DBpia

논문 17-42-06-15 The Journal of Korean Institute of Communications and Information Sciences '17-06 Vol.42 No.06
https://doi.org/10.7840/kics.2017.42.6.1225

히스토그램 데칼코마니 정합을 이용한 영상검지기용 차량 인지 알고리즘

조 상 일◆, 이 재 성°

Vehicle Detection Algorithm for VDS by Using Décalcomanie Matching Based on Histogram

Sang-Il Jo◆, Jaesung Lee°

요 약

본 논문에서는 영상검지기(VDS)를 위한 기존 차량 인지 방법들이 높은 연산복잡도 때문에 실시간 처리가 어렵다는 문제점을 극복하기 위하여 히스토그램 기반의 데칼코마니 정합 알고리즘을 제안한다. 이 방법은 정면에서 차량을 바라보고 정확히 반으로 나누었을 때, 양쪽이 대칭적이라는 형태적 특징으로부터 착안한 것이다. 전체 알고리즘은 관심영역 설정, 배경제거를 통한 객체검출, 모폴로지, 히스토그램 데칼코마니 정합의 4단계로 구성된다. 고안된 알고리즘은 OpenCV 프로그래밍을 통해 구현하였으며 소형 미니 PC 플랫폼에서 실행되었다. 그 결과 25fps@720p의 실시간 처리 성능, 좋은 날씨에서 평균 97.2%의 차량인지률, 평균 9.58%이하의 중심 좌표 오차율의 성능을 보였다.

Key Words : vehicle detection system, traffic volume, computational complexity, real-time processing

ABSTRACT

In this paper, we propose a histogram-based decalcomania matching algorithm to overcome the problem that conventional vehicle recognition methods for VDS (Vehicle Detection System) are difficult to be processed in real time due to high computational complexity. When we look at a vehicle from the front and divide it into exactly half, both halfs are symmetrical. This paper focuses on this point. The whole algorithm consists of four steps: setup of ROI (Region of Interest), object detection through background subtraction, morphology, and histogram-based decalcomania matching. The devised algorithm is implemented by OpenCV programming and executed on a small mini-PC platform. The performance evaluation results show that the proposed algorithm has run-time performance of 25fps@720p, vehicle recognition rate of 97.2% on average, and center coordinate error rate of average 9.58% or less under good weather.

Ⅰ. 서 론

최근 활발히 연구되고 있는 차세대 지능형교통시스템(C-ITS, Cooperative Intelligent Transport Systems)은 교통정보의 수집과 제공이 시간적, 공간적으로 분리되어 있는 기존 지능형교통시스템(ITS)과 달리 차

※ 본 연구는 미래창조과학부 및 정보통신기술진흥센터의 대학ICT연구센터육성 지원사업(IITP-2017-2013-0-00680)의 연구결과로 수행되었습니다.

◆ First Author : Korea National University of Transportation Department of Electronic Engineering, lupin715@naver.com, 학생회원
° Corresponding Author : Korea National University of Transportation Department of Electronic Engineering, jaesung.lee@ut.ac.kr, 정회원

논문번호 : KICS2016-12-414, Received December 30, 2016; Revised March 27, 2017; Accepted May 12, 2017

www.dbpia.co.kr

량과 차량(V2V), 차량과 도로 인프라(V2I) 간 실시간 정보공유를 통해 언제 어디서나 신속하고 능동적으로 원활한 교통 흐름과 안전사고 예방을 위한 다양한 교통정보제공이 가능하다. 이러한 C-ITS에서 가장 중요한 역할을 담당하는 것 중에 하나가 바로 영상검지기(VDS, Video Detection System) 이다[1]. VDS 는 종전의 루프검지기와 지자기 센서의 단점들(도로 파손과 그에 따른 설치 및 유지 보수 비용 증가) 뿐만 아니라 레이저와 레이다 검지기들의 단점들(고가의 장비 비용)을 극복할 수 있는 가성비가 뛰어난 장비로 알려져 있다. 하지만, VDS 도 영상처리 알고리즘의 연산복잡도 증가에 따른 하드웨어 비용 증가 및 전력소모 증가로 폭넓은 저변 확대에는 한계를 가지고 있다.

본 연구에서는 저연산량의 영상처리 알고리즘을 개발하여 소형, 저전력, 저가의 모바일 임베디드 플랫폼에도 구현이 가능하도록 함으로써 현재 루프 검지기나 VDS 가 설치되지 않은 이면도로들까지 도로 구석구석에 카메라를 설치하여 도로 전구간의 주요 교통정보(교통량 등)를 수집, 제공하는데 도움이되고자 한다.

