

온라인 학습을 이용한 비전 기반의 차량 검출 및 추적

길 성 호*, 김 경 환^o

Vision-Based Vehicle Detection and Tracking Using Online Learning

Sung-Ho Gil*, Gyeong-Hwan Kim^o

요 약

본 논문에서는 추적중인 차량의 외형 변화에 대해 온라인 학습 능력이 있는 비전 기반의 차량 검출 및 추적 시스템을 제안한다. 제안하는 시스템은 새로 검출된 차량의 연속된 프레임 간 움직임을 빠르고 강건하게 추정하기 위해 특징점 기반 추적 방법을 사용한다. 동시에 추적중인 차량에 대해 온라인 차량 검출기를 훈련시키고, 일시적인 차량 추적 실패 시 검출기의 결과를 이용해 추적기를 재초기화하여 강건한 추적을 가능하게 한다. 특히 차량 외형 모델의 업데이트 방법을 개선하여 시스템의 추적 성능을 높이고 처리시간을 단축시켰다. 다양한 주행환경에서 획득한 데이터셋을 사용하여 제안하는 시스템의 차량 검출 및 추적 성능을 평가하였다. 특히 우천 및 터널 통과와 같은 악조건에서 기존의 방법에 비해 차량 추적 성능이 상당히 개선된 것을 증명하였다.

Key Words : Vehicle detection, vehicle tracking, semi-supervised learning, bootstrapping, real-time

ABSTRACT

In this paper we propose a system for vehicle detection and tracking which has the ability to learn on-line appearance changes of vehicles being tracked. The proposed system uses feature-based tracking method to estimate rapidly and robustly the motion of the newly detected vehicles between consecutive frames. Simultaneously, the system trains an online vehicle detector for the tracked vehicles. If the tracker fails, it is re-initialized by the detection of the online vehicle detector. An improved vehicle appearance model update rule is presented to increase a tracking performance and a speed of the proposed system. Performance of the proposed system is evaluated on the dataset acquired on various driving environment. In particular, the experimental results proved that the performance of the vehicle tracking is significantly improved under bad conditions such as entering a tunnel and passing rain.

I. 서 론

해마다 교통사고로 인한 사망자 수는 사망 원인 통계에서 큰 비중을 차지한다. 운전자 및 보행자의 안전에 대한 사회적 요구가 증대됨에 따라, 자동차 안전 관련 기술도 더욱 발전되고 있다. 대표적으로 능동 안

전 시스템(active safety systems)과 첨단 운전자 보조 시스템(advanced driver assistance systems)이 있다. 이러한 시스템은 센서로부터 주변 환경에 대한 정보를 수집하고, 이를 바탕으로 사고 방지를 위한 다양한 기능을 제공한다. 그 중 운전자 보조를 위한 비전 기반의 차량 검출은 지난 이십년간 널리 연구되어온 주

* First Author : 서강대학교 전자공학과 Man Machine Interface 연구실, guilsungho@hanmail.net, 정회원

^o Corresponding Author : 서강대학교 전자공학과, gkim@sogang.ac.kr, 정회원

논문번호 : KICS2013-12-523, 접수일자 : 2013년 12월 3일, 심사일자 : 2013년 12월 12일, 최종논문접수일자 : 2013년 12월 16일

제이다^[1]. 초기의 접근법은 실시간 동작보다는 높은 검출 성능을 목표로 하였다. 이후 AdaBoost를 이용한 실시간 물체 검출 방법이 제안되면서^[2], 많은 연구들이 이를 차량 검출에 적용하였다. 최근에는 차량 검출 및 추적의 성능을 강건하게 하기 위해 카메라뿐만 아니라 레이더(radar), 라이더(lidar)와 같은 다양한 센서를 융합해 사용하는 추세이다. 하지만 비전 기반의 차량 검출은 여전히 많은 장점을 가지고 있다. 먼저 영상 해석을 통한 형태 분석은 주변 환경에 대해 많은 정보를 제공한다^[3]. 또한 비전 센서를 이용한 시스템은 비용이 저렴하고 구축이 쉬우며, 설계 방법에 따라 시야의 방향과 각, 거리를 다양하게 선택할 수 있다^[4].

