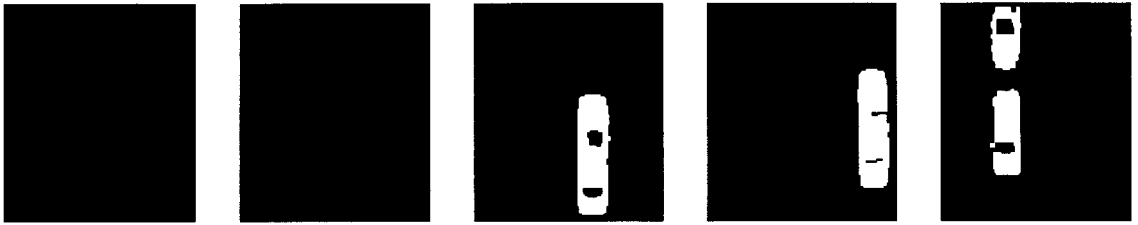
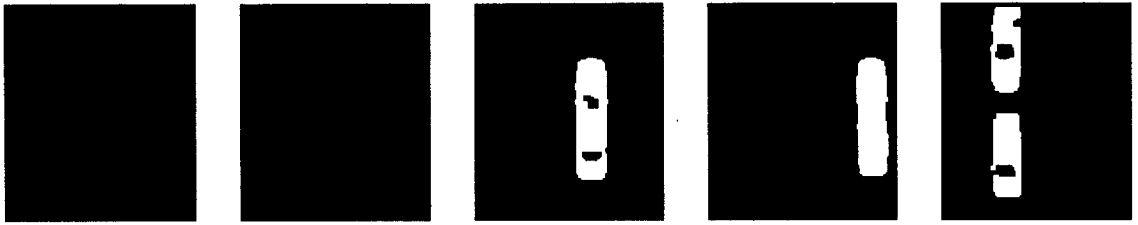


그림 4. 이치화 후의 클러스터별 영상열
Fig. 4 Clusterwise image sequence after binarization



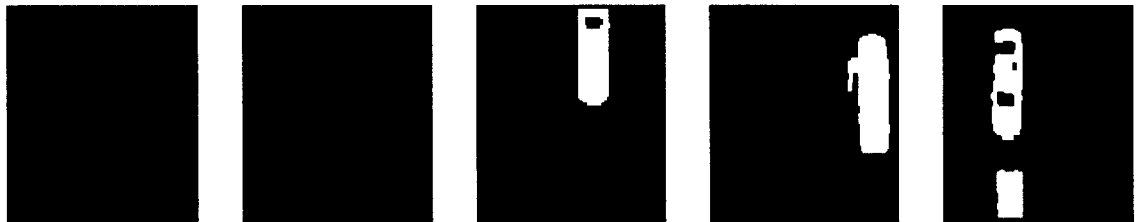
(a) F^1



(b) F^2



(c) F^3



(d) F^4

그림 5. 형태학 및 모양 검사를 통해 관련 없는 부분을 제외한 클러스터별 영상열

Fig. 5 Clusterwise image sequence after removing irrelevant parts using morphology and shape test

표 2. 5번째 클러스터에 대한 영역상응 결과
Table 2. Region correspondence result for fifth cluster

$F^1 \backslash F^2$	영역 1	영역 2
영역 1	1/15	1/132
영역 2	1/90	1/27

(a) F^1 과 F^2 사이

$F^2 \backslash F^3$	영역 1	영역 2
영역 1	1/14	1/144
영역 2	1/109	1/27

(b) F^2 과 F^3 사이

$F^3 \backslash F^4$	영역 1	영역 2
영역 1	1/23	1/150
영역 2	1/109	1/20

(c) F^3 과 F^4 사이

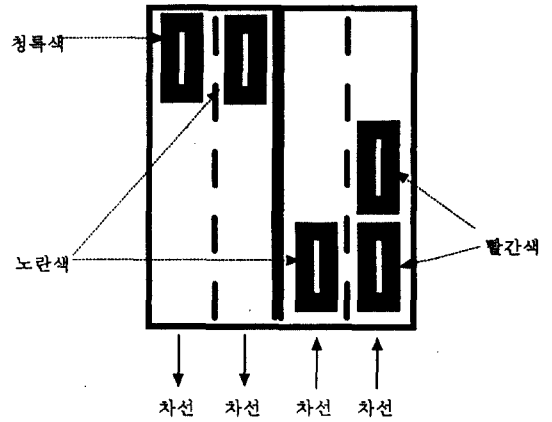


그림 7. 차량 칼라와 이동 방향
Fig. 7 Color and moving direction of vehicles

표 3. 3, 4번째 클러스터에 대한 영역상응 결과
Table 3. Region correspondence result for third, fourth clusters

