

템플릿 모델을 이용한 차량 종류 인식 방법

이정화*, 김태형*, 황영철*, 차의영*

*부산대학교 컴퓨터공학과

e-mail:junghwa@pusan.ac.kr

Vehicle Model Recognition using Template Models

Jung-hwa Lee*, Tae-hyung Kim*, Young-chul Hwang*, Eui-young Cha*

*Dept of Computer Engineering, Pusan National University

요 약

본 논문은 템플릿 모델을 이용하여 차량의 외관에 따른 종류를 인식하는 방법을 제안한다. 우선, 영상에서 차량을 검출하기 위하여 누적 차영상 기법을 이용하여 배경 영상을 추출한 후 차량 영역을 획득한다. 획득한 차량 영상은 날씨와 조명 영향에 따라서 그림자가 존재할 수 있다. 따라서 외곽선을 추출하고 가로와 세로, 대각선 방향으로 사영한 결과를 이용하여 그림자를 제거한다. 그림자 영역이 제거된 최종 차량 영역은 템플릿 모델과의 매칭을 통하여 가장 적합한 차량 종류로 인식한다. 제안된 방법을 이용하여 차량 종류를 인식하였을 때 만족할 만한 성능을 나타내는 것을 실험을 통하여 확인하였다.

1. 서론

도로상에는 방법, 단속, 측정 등을 위하여 수많은 카메라가 설치되어 있다. 각 카메라에 녹화되는 영상의 양은 방대하며, 특정 정보를 확인하기 위해서 많은 노력이 요구된다. 만약 특정 영역 주변에 지나간 승용차를 확인하기 위해서는 영역 내에 존재하는 모든 카메라를 일일이 사람이 확인해야 하는 어려움이 존재한다.

위와 같은 노력을 줄이기 위하여 본 논문에서는 영상 처리 및 분석 기술을 이용하여 차량을 검출하고 차량의 종류를 인식하는 방법은 제안한다. 일반적으로 도로상에 설치된 카메라는 고정되어 있기 때문에 움직이는 차량을 검출하기 위하여 차영상 기법을 많이 이용한다. 하지만 날씨나 조명의 변화에 따라 성능이 좌우될 수 있으며, 움직이는 속도에 따라서 정확한 차량 영역의 추출이 어려운 단점이 존재한다. 이를 해결하기 위하여 가우시안 혼합 모델(Gaussian Mixture Models) 또는 누적 차영상 기법 등을 이용하여 차량을 검출한다 [1].

차영상을 통하여 검출된 차량 영역에는 그림자도 포함될 수 있으며, 그림자는 인식 단계에서 오류를 일으킬 수 있는 요소이다. 따라서 그림자를 제거하는 단계가 필요하며 HSI 컬러 공간으로 변환 후 밝기 값의 변화를 감지하는 방법, 신경망을 이용한 방법 등을 이용하여 그림자를 제거한다.

최종 차량 후보 영역은 차량 외관에 따라서 분류되며, 분류 방법으로는 SVM(Support Vector Machine)을 이용하는 방법, 3D 모델을 이용하는 방법, 템플릿 매칭 등이 있다 [2][3][4][5]. 각 분류 방법은 사용되는 특징에 따라 달라질 수 있으며, 그 특징으로는 외곽선(edge), 윤곽

(contour) 등이 사용될 수 있다.

본 논문에서는 실시간 환경에서 차량 검출 및 종류 인식이 가능하기 위하여 다음과 같은 과정을 수행한다. 우선 영상이 입력되면 차량 후보 영역을 추출하기 위하여 누적 차영상 기법을 이용한다. 추출된 차량 후보 영역에서 그림자를 제거하기 위하여 외곽선 영상으로 변환하고 가로와 세로, 대각선 방향으로 사영(projection)을 수행하여 외곽선 성분이 적은 영역을 제거한다. 이렇게 획득한 최종 차량 영역은 5개 형태, 6개 방향을 가지는 30가지의 템플릿 모델과 매칭을 수행하여 가장 높은 유사도를 가지는 차량 종류로 분류된다.