영상을 이용한 교통정보 수집에서 가장 중요한 요소는 차량으로 의심되는 이동 객체를 1대의 완전한 차량으로 정확히 인지하는 문제이다. 지금까지는 모션 히스토리(Motion History), SIFT (Scale-Invariant Feature Transform), 히스토그램 역투영(Back-projection)을 이용한 Mean Shift, 머신 러닝 알고리즘 등을 사용하여 차량을 인지해왔다. 모션 히스토리 알고리즘은 인접하는 프레임들의 차연산을 통해 각각의 결과 이미지에서 임계치 이상의 값들을 이진화한 뒤, 그 결과 이미지들을 짧은 시간 동안 겹쳐 나타내어 이동객체의 움직임을 추정하는 알고리즘이다. 하지만 차량 정체 등으로 긴 시간동안 영상 프레임이 멈춰있을 경우 이동객체로 인지하지 못한다는 단점이 있다[2]. SIFT 알고리즘의 경우 특징점을 기반으로 프레임간 객체 매칭을 통하여 차량을 인지하기 때문에 시간에 상관없이 차량을 잘 인지할 수 있다. 하지만, 처리할 연산량이 너무 많아 실시간 처리가 불가능하다. 경우에 따라는 한 프레임 계산에 2~3초가 수행되기도 한다. 뿐만 아니라 사전에 차량 모델 이미지들을 미리 입력해줘야하기 때문에 불특정 다수의 차량을 검출하기에는 제한적이다[3]. 히스토그램 역투영 기법을 사용한 Mean-Shift 은 색상정보(Hue Channel)히스토그램을 이용해 입력영상에서 유사도를 측정하고 그 결과영상의 픽셀 값을 확률 값으로 변경시켜 구한 확률 값 분포에 대해 Mean-Shift를 수행한다. 하지만 SIFT 와 마찬가지로 사전 정보를 요구하기 때문에 불특정 다수의 차량을 검지하는 것은 는 불가능하다[4]. 한편, 최근에 머신 러닝을 이용한 차량 검지 방법도 많이 연구되고 있는데 SIFT 와 마찬가지로 연산복잡도가 너무 커서 실시간 처리는 불가능하다[5].

본 논문에서는 기존 알고리즘들의 높은 연산복잡도(run-time 처리 불가능) 또는 사전에 약속된 차량만을 검지하는 문제점을 극복하기 위하여 히스토그램 기반의 데칼코마니 정합 알고리즘을 제안한다. 정면에서 차량을 바라볼 때, 차량을 정확히 반으로 나누면 좌우가 동일하다는 형태적 특성에 착안하여 본 연구를 수행하였으며 차량을 반으로 나누어 획득한 좌우영상의 유사도를 보다 정확히 측정하기 위해 히스토그램 비교기법을 사용하였다.

본문 2장에서 제안하는 히스토그램 데칼코마니 정합을 이용한 차량검출기법을 설명하고, 3장에서 제안하는 알고리즘의 성능분석을 수행한다. 마지막으로 4장에서 결론을 맺는다.

Ⅱ. 히스토그램 데칼코마니 정합을 이용한 차량검출

제안하는 영상검지 시스템의 전체적인 알고리즘 수행과정을 그림 1에 나타내었다. 도로 인프라 카메라로부터 입력되는 영상에 대해 전처리(Preprocessing) 과정을 통해 도로영역을 정의함으로써 도로 이외 영역의 잡음으로 인한 영향을 차단한다. 이후 입력 프레임

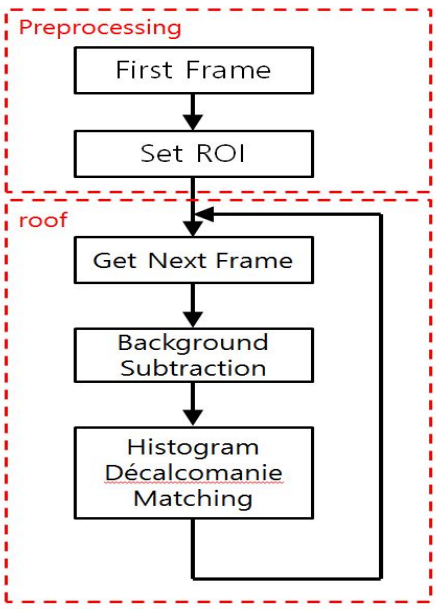


그림 1. 제안하는 알고리즘 흐름도
Fig. 1. Algorithm Flow chart for the proposed system

www.dbpia.co.kr

들을 누적하여 평균배경영상을 생성하고 배경영상과 현재입력프레임간의 차 연산(Difference Operation) 수행하여 도로 위의 이동 객체들을 검출한다. 이후 각 객체를 별도의 이미지 윈도우로 분리한 다음 그 윈도우 안에서 히스토그램 기반 데칼코마니 정합 과정을 통해 유사도를 계산한다. 마지막으로 유사도 값을 분석하여 차량의 진위여부를 판단한다.