비전 기반 시스템은 위와 같이 다양한 장점을 가진 반면, 조명과 기상 상태에 민감하며 영상 데이터는 연산비용이 많이 드는 단점이 있다^[5]. 비전 센서 자체의 한계 외에도 주행 중 차량 자세 변화에 따른 외형 변화는 차량 검출 및 추적을 어렵게 만드는 요인이다. 따라서 최근의 비전 기반 연구는 다양한 주행환경에 의한 차량의 외형 변화를 온라인으로 학습하여 차량 검출 및 추적의 성능을 강건하게 만드는 방향으로 이루어지고 있다. 이와 관련된 연구에는 분류기 온라인 훈련^[6], 차량의 외형 모델링 및 모델 업데이트^[7], 둘을 모두 사용한 방법^[4] 등이 있다.

Chang 등^[6]은 차량 검출에 AdaBoost 기반 온라인 부스팅(boosting) 알고리즘을 사용하였다. 이 방법은 분류기 오프라인 훈련과 마찬가지로 계산 복잡도가 높다. 또한 하나의 분류기를 훈련시키기 때문에 차량의 수가 많고 종류가 다양한 경우에는 차량 검출 성능이 낮다. Mei 등^[7]은 l_1 최소화(l_1 minimization)를 통한 추적 방법을 제안하고, 이를 차량 추적 및 분류 문제에 적용하였다. 추적차량은 템플릿(target and trivial template)의 희소성 표현(sparse representation)으로 모델링되며, 템플릿 업데이트를 통해 추적차량의 외형 변화를 학습한다. 이 방법은 기존의 다른 추적 방법에 비해 부분적인 가려짐(partial occlusions)에 강건하지만, 연산 복잡도가 상당히 높기 때문에 실시간 시스템에 적용하기에는 제약이 따른다. Caraffi 등^[4]은 TLD (tracking-learning-detection) 추적기와 오프라인 훈련된 차량 검출기를 결합한 차량 검출 및 추적 시스템을 제안하였다. Kalal 등^[8]이 제안한 TLD 추적기는 일시적으로 타겟을 놓친 경우 검출기에 의한 추적기 재초기화 메커니즘을 갖고 있다. 이를 위해 준교사 학습(semi-supervised learning) 방법에 속하는 P-N 학습^[9]을 이용하여 온라인으로 검출기를 훈련시킨다.

TLD 추적기는 기존의 여러 추적기와 비교하였을 때 타겟의 외형 변화가 심한 경우에도 강건한 성능을 나타내는 것을 증명하였다.

본 논문에서는 차량의 외형 변화에 대한 온라인 학습 능력과 실시간 동작을 고려한 차량 검출 및 추적 시스템을 제안한다. 특히 다양한 주행환경에도 차량을 강건하게 추적하기 위해 차량의 외형 모델과 온라인 차량 검출기를 도입한다. 제안하는 시스템은 TLD 알고리즘의 학습 프레임워크 차용 관점에서 Caraffi 등^[4]의 방법과 유사하지만, 차량의 외형 모델 업데이트 방법을 개선하여 온라인 차량 검출기의 검출 성능을 높이고, 처리시간을 단축시킨 점에서 차별화된다.