노란색

$F^1 \backslash F^2$	영역 1	영역 2
영역 1	1/8	1/44
영역 2	1/59	1/7

(a) F^1 과 F^2 사이

$F^1 \backslash F^2$	영역 1	영역 2
영역 1	1/7	1/48
영역 2	1/59	1/5

(b) F^2 과 F^3 사이

$F^1 \backslash F^2$	영역 1	영역 2
영역 1	1/5	1/48
영역 2	1/60	1/7

(c) F^3 과 F^4 사이

빨간색

$F^1 \backslash F^2$	영역 1	영역 2
영역 1	1/18	1/74
영역 2	1/72	1/15

(a) F^1 과 F^2 사이

$F^1 \backslash F^2$	영역 1	영역 2
영역 1	1/22	1/42
영역 2	1/39	1/20

(b) F^2 과 F^3 사이

$F^1 \backslash F^2$	영역 1	영역 2
영역 1	1/14	1/29
영역 2	1/36	1/15

(c) F^3 과 F^4 사이

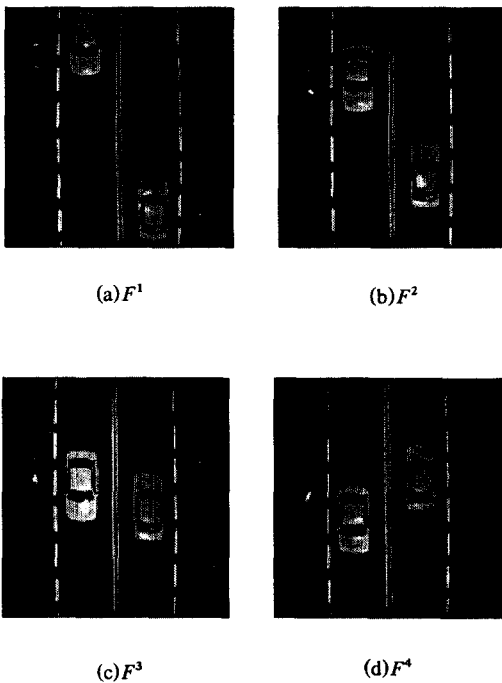
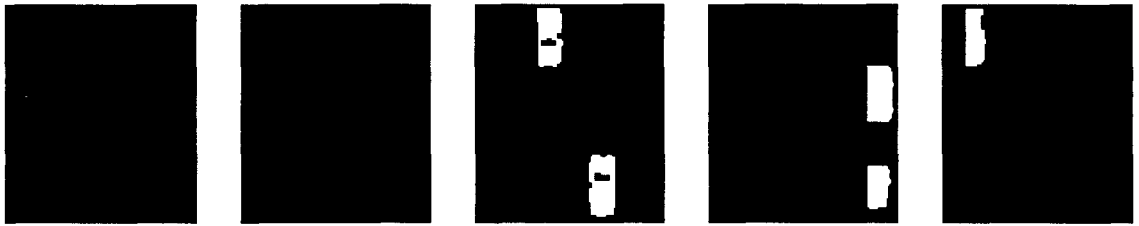
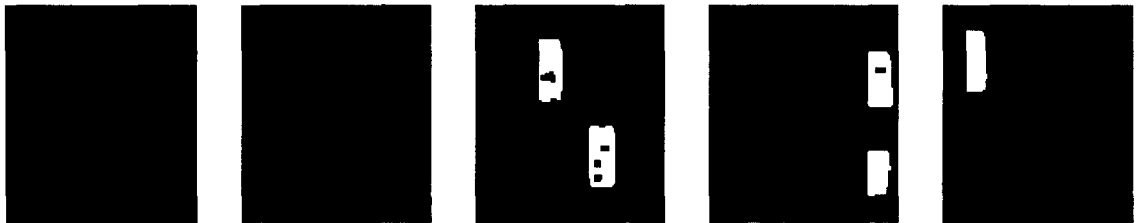


그림 6. 입력 영상열
Fig. 6 Input image sequence

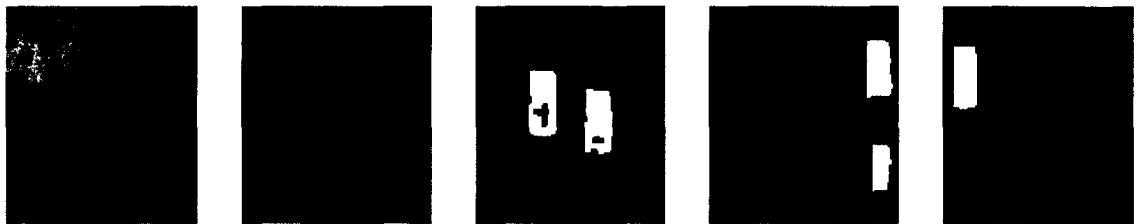
이동차량 추적은 차량평균속도 측정을 위해 필수적으로 요구되는 단계로써 여러 가지 가정하에 여러 방법들이 제안되었다. 본 논문에서는 클러스터 collapse 기능을 갖는 FCM에 기반을 둔, 칼라 정보를 이용한



