

교통감시영상에서 SURF 알고리즘을 이용한 차량추적시스템

김상기, *한동석
경북대학교

sanggi0430@gmail.com, *dshan@knu.ac.kr

A Vehicle Tracking System using SURF Algorithm in Vision-based Traffic Surveillance

SangGi Kim *Dong Seog Han

School of Electronics Engineering, Kyungpook National University

요 약

본 논문에서는 교통 감시 시스템에서 차량추적방법을 제안한다. 교통 감시 카메라를 이용한 차량추적시스템은 차량 감시, 사고감지 및 교통정보를 확인할 수 있게 하는 시스템이다. 차량추적을 위하여 먼저 가우스 혼합 모델(Gaussian Mixture Model)을 이용하여 배경과 전경을 분리하고 형태학적 필터링을 이용하여 차량을 검출한다. 검출된 차량으로부터 SURF(Speed Up Robust Features) 매칭을 통하여 차량추적방법을 제안한다.

1. 서론

일반도로 및 공공장소에서 CCTV 는 우리 실생활에 광범위하게 사용되고 있다. 특히 일반도로에서 교통감시영상을 이용한 차량추적시스템은 차량 감시, 사고감지 및 교통정보를 확인할 수 있게 하는 시스템이다. 그러나 교통감시영상을 이용한 차량추적시스템은 주변환경변화, 시점변화 등으로 인해 정확히 차량을 추적 하기 어렵다. 이러한 문제점을 해결하기 위하여 본 논문에서는 주변환경변화에 강인한 배경모델링 방법인 가우스 혼합 모델[1]을 이용하여 배경과 전경을 분리한다. 배경과 전경이 구분된 영상으로부터 형태학적 필터링을 이용하여 차량을 검출하게 되고 검출된 차량영상에 SURF(Speed Up Robust Features) [2] 알고리즘을 이용하여 두 차량영상에 대한 특징점을 검출하고 두 영상의 매칭을 통하여 차량을 추적하는 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2 장에서는 제안하는 차량추적방법에 대해 설명하고 3 장에서는 제안한 방법의 성능을 실험을 통하여 확인한다. 마지막으로 4 장에서는 본 논문에 대한 결론을 맺는다.

2. 다중 차량 추적 방법

영상에서 정확한 차량검출은 차량추적시스템에서 중요한 기능이다. 본 논문에서는 차량을 검출하기 위하여 가우스 혼합 모델을 이용하여 배경과 전경을 분리하여 차량의 후보영역을 검출한다. 가우스 혼합 모델은 여러 개의 영상에 대해 가우스 함수로 모델링하고 다음을 이용하여 각 화소의 확률을 측정한다.

$$P(X_i) = \sum_{i=1}^K w_{i,t} \cdot \eta(X_i, \mu_{i,t}, \Sigma_{i,t}) \quad (1)$$

여기서 K 는 단일 가우스 분포의 개수를 나타내며 본 논문에서는 3 을 적용하였다. $w_{i,t}$ 는 t 프레임에서의 i 번째 가우스 모델의 가중치, 평균 $\mu_{i,t}$ 와 공분산 $\Sigma_{i,t}$ 가 있다. η 는 가우스 확률 밀도 함수이다. 가우스 혼합 모델에서 배경에 해당하는 가우스 모델은 객체에 비해 가중치가 높고 작은 분산을 가지는 특성이 있다. 이를 이용하여 가우스 모델들을 w_k / σ 값에 대해 내림차순으로 정렬하고 다음의 조건을 만족하는 모델을 배경으로 결정한다.

$$B = \arg \min(\sum_{k=1}^b \omega_k > Th) \quad (2)$$

배경모델링으로부터 얻은 영상을 이진 영상으로 변환하여 레이블링을 한다. 레이블링 적용 후 영상의 잡음 제거 및 차량검출을 위하여 형태학적 필터링을 적용한다. 일반적으로 차량의 영역은 다른 영역보다 화소 수가 많으며 가로 세로의 비율이 비슷하다. 이 점을 이용하여 다음과 같은 두 조건을 만족하면 검출된 객체를 차량 후보로 판단한다.

$$O_{area} > T_1 \quad (3)$$

$$R = \frac{O_{width}}{O_{height}} < T_2 \quad (4)$$

여기서 O_{area} , O_{width} , O_{height} 는 각각 검출된 객체의 화소수, 폭,

높이이다. 본 실험에서는 $T_1 = 7000, T_2 = 1.2$ 로 설정하였다.