II. 시스템 개요

제안하는 시스템의 목적은 전방의 차량을 검출하고 검출된 차량을 강건하게 추적하는 것으로서 블록도는 그림 1과 같다. 검출(Detection) 과정은 전방에 새로 나타난 차량을 찾는 역할을 한다. 새로 검출된 차량에 대해 이후 검출 성공여부와 관계없이 추적(Tracking) 과정을 통해 연속 프레임 간 차량의 움직임을 추정한다. 학습 및 재검출(Learning and Re-detection)은 P-N 학습을 이용하여 추적차량을 재검출하기 위한 별도의 차량 검출기를 훈련시키는 과정으로서, 일시적인 추적 실패 시 재검출된 차량을 이용해 추적기를 재초기화하는 역할을 한다. 통합(Integration) 과정에서는 검출, 추적, 학습 및 재검출 과정의 결과를 결합하여 차량의 현재 위치와 크기를 출력한다. 차량 위치 및

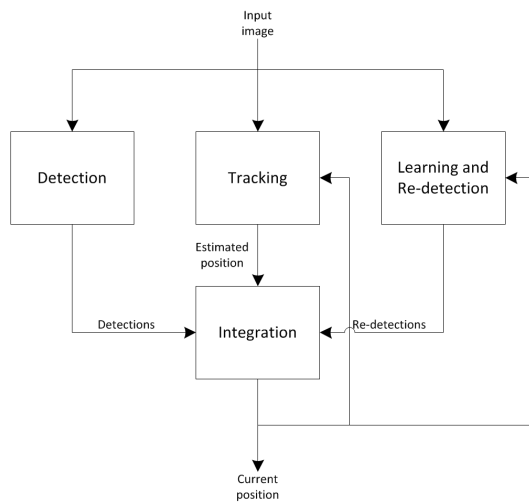


그림 1. 제안하는 시스템의 블록도
Fig. 1. The block diagram of the proposed system.

크기는 다시 다음 프레임에서 해당 차량에 대한 추적, 학습 및 재검출 과정에 사용된다.

제안하는 시스템에서는 차량의 외형 변화를 학습하기 위해 템플릿 기반 차량 모델을 사용한다. 특정 시점에 차량의 외형은 영상 패치로 나타낼 수 있으며, 이러한 패치 여러 장을 모아 차량 모델로 사용할 수 있다. 차량 모델의 크기 및 밝기(intensity)에 대한 불변성을 위해 모든 패치는 크기 15×15 화소, 밝기 평균 0으로 정규화(normalization) 시킨다. 본 논문의 이후 설명에서는 정규화 된 패치를 기호 p 로 나타낸다. 따라서 차량 모델 M 은 현재까지 관측된 차량 및 배경 패치 p 의 집합으로 수식 (1)과 같이 나타낼 수 있다⁸⁾.

$$M = \{p_1^+, p_2^+, \dots, p_m^+, p_1^-, p_2^-, \dots, p_n^-\} \quad (1)$$

이 때 p^+ 와 p^- 는 각각 긍정(차량) 패치와 부정(배경) 패치를 나타낸다. 특히 부정 패치는 차량과 구별이 어려운 배경 클러터(clutter)로 구성된다. 그림 2는 제안하는 시스템 수행 도중 업데이트된 차량 모델 M 의 예를 보여준다. 차량 모델 M 은 학습 및 차량 재검출 과정에서 입력 패치를 차량 또는 배경으로 분류하는데 사용하며, 이에 대해 IV장 및 V장에서 자세히 설명한다.

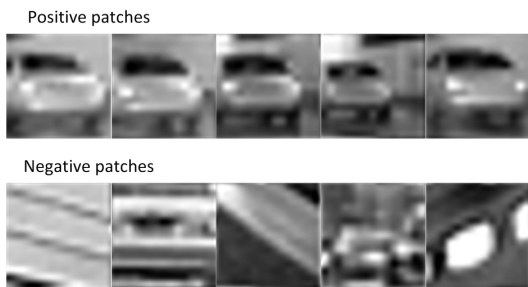


그림 2. 차량 모델의 긍정 패치 및 부정 패치 예
Fig. 2. The examples of positive and negative patches of a vehicle model.