2.1 관심영역(ROI, Region Of Interest)설정

본 논문이 제안하는 차량검출 알고리즘의 첫 번째 과정은 입력되는 프레임에서 관심영역을 설정하는 것이다. 고정된 카메라를 통해 입력되는 프레임에서 차량의 검출이 이루어지는 도로영역을 제외한 영역은 이후 단계에서 수행될 연산에 잡음의 역할을 한다. 따라서, 도로영역만을 관심영역으로 설정하여 영역외부 요인에 의한 잡음과 오차를 최소화 한다.

관심영역설정은 아래 그림 2와 같다. 입력되는 첫 번째 프레임에서 정의 영역 좌표들을 입력받아 내부 픽셀 값이 255, 외부픽셀 값이 0인 이진화(Image Binarization)된 다각형 이미지(그림 2의 좌측하단)를 생성하여 관심영역을 정의한다. 그 다음 다각형 이미지와 현재 입력되는 프레임 간의 논리(AND) 연산을 통해 다각형 내부영역만을 획득한다.

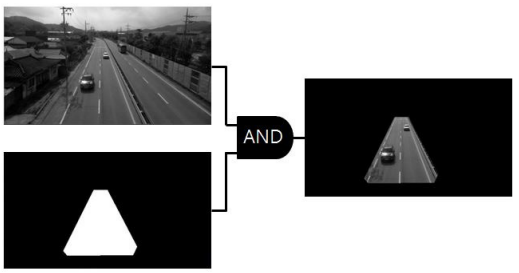


그림 2. 객체검출이 수행될 관심영역 설정(우측 결과이미지)
Fig. 2. Setting the region of interest in which object detection will be performed (right result image)

2.2 배경제거를 통한 객체검출

차량검출에 있어서 도로 위 이동하는 객체(Moving Object)들은 차량으로 의심되는 중요한 정보이다. 주로 사용되는 이동객체검출 알고리즘으로는 배경제거(Background Subtraction), 모션 히스토리(Motion History), 광류(Optical Flow)[6] 등이 있으나 본 논문에서는 실시간 처리가 가능하도록 비교적 연산량이 적고 구현이 간단한 배경제거기법을 사용하여 움직이는 도로 위 객체를 검출하였다[7]. 배경제거기법을 통한 객체검출 방법은 가장 간단하면서도 효과적인 움직임 검출 방법으로, 배경이미지와 현재 입력프레임 사이 차 연산(Difference Operation)을 계산하고, 그 결과가 일정 임계 값 이상을 초과하면 움직임이 있는 객체로 인지하는 방식으로 수행된다.

아래 식 1은 입력되는 연속적인 프레임을 일정 가중치로 누적하여 배경이미지((Background Image)를 획득하는 수식(running average 계산식)을 나타내고 있다.

$$\mu_t = \alpha I_t + (1-\alpha)\mu_{t-1} \tag{1}$$

위 식에서 I_t 는 현재 프레임 t에서 입력된 영상의 픽셀 값, μ_t는 t까지의 평균 픽셀 값, 그리고 α는 가중치로 새로이 입력된 영상I_t 가 이전까지 평균값 μ_{t-1}에 얼마만큼 영향을 주는지에 대한 값으로, 적절한 이동평균배경을 얻기 위해서 적절하게 설정해 주어야 한다. 예를 들어 200프레임의 영상을 누적하여 배경을 획득할 경우 α의 값은 0.005의 값이 된다.

입력되는 프레임에서 아래 식 2를 사용해 움직임이 발생하는 픽셀을 계산할 수 있다. 현재 입력프레임과 배경이미지간 차 연산(Difference Operation) 결과에

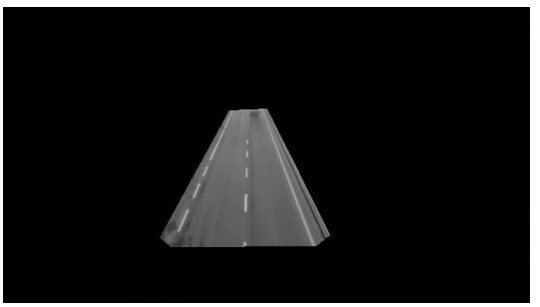


그림 3. Running average를 이용한 배경이미지
Fig. 3. Background image using running average

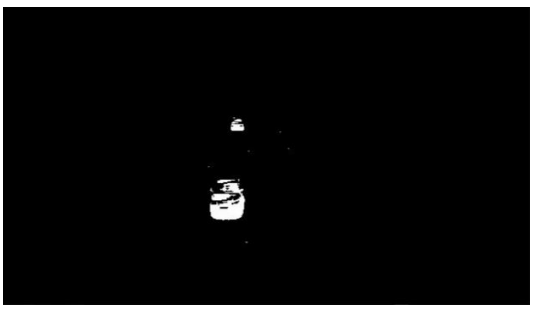


그림 4. 배경제거를 이용한 객체검출 결과
Fig. 4. Result of object detection using background subtraction