그림 1 차량검출결과영상

형태학적 필터링을 통하여 최종적으로 차량만 남은 영상으로부터 SURF 를 이용하여 차량의 특징점을 검출하고 검출된 특징점을 비교하여 동일한 차량인지 아닌지 판단하며 차량을 추적한다. t 번째 프레임 (F^t)에서 얻은 추적 대상 영역을 P^t 라고 하고 차량추적은 다음과 같이 P^{t-1} 과 P^t 사이의 SURF 매칭 [6]을 통하여 새로운 추적대상 P^t 를 얻는 방법으로 차량을 추적한다

$$SURF(P^{t-1}, P^t) = P^t \quad (5)$$

식 (5)는 차량의 수가 증가하게 되면 계산시간도 같이 증가하게 된다. 따라서 여러 대의 차량을 추적할 때에는 계산시간을 줄이기 위하여 매칭전 P_i^{t-1} 과 P_i^t 의 유클리드 거리가 가장 짧은 두 영상을 구한다. 유클리드 거리는 다음과 같다.

$$d(p, q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2} \quad (6)$$

유클리드 거리가 가장 작은 두 영상은 식(5)를 이용하여 매칭을 하게 되고 매칭된 특징점의 수가 미리 설정한 임계값을 넘게 되면 동일한 차량이라 판단하고 차량을 추적한다.

3. 실험

본 논문에서는 일반 도로에서 사용되는 CCTV 를 이용하여 촬영된 영상을 사용하였고 영상의 해상도는 1080x1980 이며 초당 11 프레임의 컬러영상이다. 그림 2 와 그림 3 은 본 논문에서 제안한 방법을 적용한 실험 결과 영상이다. 실험을 통하여 동일한 차량이 매칭될 경우는 특징점의 매칭수가 많고 서로 다른 차량이 매칭될 경우는 특징점의 매칭수가 작은 것을 알 수 있다.

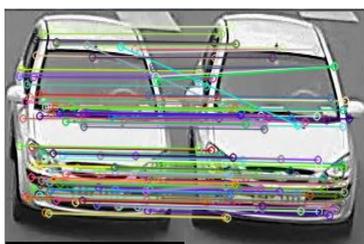


그림 2 동일한 차량의 매칭결과



그림 3 서로 다른 차량의 매칭결과

4. 결론

본 논문에서는 입력된 영상으로부터 배경모델링을 하여 전경과 배경을 분리한 뒤 형태학적 필터링을 이용하여 차량을 검출하였고 차량을 검출한 영상으로부터 두 차량의 SURF 매칭을 통하여 두 차량간의 매칭수가 가장 많은 것을 추적하는 방법을 제안하였다. 매 프레임마다 SURF 매칭을 하여 두 차량의 특징점 매칭의 수가 많은 것을 스택에 저장하기 때문에 정확도가 높지만 처리속도는 느리다. 향후 연구로서, 알고리즘의 경량화를 통하여 계산속도를 개선할 예정이다. 또한, 배경모델링 영상에서 겹침이 발생하게 되면 관심영역을 구분하는 것에 대해 연구하고자 한다.

감사의 글

"본 연구는 미래창조과학부 및 정보통신기술진흥센터의 ICT 융합고급인력과정지원사업의 연구결과로 수행되었음" (IITP-2015-H8601-15-1002)

참고 문헌

- [1] C. Stauffer and W.E.L. Grimson, "Adaptive background mixture models for real-time tracking" In: CVPR, pp. 246- 52 (1999)
- [2] H. Bay, T. Tuytelaars, and Luc Van Gool, "SURF: Speeded Up Robust Features", Lecture Notes in Computer Science pp 404-417
- [3] N. Friedman and S. Russel, "Image Segmentation in Video Sequences: A Probabilistic Approach", Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence (UAI' 97), pp. 175-181, 1997.
- [4] ZHOU Dan and HU Dong, "A Robust Object Tracking Algorithm Based on SURF" Wireless Communications & Signal Processing(WCSP), 2013 International Conference on
- [5] 한세진, "적외선 영상에서 객체 탐색을 위한 적응적인 GMM 방법" 전남대학교 석사학위논문, 2015년 2월
- [6] 유주희, "SURF 와 배경제거를 결합한 다수 사람 추적 알고리즘" 덕성여자대학교 석사학위논문, 2015년 2월