III. 차량 검출 및 추적

3.1 차량 검출

차량 검출은 입력 영상에서 새로 나타난 차량을 찾는 과정이다. 시스템의 실시간 동작을 고려하여 차량 검출에는 Haar-like 특징과 Adaboost 기반 분류기 조합을 사용한다. Haar-like 특징은 차량 후면과 같이 수

평, 수직 및 대칭적인 구조를 검출하는데 적합하며, 적분 영상(integral image)을 이용해 특징을 빠르게 계산할 수 있다. Adaboost 알고리즘으로 학습된 분류기의 캐스케이드 구조(cascade of classifiers)는 검출율이 높고 분류속도가 빠르다는 장점이 있다¹²⁾. 분류기의 오프라인 훈련에는 차량 표본(positive sample) 5,500개와 비차량 표본(negative sample) 3,150개를 사용하였다.

분류기의 일시적인 오 검출(false positive)이 차량으로 검출되는 것을 막기 위해 매 프레임마다 이전 $n-1$ 프레임의 분류결과를 참조한다. 즉 연속된 n 프레임에 걸쳐 일정한 영역에 존재하는 분류결과를 차량으로 검출한다. 인접 프레임 간 분류결과와의 정합 여부는 오버랩(overlap)을 기준으로 판단한다. 두 경계상자(bounding box) bb_1, bb_2 의 오버랩은 수식 (2)와 같이 계산된다.

$$O(bb_1, bb_2) = \frac{|bb_1 \cap bb_2|}{|bb_1 \cup bb_2|} \quad (2)$$

이 때 bb_1, bb_2 에 대해 $|bb_1 \cap bb_2|$ 는 교집합의 면적, $|bb_1 \cup bb_2|$ 는 합집합의 면적을 나타내며 O 의 값은 0에서 1사이이다. 본 논문의 실험에서는 인접 프레임 간 오버랩이 0.6이상(두 경계상자 크기가 같을 때 75%이상 겹침)이고 세 프레임 연속($n=3$)으로 나타나는 경계상자를 차량으로 판단한다. 위 과정을 통해 검출된 차량은 결과 통합 과정에서 이미 추적중인 차량인지 여부를 검증한다.

3.2 차량 추적

차량 추적은 새로 검출된 차량에 대해 이후 연속 프레임 간 움직임을 추정하는 과정이다. 여러 대의 차량을 빠르게 추적하기 위해 특징점 기반 추적 방법을 사용한다. KLT 특징 추적기¹⁰⁾는 소수의 특징점에 대해서만 옵티컬 플로우(optical flow)를 계산하기 때문에 다른 추적 방법에 비해 빠르다. 또한 피라미드 영상(pyramidal image)을 이용하여 특징점의 큰 움직임도 추정할 수 있다.

먼저 차량 경계상자 내에서 Shi 등의 방법¹¹⁾으로 특징점을 추출한다. 이때 경계상자에 포함된 배경의 영향을 최소화하기 위해 가장자리의 특징점은 제외한다. 이후 Lucas-Kanade 방법으로 특징점의 변위를 추정한다. 다음 원래 특징점과 추정된 특징점 주변 10×10 크기 패치 밝기 차(difference)를 비교하여 오

차가 큰 특징점은 제외한다. 남은 특징점 변위의 중앙값(median)을 방향별로 계산하여 차량의 전체 움직임으로 추정한다. 추적기가 차량 외형에 적응적으로 동작하도록 특징점은 매 프레임마다 새로 추출한다.

Lucas-Kanade 방법은 기본적으로 밝기 항상성(brightness constancy)을 가정하므로, 터널 진입 시와 같은 급격한 조명 변화에서는 대부분의 특징점 추정에 실패하게 된다. 제안하는 시스템에서는 추가적으로 인접 프레임 간 차량의 움직임이 크지 않다고 가정하여, 특징점 변위로부터 추정된 차량 움직임의 크기가 θ_m 이상인 경우 추적 실패로 판단한다. 이 때 θ_m 는 10~20 화소이며 프레임율에 따라 실험적으로 정한다. 추적 실패의 경우 추적기는 차량의 위치와 크기를 나타내는 경계상자를 출력하지 않는다.