www.dbpia.co.kr

절대값을 취해 일정 임계치와 비교를 통해 이진화(Image Binarization)를 수행한다. 결과적으로 입력되는 프레임 내 움직이는 객체에 해당하는 픽셀들을 해당 위치의 배경 픽셀들과의 차이를 계산하여 일정 수준 이상의 극심한 값의 변화를 보이는 픽셀들에 대해 이진화를 하여 이동 객체 부분을 추출할 수 있다.

$$|I_t - \mu_t| > threshold \tag{2}$$

2.3 모폴로지(Morphology)

전단계에서 얻은 이진화된 객체영상은 배경영상과 객체간 픽셀 밝기 차이가 임계치를 넘지 않아 하나의 객체가 여러 덩어리로 분할되는 현상이 일어날 수 있다. 따라서 본 과정에서는 객체 분리현상을 억제하기 위해 모폴로지 연산을 적용한다.

분리된 객체덩어리를 하나의 객체로 합병(Merge)하기 전, 전처리 작업으로 소금후추 잡음(salt and pepper noise)을 제거하기 위해 미디안 필터링을 먼저 수행한다. 미디안 필터링은 대표적인 비선형 필터링으로서 백색(픽셀 값이 255)과 흑색(픽셀 값이 0)에 가까운 잡음을 제거하는데 효과적으로 사용된다.

잡음을 제거한 뒤 분할된 객체 덩어리의 병합을 위한 모폴로지 연산을 본격적으로 수행한다. 아래 그림 6과 같이 차량으로 의심되는 이동객체는 도로를 정면으로 바라보도록 설치된 카메라가 촬영하여 이진화하면 수평 방향으로 분할되는 현상이 종종 발생한다. 따라서, 수직방향으로 좀 더 긴 마스크(Structuring Element)를 정의하여 수직방향으로 더 많은 불림(Dilation)이 되도록 할 필요가 있다. 본 논문에서는 8×2 의 마스크를 정의하여 사용하였다. 이런 식으로 마스크의 크기를 경험적으로 정해야하면 차량과 카메라간 거리(원근)에 따라 마스크 인위적으로 조작해줘야 하는데 일반적으로 VDS 의 카메라가 교통량 측정을 위해 차량을 인지할 때는 차량을 정면으로 바라보고 촬영하다가 차량이 특정 기준 지점(ROI)을 통과할 때 인지하기 때문에 카메라와 피사체간 거리가 일정하게 되므로 최초 카메라 설치시만 인위적으로 결정해주면 된다. 또한, 일반적으로 카메라 설치 지점으로부터 멀지 않은 일정 거리에 그 기준 지점을 지정하므로 설치 환경이 다를지라도 마스크 모양과 크기를 정하는 튜닝 시간은 그리 많이 걸리지 않게 된다.

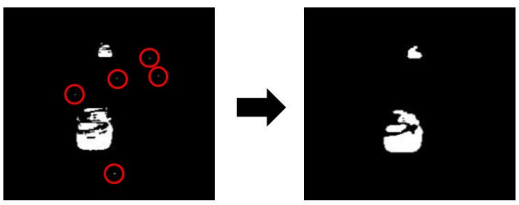


그림 5. 미디안필터링을 이용한 소금잡음(원으로 표시) 제거
Fig. 5. Removing salt noises (marked with circles) through median filtering

$$f \oplus M = \bigcup_{X \in M} f_X \tag{3}$$

모폴로지 불림 연산은 식 3과 같이 수행된다. 입력영상 f의 객체 부분(white 영역)의 모든 픽셀 위치로 마스크 M의 중심좌표가 지나다니면서 마스크 M 이 덮치는 부분은 모두 white 픽셀로 전환하는 과정이다. 결과적으로 차량으로 의심되는 객체를 마스크 크기만큼 외부로 팽창시켜 끊어져있던 요소들을 상호 연결해주는 역할을 한다.

그런 다음 침식연산을 통해 팽창연산 후 외부로 팽창한 객체를 다시 깍아 원래 크기로 복원시킨다. 아래 식 4는 침식연산식이다.

$$f \ominus M = X \,|\, M_{(X)} \subseteq f \tag{4}$$

아래 그림 6은 모폴로지 연산을 통해 분리된 객체의 요소들이 바르게 결합된 예를 보여준다.