2. 차량 검출

2.1 누적 차영상을 이용한 차량 후보 영역 추출

영상에서 이동 중인 객체를 검출하는 방법은 여러 가지가 있지만 그 중에서 차영상이 실시간 영상처리에 많이 사용한다. 기본적인 차영상은 이전 영상 프레임과 현재 입력되는 영상 프레임의 차를 구하여 픽셀의 변화를 검출한다. 그러나 이러한 연속 프레임의 차영상은 잡음(noise)에 민감하며 시간의 변화에 비하여 위치의 변화가 작은 이동 객체를 검출하기에는 힘들다. 또한 이동 객체의 전체가 아니라 두 프레임간의 변화된 부분만 검출이 가능하기 때문에 차량 후보 영역을 추출하기에 부적합하다.

이를 보완하기 위하여 연속 프레임간의 차이를 이용한 차영상이 아니라 배경영상과 현재 프레임의 차이를 구하는 차영상 방법이 있다. 즉, 카메라의 위치와 촬영되는 면이 고정인 점을 이용하여 이동 객체가 없는 화면을 배경 영상으로 등록하는 방법이다. 그러나 이 방법도 조명과 날

씨에 따른 외부적인 조건에 영향을 받게 되며, 배경 영상의 등록시점도 결과에 많은 영향을 주는 단점이 존재한다.

따라서 위와 같은 단점을 보완하면서 차량 후보 영역을 추출하기 위하여 누적 차영상(Accumulative Difference Pictures) 방법을 사용한다 [6].

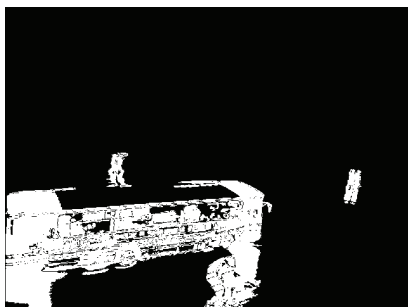
$$ADP_t = (I_t + ADP_{t-1} * k) / (k + 1) \quad (1)$$

위의 식 1을 보면, t 시간의 누적 배경영상은 t-1 시간의 누적 배경영상에 누적된 영상의 수 k를 곱하고, 현재 프레임에 더한 후 증가된 영상의 수 k+1을 나누어서 구한다. 이렇게 획득한 영상은 배경영상으로 사용이 되며, 현재의 영상과 차영상을 구하면 이동 객체의 영역이 추출된다. 차영상의 결과는 픽셀 값의 차이 정도보다는 차이의 유무가 더 의미 있으며, 미세한 변화까지 검출할 필요는 없기 때문에 이진화를 수행하여 차량의 움직임과 같은 큰 변화만 검출하도록 한다. 누적 차영상 방법은 입력되는 맵 프레임마다 배경영상이 수정되기 때문에 시간적, 공간적 변화에 적응적으로 적용이 가능하며 작은 위치의 변화도 쉽게 감지할 수 있다. 아래의 그림 1은 누적 차영상을 이용하여 획득한 배경영상 및 차영상을 나타내며, 차량의 형태가 어느 정도 나타나는 것이 확인된다.



(그림 1) 누적 차영상을 이용한 배경영상(좌)과 차영상(우)

그러나 차영상을 통해 획득한 영상은 배경영상과의 픽셀 차이를 나타내므로 완전한 차량의 형태를 나타내지 않는다. 따라서 템플릿 모델과 매칭을 하기 위하여 윤곽선을 추출한다. 윤곽선은 일정거리 이내에 있는 픽셀들을 같은 그룹으로 묶으며, 그룹 내에 존재하는 홀(hole)은 가득 채워 차량 전체를 포함할 수 있는 윤곽선을 획득한다. 그림 2는 차영상에서 윤곽선을 추출한 영상이며, 차량 및 주변의 잡영과 그림자가 포함되어 있는 것을 확인할 수 있다.