IV. 학습 및 차량 재검출

조명 변화나 차량 간 가림 발생 시, 특징점 기반 추적 방법은 대부분의 경우 차량 움직임 추정에 실패하게 된다. 따라서 학습 및 차량 재검출 과정에서는 추적중인 차량에 대해 온라인 학습을 이용하여 별도의 차량 검출기를 훈련시키고, 매 프레임 검출기를 수행한다.

온라인 차량 검출기의 목적은 차량 모델 M 을 이용하여 입력 패치를 차량 또는 배경으로 분류하는 것이다. 이를 위해 그림 3과 같이 random ferns와 nearest neighbor 분류기의 캐스케이드 구조^[8]를 사용하며, 슬라이딩 윈도우(sliding window) 접근법에 따라 입력 패치를 분류한다. 분산 필터는 분산이 낮은 패치(차량 초기 패치 분산의 50%이하)를 탈락시켜 도로면과 같은 배경의 분류시간을 최소화하는 전처리 역할을 한다.

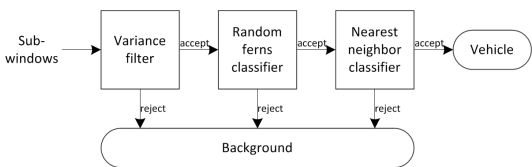


그림 3. 온라인 차량 검출기의 블록도
Fig. 3. The block diagram of the online vehicle detector.

4.1 P-N 학습

P-N 학습^[9]은 레이블 된 데이터(labeled data)와 레이블 되지 않은 데이터(unlabeled data)를 모두 사용하

여 이진 분류기(binary classifier)를 훈련시키는 방법이다. 이때 레이블 되지 않은 데이터는 구조화(structured)되어 있다고 가정하여, 구조적 제약을 학습에 이용한다. 그림 4와 같이 P-제약은 분류된 부정예제 중 제약에 따라 긍정예제(false negatives)를, N-제약은 긍정예제 중 제약에 따라 부정예제(false positives)를 찾는다. 이와 같이 두 제약은 독립적으로 분류기의 오차를 추정하고, 잘못 분류된 예제를 사용하여 분류기를 재훈련시킨다. 즉, P-N 학습은 레이블 되지 않은 예제에 대해 분류기와 구조적 제약에 따른 결과 레이블이 다른 예제를 트레이닝 세트에 사용하여 분류기의 성능을 부트스트래핑(bootstrapping)하는 방법이다.

주행영상의 경우 패치들은 하나의 영상 내에서 공간적으로 관련되어 있고, 인접 프레임에서 시간적으로 관련되므로 시공간(spatio-temporal) 제약을 구조적 제약으로 사용한다. 추적중인 차량의 일련의 위치를 궤적(trajectory)이라 하면 추적차량 패치의 신뢰도가 높은 경우를 유효한 궤적으로 판단할 수 있다(신뢰도의 계산은 결과 통합에서 설명한다). P-제약은 유효한 궤적과 가까운 패치를 긍정 레이블로, N-제약은 유효한 궤적의 주변 패치를 부정 레이블로 제한하게 된다.

제안하는 시스템에서는 매 프레임마다 추적기의 궤적과 오버랩이 높은 패치를 긍정예제로, 궤적과 오버랩이 낮은 패치를 부정예제로 사용한다. 추적차량 패치의 신뢰도를 계산하여 궤적이 유효할 경우에만 위의 예제를 이용하여 온라인 검출기를 훈련시킨다. 추적 궤적이 유효하지 않고 온라인 검출기의 결과 신뢰도가 높은 경우, 이 결과를 이용하여 추적기를 재초기화시킨다.