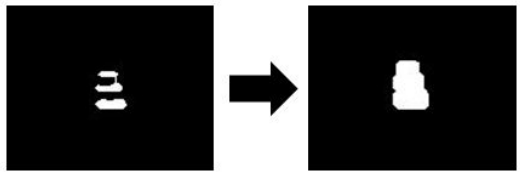


그림 6. 모폴로지 연산을 이용한 분리된 객체의 결합
Fig. 6. Combining separated objects into a single chunk using morphology operations

2.4 히스토그램 데칼코마니 정합

본 논문에서는 도로 위 이동객체의 차량 여부를 판단하기 위해 히스토그램 데칼코마니 정합 기법을 제안한다. 히스토그램(Histogram)은 이미지의 특성을 분석한 대표적인 기법으로 이미지 내 각 픽셀 값들의 발생 빈도수를 나타낸다. 한편, 일반적인 차량은 정면 또는 후면에서 바라볼 경우 수직 중앙선을 중심으로 좌우가 대칭(Décalcomanie)이라는 시각적 특성을 가지고 있다. 따라서 전 단계에서 얻어진 이동객체의 윤곽선(coutour)으로 정의되는 직사각형 영역(그림 7 의

www.dbpia.co.kr

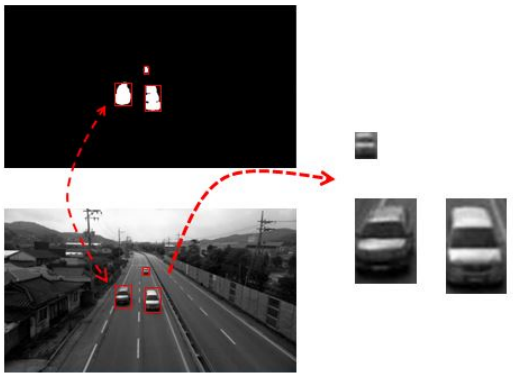


그림 7. 이동객체의 윤곽선으로 정의되는 의심 영역
Fig. 7. Region of suspicion defined by contours of moving objects

빨간색 사각형들)을 수직 중앙선을 중심으로 반으로 나눈 뒤 좌우 각 영역의 오리지널 픽셀값들을 가지고 히스토그램을 구해 상호 비교해보면 높은 유사도를 가짐을 확인할 수 있다. 이러한 높은 유사도 계수는 그 이동객체가 차량인 지 여부를 판단하기 위한 중요한 척도로 사용할 수 있다.

객체의 데칼코마니 정합 정도를 수치화하기 위해서는 객체를 수직으로 나누어 분할된 두 이미지의 히스토그램을 계산한 뒤 유사도를 계산해야 한다. 하지만 차량으로 추정되는 객체 영역(직사각형 영역)은 상황에 따라 그림자 성분을 포함하거나 차량이 바라보는 각도가 완벽한 정면이 아닌 경우가 많아 그 사각형의 중심과 차량의 중심은 일치하지 않는다. 따라서 그림 8과 같이 좌우 윈도우의 폭을 조금씩 증가시켜가며 슬라이딩시키면서 좌우 유사도를 계산하여 그 값이 가장 높은 지점에서 슬라이딩을 멈춤으로써 진정한 차량의 중심을 찾는다.

그림 8은 유사도 계산을 위한 두 영역 윈도우들이 슬라이딩하는 과정을 보여준다. 앞서 언급한 바와 같이 불필요한 그림자가 포함되어 있을 확률이 높으므로 히스토그램이 계산될 좌우 두 윈도우를 일정 가중치로 증가시켜가며 좌측에서 우측으로, 우측에서 좌측으로 양방향을 비교하면 그림자가 생성된 방향이 반대방향에 비해 낮은 유사도를 갖게 되어 그림자에 의한 오차를 제거할 수 있다.

유사도 측정을 위해 본 논문에서는 대표적인 히스토그램 비교 기법인 바타차야 매칭(Bhattacharyya matching) 기법의 유사도 계산 수식을 사용하였다. 바타차야 매칭은 히스토그램으로 표현된 데이터의 유사성을 측정하는 통계학적인 알고리즘으로 정규화가 수

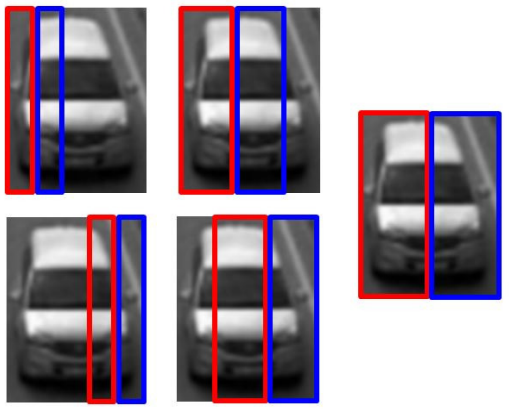


그림 8. 진정한 중심을 찾기 위한 윈도우 슬라이딩
Fig. 8. Window sliding to find the real center point