(그림 2) 윤곽선 추출 영상

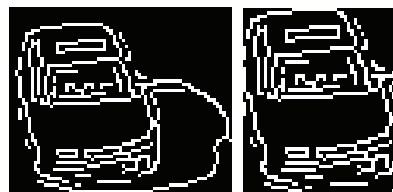
2.2 그림자 제거

차영상의 결과를 이용하여 추출한 윤곽선은 차량 영역뿐만 아니라 차량의 그림자의 영역도 포함한다. 차영상은 단순히 픽셀 값의 차이를 이용한 것이기 때문에 그림자도 이동 객체로 추출되는 것이다. 그림자는 태양광과 카메라의 방향에 의해 일정하게 생기는 것이 아니며 크기도 다르므로 쉽게 제거하기 힘들다.

영상처리에서 사용되는 일반적인 그림자 제거 방법으로는 컬러 공간을 변환하여 그림자 영역을 찾아내는 방법과 신경망을 이용하는 방법 등이 있다. 그림자의 유무에 따라 색도는 일정하고 밝기 값의 차이만 존재한다는 점을 바탕으로 밝기 값만 변화된 영역을 그림자로 추출하는 방법이 있다. 즉 RGB 컬러 공간을 HSI, YCbCr 등의 컬러 공간으로 변환하여 그림자 영역을 찾아낸다. 그러나 밝기 값의 변화를 이용하는 방법은 배경영상에 대한 의존도가 높고 누적 차영상의 경우는 적용하기 힘들다는 단점이 존재한다. 신경망을 이용한 방법은 그림자 영역의 샘플 데이터를 학습한 이후, 현재의 픽셀이 그림자 영역인지 아닌지를 분류한다. 이 방법도 샘플 데이터의 의존도가 높으며 장소에 따라서 그림자의 색상 정보가 다르기 때문에 신뢰도가 떨어진다.

본 논문은 그림자의 색상 정보가 아닌 외곽선 정보를 이용하여 그림자를 제거하였다. 그림자는 단색으로 이루어진 덩어리이기 때문에 윤곽선 영역 내에 포함된 그림자의 외곽선은 차량 영역의 외곽선에 비해서 그 성분이 약하다. 즉, 윤곽선 내의 외곽선 정보를 추출하여 약한 성분을 가지는 영역을 제거하는 방법으로 그림자를 제거한다.

우선 외곽선 영상을 얻기 위하여 Canny edge detection을 사용한다. Canny edge detection은 다른 방법에 비하여 잡음에 덜 민감하며 강한 외곽선 정보를 얻을 수 있다. 외곽선이 추출된 차량 후보 영역은 가로, 세로, 대각선의 4 방향 사영(projection)을 수행한다 [6]. 가로 방향의 사영을 통해 위쪽과 아래쪽에 존재하는 그림자를 제거하고, 세로 방향의 사영을 통해 왼쪽과 오른쪽에 존재하는 그림자를 제거한다. 또한 오른쪽 위에서 왼쪽 아래, 왼쪽 위에서 오른쪽 아래로 두 방향의 대각선 사영을 통해 대각에 존재하는 그림자까지 제거를 한다. 외곽선이 존재하는 후보 영역에 사영을 수행하여 양끝에서부터 사영된 값이 갑자기 증가하는 지점을 찾고, 그 지점을 차량의 실제 영역으로 지정한다. 아래의 그림 3은 사영을 이용하여 그림자를 제거한 결과이다.



(그림 3) 사영을 이용한 그림자 제거

3. 차량 종류 인식

3.1 템플릿 모델

템플릿 영상은 매칭 대상에 따라 컬러 영상, 그레이 영상, 외곽선 영상 등을 사용한다. 컬러 정보가 중요한 특징이라면 컬러 템플릿 매칭을 사용하는데 그 성능은 좋지 않다. 그레이 영상을 이용한 템플릿 매칭은 찾으려는 물체의 형태가 고정적이고 크기와 위치에 대한 변화가 있을 때 유용하다. 또한 외곽선 영상을 이용한 템플릿 매칭은 형태적인 정보가 중요한 특징일 때 자주 사용한다.