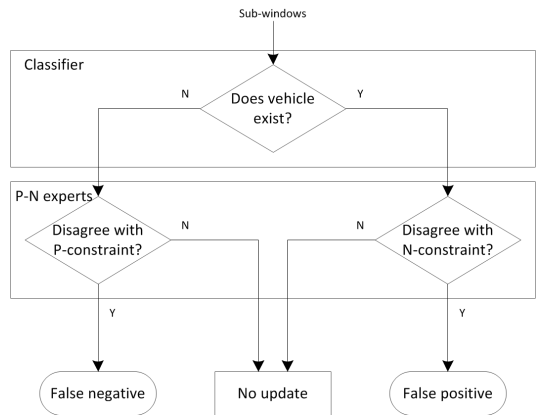


그림 4. P-N 학습 과정의 흐름도
Fig. 4. The flowchart of the P-N learning.

4.2 Random ferns classifier

분산 필터링을 통해 분류할 패치의 수를 크게 줄일 수 있으나, 차량 모델 M 과 템플릿 매칭을 통해 분류하기에는 여전히 많은 수의 패치가 남게 된다. 따라서 특징점 매칭으로 패치를 빠르게 분류할 수 있는 random ferns (RF)를 사용한다. RF는 특징점 매칭 과정의 속도 및 성능이 뛰어나며, 점진적인 훈련이 가능하므로 온라인 분류기에 적합한 특성을 가지고 있다.

RF의 특징은 무작위로 추출된 화소 쌍의 밝기 비교^[13]를 사용한다. 패치의 임의의 두 화소 밝기에 따른 이진 특징(binary feature) f_i 의 값은 수식 (3)과 같다.

$$f_i = \begin{cases} 1 & \text{if } I(l_{i,1}) < I(l_{i,2}) \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

이 때 I 는 영상 패치, $l_{i,1}$ 와 $l_{i,2}$ 는 f_i 의 두 화소 위치를 나타낸다. RF는 특징점 매칭의 계산 복잡도를 줄이기 위해 Semi-Naive Bayesian 접근법^[12]을 따른다. 즉 영상 패치에서 추출된 특징을 다수의 그룹(fern)으로 나누고 서로 다른 그룹 간에 독립성을 가정한다. 하나의 fern에 속하는 이진 특징의 수를 S 개라 할 때, 리프 노드(leaf node)의 수는 2^S 개가 된다. 각 리프 노드에는 분류기 훈련 중 이 리프 노드에 도달한 긍정 및 부정 패치의 수가 저장된다.

그림 5는 RF 분류기를 이용한 영상 패치 분류를 보여준다. 각 열은 하나의 fern을, 한 쌍의 흑백점은 각각 하나의 이진 특징을 나타낸다. 각각의 fern i 는 입력 패치에 대해 이진특징을 계산하여 fern i 의 리프 노드 중 하나를 가리키는 특징벡터 x_i 를 출력한다. 이 리프 노드에 도달한 입력 패치가 긍정 레이블일 사후 확률(posterior probability) $P(y=1|x_i)$ 은 수식 (4)와 같다.

$$P(y=1|x_i) = \frac{n^+}{n^+ + n^-} \quad (4)$$

이 때 n^+ 와 n^- 는 각각 이 리프 노드에 저장된 긍정 및 부정 패치의 수이다. 모든 ferns의 사후 확률 평균이 0.5 이상이면 입력 패치는 긍정 레이블로 분류된다.

차량 검출 과정에서 새로 발견된 차량마다 새 RF를 생성한다. 제안하는 시스템에서는 성능과 속도 간 트레이드오프로 ferns의 수 10개, fern의 이진 특징 수 10개를 사용한다. Ferns의 사후 확률은 모두 0으로 초