행된 히스토그램에서 식 6으로 정의된다.

$$d(H_1, H_2) = \sqrt{1 - \frac{\sum_i \sqrt{H_1(i) H_2(i)}}{\sqrt{\overline{H_1}\,\overline{H_2} N^2}}} \qquad (5)$$

위 식에서 $H_1(i)$와 $H_2(i)$는 이전 단계에서 분할된 객체의 좌우 이미지의 히스토그램으로 다시말해 단계별 각 휘도값(bin), 즉 i에 해당하는 휘도 값을 갖는 픽셀의 개수를 의미한다. $\overline{H_k}$는 히스토그램의 모든 bin의 값을 bin의 개수로 나눈 평균값으로 $\overline{H_k} = \frac{1}{N}\sum_j H_k(j)$ 으로 나타낼 수 있다. N은 히스토그램 bin의 수를 나타낸다. 이렇게 얻어진 비교계수 $d(H_1, H_2)$는 $0 \le d(H_1, H_2) \le 1$ 의 범위를 가지고 있으며 d의 값이 0에 가까울수록 두 히스토그램이 유사한 히스토그램임을 나타낸다. 완벽하게 일치하는 이미지일 경우 0의 값을 갖는다.

아래 그림 9는 지금까지 설명한 히스토그램 데칼코마니 정합 과정을 샘플 이미지에 적용해 계산한 유사도의 그래프를 보여준다. 그림 8에서 설명한 윈도우 슬라이딩을 수행하면서 각 크기의 윈도우마다 히스토그램 비교를 수행한다. 비교 결과 계수가 가장 낮게(가장 유사한 두 이미지) 나타나는 윈도우를 차량 정확한 절반으로 선정할 수 있다.

1,000회 이상의 차량 정면 이미지 샘플을 가지고 실험을 수행하였고, 0 ~ 0.376 범위의 유사도 계수에서 차량의 정중앙점을 발견할 수 있었다. 유사도 계수가 그 범위안에 있지 않을 때는 슬라이딩 스텝을 넓히

www.dbpia.co.kr

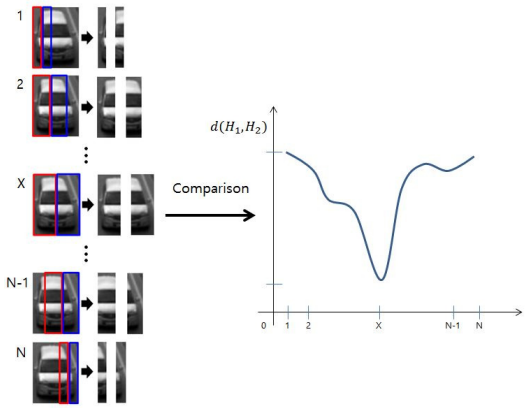


그림 9. 유사도 그래프 예(X에서 제일 작은 값을 가짐)
Fig. 9. Example of similarity graph (it has the smallest value at X)

고 그 범위에 들어오면 슬라이딩 스텝을 조금씩 증가시켜 연산 속도와 정밀도를 동시에 높이도록 하였다.

Ⅲ. 히스토그램 데칼코마니 정합 기법의 성능 분석

지금까지 설명한 히스토그램 데칼코마니 정합 알고리즘을 전문 영상처리 C++ 라이브러리인 OpenCV를 이용해 프로그래밍하여 구현하였고 실행 파일을 실행시켜 입력 영상에 대해 자동화된 처리를 수행하도록 하였다. 본 실행 파일은 Intel®CoreTM i5-6400T @2.2GHz, 4GB RAM, 그래픽 Intel® HD Graphics 530 의 사양을 갖는 소형 미니 PC 플랫폼에서 실행되었으며 720p HD 이미지에 대해 평균 25 frames per second 의 처리 성능을 보여 HD 급 영상에 대해서도 실시간 처리가 가능함을 확인하였다.

본 알고리즘의 성능 평가를 위해 실제 차량 영역의 중심 좌표와 본 데칼코마니 정합 알고리즘을 통해 자동 인지된 차량 영역의 중심 좌표간 차이(오차)를 계산하였다. 오차 거리의 측정 단위는 픽셀이며 오차 비율은 오차 픽셀수를 실제 차량의 영역으로 판단되는 영역의 폭(Width)으로 나눈 비율로 표기하였다. 또한, 인지률은 실제 차량 영역과 인지된 차량 영역간 공유 면적을 실제 차량 영역으로 나눈 비율로 표기하였다. 아래 그림 10은 맑은 날 낮 시간대에 카메라로부터 20~40m 거리에서 이동하는 차량에 대한 데칼코마니 정합 알고리즘을 적용한 결과 예제를 보여주며 표 1은 일정시간동안 도로를 촬영하면서 본 알고리즘을 적용하여 계산한 평균적인 오차와 평균 차량 인지률을 나타내고 있다.