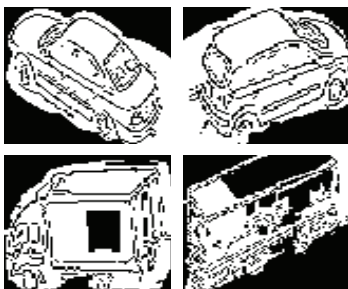
차량 영상의 경우, 차량이 다양한 색상과 형태를 가지기 때문에 컬러 영상과 그레이 영상을 이용한 매칭을 적용하기 어렵다. 그리고 형태적인 정보가 각 차량 종류를 분류하는데 중요한 정보가 되지만 단순히 외곽선 정보만으로 매칭을 하기엔 부족하다. 차량의 외곽선은 배경영역에 비하여 전체 면적에서 차지하는 비율이 낮으므로 대표 특징으로 사용하기에 부족하기 때문이다.

따라서 템플릿 매칭을 위한 템플릿 모델은 외곽선 정보와 윤곽선 정보를 합쳐서 사용한다. 윤곽선은 차량의 전체적인 픽셀의 분포를 나타내며 외곽선에 비해 많은 정보를 가진다. 또한 외곽선은 윤곽선의 형태가 비슷한 모델일 경우 차량이 가지는 내부적인 형태를 특징적으로 나타내어 준다. 따라서 윤곽선과 외곽선을 결합한 템플릿 모델은 매칭에 더욱 효과적이며, 생성과정은 아래와 같다.

- ① 원본 그레이 영상에서 외곽선 추출
- ② 외곽선 영상을 반전
- ③ 반전된 영상을 이진화
- ④ 반전 이진화 영상과 윤곽선 영상을 AND 연산

반전 이진화된 외곽선 영상과 윤곽선 영상을 AND 연산하면 차량 영역은 밝은 값을 가지고 차량 영역내의 외곽선 부분은 어두운 값을 가진다.

본 논문에서는 차량의 종류를 승용, RV, 트럭, 특장트럭, 버스에 해당하는 5가지 모델로 분류하고 좌측, 우측, 좌측전면, 우측전면, 좌측후면, 우측후면에 해당하는 6방향의 템플릿 모델을 생성하였다. 템플릿의 크기는 80×80이며 30개의 템플릿 모델을 생성하고 가장 상관도가 높은 차량의 종류를 인식하도록 한다. 아래의 그림 4는 템플릿 모델의 예시이다.



(그림 4) 템플릿 모델의 예시

3.2 템플릿 매칭

템플릿 모델과 입력 영상과의 매칭을 수행하기 위해서 먼저 입력 영상을 80×80 크기로 정규화 한다. 입력 영상은 그림자가 제거된 차량 후보 영역이다. 크기 정규화를 수행한 영상은 템플릿 모델과 동일한 방법으로 외곽선을 추출하고 반전 이진화를 수행하여 최종적으로 윤곽선 영상과 AND 연산된 영상을 획득한다. 이 영상과 30개의 템플릿 모델과 매칭을 수행하게 되며 상관 계수(Correlation coefficient)를 이용하여 상관도를 측정한다.

상관 계수는 두 x, y에 대하여 상관관계의 정도를 나타내는 계수로써 식 2를 사용한다.

$$Y = \frac{\frac{1}{N} \sum_{r=1}^N (x_r - \bar{x})(y_r - \bar{y})}{\sigma_x \sigma_y} \quad (2)$$

\bar{x} 와 \bar{y} 는 각각 x와 y의 평균값이고, σ_x 와 σ_y 는 x와 y의 표준편차이다. 상관계수는 $-1 \leq Y \leq 1$ 의 값을 가지며 서로 양의 상관관계에 있을 때는 $Y > 0$, 음의 상관관계에 있을 때는 $Y < 0$, 아무런 상관이 없을 때에는 $Y = 0$ 이다.

30개의 모델 중 가장 상관도가 높은 모델이 정해지면 해당 모델의 차량 종류에 따라 입력된 차량의 종류로 분류가 되며, 최종적으로 소형(승용, RV), 중형(트럭, 특장트럭), 대형으로 분류하였다.