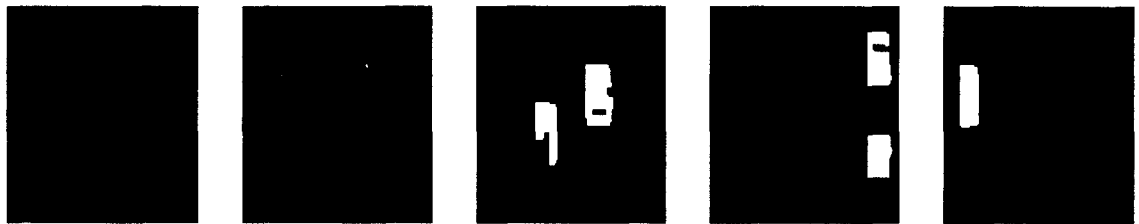
(a) F^1



(b) F^2



(c) F^3



(d) F^4

그림 8. 형태학 및 모양 검사를 통해 관련 없는 부분을 제외한 클러스터별 영상열

Fig. 8 Clusterwise image sequence after removing irrelevant parts using morphology and shape test

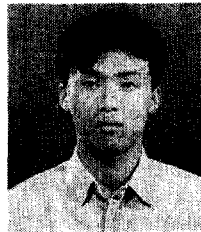
이동차량 추적 방법을 제안하였고 프레임 경계선과 occlusion과 disocclusion현상을 보이고 있는 실험 영상에서 좋은 결과를 보였다. 이를 실제 시스템에 응용하기 위해서는 실세계 영상에 대한 시험이 필요하고 또한 시스템 성격상 실시간 시스템의 설계가 필수적인데 현재 사용하고 있는 FCM알고리즘의 경우 많은 양의 칼라정보에 대해 비슷한 내용을 반복 수행함으로써 인해 소요시간이 많다. 이의 개선을 위해 AFCM (approximate fuzzy c-means) 알고리즘[4]이 사용될 수 있으며 이를 사용할 경우 약 1/6정도로 소요시간을 S/W적으로 단축가능하며 병렬 H/W구축으로 실시간에 가까운 성능을 얻을 수 있을 것으로 예상된다.

참 고 문 헌

1. Marie-Pierre Dubuisson and Anil K.Jain. 2D Matching of 3D Moving Objects in Color Outdoor Scenes. IEEE Trans. PAMI, pages 887-891, 1994.
2. N. Hoose and L.G. Willumsen. Automatically extracting traffic data from videotape using the CLIP4 parallel image processor. Pattern Recognition Letters 6, pages 199-213, 1987.
3. T. L. Huntsberger, C. L. Jacobs and R. L. Cannon. Iterative Fuzzy Image Segmentation. Pattern Recognition Vol. 18, No. 2, pages 131-138, 1985.
4. Robert L. Cannon, Jitendra V. Dave, and James C. Bezdek. Efficient Implementation of Fuzzy c-Means Clustering Algorithms. IEEE Trans. PAMI-8, No. 2, pages 248-255, MARCH 1986.
5. H. J. Zimmermann. Fuzzy Set Theory and Its Applications. Kluwer-Nijhoff Publishing, pages 187-212, 1985.
6. Michael J. Swain and Dana H. Ballard. Color Indexing. International Journal of Computer Vision, 7:1, pages 11-32 1991.
7. Daniel P. Huttenlocher, Gregory A. Klanderman, and William J. Rucklidge, Comparing Images Using the Hausdorff Distance, IEEE Trans. PAMI, Vol. 15, No. 9, pages 850-863, SEPTEMBER 1993.



양 상 규(Sangkyu Yang) 정회원
1972년 8월 21일생
1995년:명지대학교 정보통신공학과 학사
1996년:명지대학교 정보통신공학과 석사 재학중



이 정 재(Jungjae Lee) 정회원
1973년 4월 2일생
1996년:명지대학교 정보통신공학과 학사
1996년:명지대학교 정보통신공학과 석사 재학중



소 영 성(Youngsung Soh)정회원
1956년 3월 4일생
1978년:서울대학교 공업교육학과 학사
1986년:미 Univ. of South Carolina 전산학과 석사
1989년:미 Univ. of South Carolina 전산학과 박사
1989년 9월~1991년 2월:시스템공학연구소 선임연구원
1991년 3월~현재:명지대학교 전기전자공학부 부교수
※관심분야:영상처리, 비전, 인공지능