표. 1 제안하는 알고리즘의 성능
Table 1. performance of the proposed algorithm

center coordinate mean error		average perception rate (%)
by pixel	ratio (%)	
4.624259	9.58	97.2

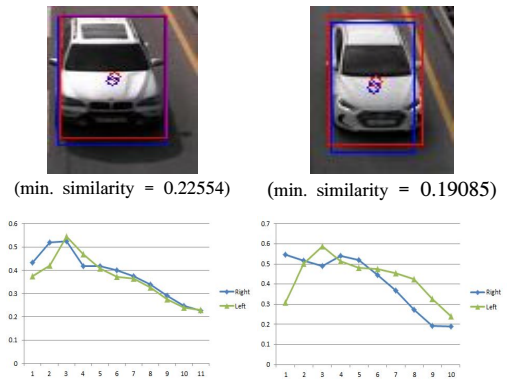


그림 10. 실험 결과(파란색 영역 : 제안하는 알고리즘이 인지한 차량의 영역, 빨간색영역 : 실제차량영역, 녹색 그래프 : 왼쪽 방향으로 윈도우 슬라이딩, 파란색 그래프 오른쪽 방향으로 윈도우 슬라이딩)
Fig. 10. Experimental results (blue box: vehicle area found by the proposed algorithm, red box: actual vehicle area, green graph: window sliding form right to left, blue graph: window sliding form left to right)

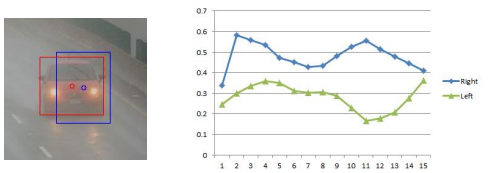
계산된 데이터로부터 알 수 있듯이 제안하는 알고리즘은 날씨가 맑은 날의 경우 비교적 매우 정확한 인지률과 적은 오차를 가짐을 알 수 있다. 한편 악천후 및 시간대별 인지률과 오차 역시 중요하기 때문에 여러 가지 상황에 대한 실험을 실시하였으며 그 결과를 그림 11과 표 2에 정리하였다. 예상대로 비가 오는 경우는 오차율이 높았으며 비가 오는 저녁시간대의 경우 그 비율이 급격히 증가함을 알 수 있다. 이는 도로 바닥의 빗물에 의해 반사되는 전조등이 매칭을 방해하기 때문인 것으로 확인이 되었다. 그에 따라 인지률

표. 2. 제안하는 알고리즘의 성능
Table 2. performance of the proposed algorithm

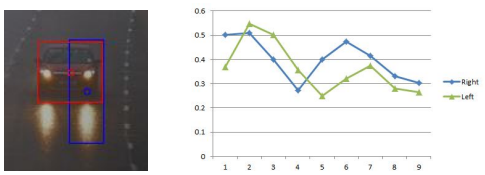
weather and time	center coordinate mean error ratio (%)	average perception rate (%)
rain, daytime	18.13	91.3
rain, nighttime	37.96	87.6
snow, daytime	6.43	98.9
snow, nighttime	12.40	95.5

www.dbpia.co.kr

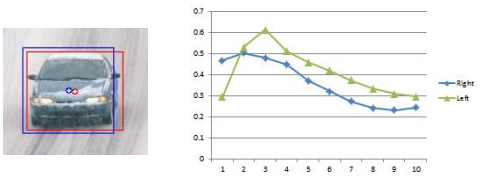
도 급격히 떨어진다. 반면 적당량의 눈이 오는 경우는 오히려 맑은 날씨 보다 인지률이 높았다. 이유는 바닥면이 하얗게 변하기 때문에 노면의 차선이나 그림자 등의 방해 요소들이 제거되고 배경 분리가 명확히 되기 때문이다. 눈오는 날 저녁의 경우는 정확도가 감소되는데 그대로 비오는 날 저녁에 비하면 양호한 것으로 나타났다.



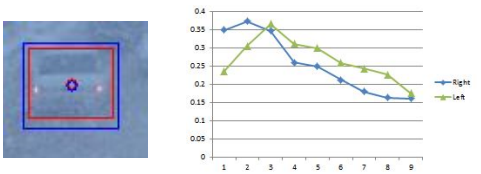
(a) rain, daytime (min. similarity = 0.16569)



(b) rain, nighttime (min. similarity = 0.24847)



(c) snow, daytime (min. similarity = 0.23155)



(d) snow, nighttime (min. similarity = 0.16106)