4. 실험 결과

실험은 차량이 여러 방향에서 촬영될 수 있도록 다양한 위치에서 촬영한 영상을 대상으로 하였다. 영상은 20 프레임/초 로 입력되며 매 프레임마다 인식되는 차량의 종류에 대하여 성능을 평가하였다. 한 프레임에 대하여 처리시간은 37 ms 이며 이는 실시간 처리에 만족할 만한 성능으로 평가된다. 아래의 그림 5는 실험에 사용된 영상의 일부분이다.



(그림 5) 실험 데이터 캡처 화면

실험 결과, 각 차량 영역에 대하여 표 1과 같은 결과를 나타내었다. 실험은 한 프레임 내에 포함된 차량이 어느 차종으로 분류가 되는지 판단하였고, 실험 결과 중형의 경우 인식률이 50%를 조금 넘는 수준이며 소형의 경우는

70%가 넘는 인식률을 보인다. 일반적으로 차량 종류 인식 성능은 실험 환경이 고정적일 경우 80~90%의 인식률을 보이며, 고정적이지 않은 환경에서는 약 70%의 성능을 보인다.

중형 차량이 다른 차량에 비해 낮은 성능을 보이는 이유는 특장 트럭의 윤곽선이 소형 트럭의 윤곽선과 비슷한 형태를 가지기 때문으로 분석된다. 또한 전체적으로 차량 외에 여러 사람이 지나가거나 태양광에 의한 밝기 변화로 차량을 잘못 검출한 경우 때문에 성능이 저하되었다. 그러나 인식률에 비하여 처리 속도는 실시간 처리를 만족할 만한 성능이며, 템플릿 모델의 개선과 객체 추적 등의 기술을 더한다면 인식률의 향상도 가져올 수 있을 것이다.

<표 1> 차량 종류 인식의 성능평가 결과

모델 차량	소형	중형	대형	인식률 (%)
소형	2308	615	174	74.5
중형	55	83	16	53.8
대형	49	61	222	66.8

5. 결론

본 논문에서는 도로상에 설치된 카메라의 영상에서 차량을 검출하고 차량의 종류를 인식하는 방법을 제안하였다. 누적 차영상을 이용하여 차량을 검출하고, 태양광으로 인하여 생기는 그림자를 가로, 세로, 대각선 방향으로 사영 통해 제거하였다. 또한 차량의 윤곽선과 외곽선을 결합한 템플릿 모델과 매칭을 통하여 차량의 종류를 인식하였는데, 실시간 환경에서 동작할 수 있을 정도로 프레임 당 37 ms의 성능을 보여주었다.

전체적인 인식률은 65% 로써 양호한 수준이며, 앞으로 차량 검출 부분의 성능을 보완한다면 더 높은 인식률을 기대할 수 있을 것으로 생각된다. 또한 프레임 각각의 인식 결과를 이용하는 것이 아니라 차량 이동에 대한 추적을 통해서 차량의 이동 방향에 대한 정보를 추가적으로 활용하고, 인식된 차량의 정보도 유지하는 방법을 적용한다면 실제 시스템에도 적용이 가능할 것이라고 기대한다.

참고문헌

- [1] N.Buch, J.Orwell, S.A.Velastin, Detection and Classification of Vehicles for Urban Traffic Scenes, VIE2008, pp182-187, 2008
- [2] W.Wu, Z.QiSen, W.Mingjun, A Method of Vehicle Classification Using Models and Neural Networks, Proceedings of the IEEE Conference on Vehicular

Technology Conference, Vol.4, pp.3022-3026, 2001

[3] A.Goyal, B.Verma, A Neural Network based Approach for the Vehicle Classification, CIISP 2007, pp.226-231, 2007

[4] T.N.Tan, G.D.Sullivan, K.D.Baker, Model-Based Localisation and Recognition of Road Vehicles, International Journal of Computer Vision, Vol.27, Issue 1, pp.5-25, 1998

[5] G.D.Sullivan, A.D.Worrall, J.M.Ferryman, Visual Object Recognition Using Deformable Models of Vehicles, Workshop on Context-Based Vision, pp.75-86, 1995

[6] R.Jain, R.Kasturi, B.G.Schunck, "Machine Vision", McGraw-Hill, 1995