그림 11. 악천후에서의 실험 결과
Fig. 11. Experimental results under bad weather

Ⅳ. 결 론

본 논문에서는 차량의 인지를 위한 새로운 기법인 히스토그램 데칼코마니 정합기법을 제안하였다. 차량을 반으로 나누었을 때 좌우가 동일하다는 형태적 특성을 이용해 차량을 인지하였다. 성능분석을 통해 맑은 날씨의 경우 평균 97.2%의 차량인지률, 미니 PC 플랫폼에서 25fps@720p의 실시간 처리 성능과 9.58%이하의 중심 좌표 오차율의 성능을 가짐을 확인할 수 있다. 이는 SFIT 알고리즘을 사용하였을 때 평균적으로 90%의 인지률을 보인 기존 연구[9]보다 우월한 성능이다. 또한, 기존의 SIFT 알고리즘이 일반 데스크탑 PC (Intel Core i-3 3210 @ 2.3 GHz processor, 8 GB RAM)에서 640×478의 작은 영상에 대해 1.543초 당 한 장의 이미지를 처리하는 성능을 보인 것[8]에 비해 약 39배 빠르므로 충분히 실시간성을 확보한 것으로 판단된다.

한편 악천후에서는 다소 성능이 감소하는 것으로 나타났으나 평균적으로 90% 이상의 인지률을 보여 일반적인 영상검지기(VDS) 성능과 동등한 수준이다. 같은 맥락에서 본 논문의 히스토그램 데칼코마니 정합 알고리즘은 다소 인위적인 상황에서만 차량을 인지할 수 있다는 단점도 가지고 있다. 즉, 차량의 측면, 또는 좌우가 대칭이 아닌 특수 차량, 색상이 도로의 색상과 비슷한 차량 등은 인지를 하지 못한다. 하지만 기존 루프 검지기나 영상 검지기도 약 90% 수준의 인지률을 보이지만 현재 도로에 설치되어 사용되고 있다. 본 연구의 주요 목적은 그 인지률을 높이는 것보다는 경량 알고리즘을 개발하여 영상처리에 필요한 연산량을 최소화하여 실시간 처리가 가능하면서도 저가, 저전력 미니 모바일 플랫폼에 구현할 수 있도록 하여 이면도로까지 도로 구석구석에 설치하여 촘촘한 교통정보를 수집할 수 있도록 하는 것이다. 따라서, 교통 정보의 특성상 그러한 특수한 경우 단위 시간동안 차량 몇 대를 인지하지 못하더라도 수집된 교통 데이터의 신뢰도에 크게 영향을 주지는 않는다.

References

[1] K.-H. Kim, “Advanced road infrastructure management plan considering the paradigm change of intelligent transportation system,” *KRIHS POLICY Brief,* vol. 593, pp. 1-8, 2016.

[2] M. A. R. Ahad, J. K. Tan, H. Kim, and S. Ishikawa, “Motion history image: its variants and applications,” *Machine Vision Appl.*, vol. 23, no. 6, pp. 255-281, Mar. 2012.

[3] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.

[4] D. Comaniciu, V. Ramesh, and P. Meer, “Real-time tracking of non-rigid objects using mean shift,” *IEEE CVPR*, vol. 2, pp. 142-149,

www.dbpia.co.kr

2000.

[5] S. Kim, T. Z. Han, H.-J. Lee, and H.-J. Lee, "Car detection and recognition with Haar-like features and PCA," *2012 Int. Conf. Electron., Inf. and Commun.*, Feb. 2012.

[6] S. Heo and B. Moon, "Optical flow based object tracking using adaptive ROI," *IEIE Conf.*, pp. 403-406, Nov. 2014.

[7] M. Piccardi, "Background subtraction techniques: a review," *2004 IEEE Int. Conf. Syst., Man and Cybernetics*, vol. 4, pp. 3099-3104, 2004.

[8] P. M. Panchal, S. R. Panchal, and S. K. Shah, "A comparison of SIFT and SURF," *Int. J. Innovative Res. Comput. and Commun. Eng.*, vol. 1, no. 2, pp. 323-327, Apr. 2013.

[9] L.-C. Chiu, et al., "Fast SIFT design for real-time visual feature extraction," *IEEE Trans. Image Process.*, vol. 22, no. 8, pp. 3158-3167, 2013.

조 상 일 (Sang-Il Jo)

2011년 3월~2017년 2월 : 한국교통대학교 전자공학과 학사
<관심분야> 영상 처리, 멀티미디어 SoC 설계

이 재 성 (Jaesung Lee)

2008년 8월 : 서울대학교 전기컴퓨터공학부 박사
2001년 2월~2011년 8월 : 한국전자통신연구원(ETRI) 선임연구원
2011년 9월~2013년 9월 : 한국교통대학교 조교수
2013년 10월~현재 : 한국교통대학교 부교수
<관심분야> 영상 처리, 멀티미디어 SoC 설계

www.dbpia.co.kr