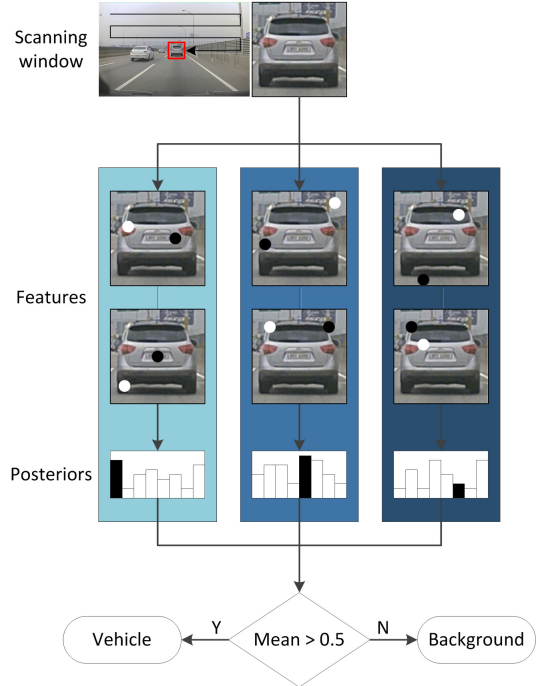


그림 5. Random ferns classifier를 이용한 영상 패치 분류
Fig. 5. Image patch classification using random ferns classifier.

기화하고, 새로 발견된 차량의 패치와 이 패치의 아핀 변환(affine deformations)을 사용하여 RF를 초기 훈련 시킨다. 이후 프레임부터는 P-N 학습을 이용하여 각 리프 노드의 긍정 및 부정 패치의 수를 업데이트한다.

4.3 Nearest neighbor classifier

RF 분류기를 통해 긍정 레이블로 분류되는 패치는 보통 수십 개 미만이다. Nearest neighbor (NN) 분류기는 차량 모델 M 을 이용하여 최근접 이웃 법칙에 따라 입력 패치를 차량 또는 배경으로 분류한다. 템플릿 매칭을 위해 모든 입력 패치는 p 로 정규화 한다. 임의의 두 패치를 각각 p_i, p_j 라 할 때, 패치 간의 거리 d 는 수식 (5)와 같다.

$$d(p_i, p_j) = 1 - 0.5(NCC(p_i, p_j) + 1) \quad (5)$$

이 때 NCC는 normalized correlation coefficient를 나타낸다. 따라서 d 는 0에서 1 사이의 값이며, 값이 작을수록 패치 간의 거리가 가까운 것을 의미한다. 입력 패치 p 에 대해 긍정 및 부정 최근접 이웃 거리는 각각 수식 (6), (7)과 같다.

$$d^+(p, M) = \min_{p_i^+ \in M} d(p, p_i^+) \quad (6)$$

$$d^-(p, M) = \min_{p_i^- \in M} d(p, p_i^-) \quad (7)$$

d^+ 와 d^- 의 값은 각각 긍정 및 부정 최근접 이웃에 의해 결정된다. 수식 (6)과 (7)을 이용해 p 가 차량 모델 M 의 긍정 또는 부정 패치 중 상대적으로 어느 쪽에 가까운지 나타내는 상대 거리(relative distance) d^r 를 수식 (8)과 같다.

$$d^r(p, M) = \frac{d^-}{d^- + d^+} \quad (8)$$

d^r 의 값은 0에서 1 사이이며, 클수록 임의의 패치가 차량 모델 M 의 긍정 패치에 가까운 것을 의미한다. NN 분류기는 $d^r(p, M) > \theta_{NN}$ 이면 p 를 차량으로 분류하며, 이 때 θ_{NN} 의 값은 0.6~0.7로 정한다.

4.3.1 기존의 모델 업데이트 방법

차량 모델 M 의 긍정 템플릿은 차량의 초기 패치와 이 패치의 이완 변환으로, 부정 템플릿은 초기 패치로부터 오버랩이 낮은 주변 패치로 초기화한다. 이후 업데이트는 RF 분류기와 같이 P-N 학습 과정을 따른다.

기존의 모델 업데이트는 이미 추가된 템플릿은 삭제하지 않거나 무작위 삭제하는 방법을 사용한다. 긍정 템플릿이 잘못 추가된 경우(배경 클러스터 등)는 결국 추적차량에 드리프트가 발생하게 된다. 단일 타겟을 추적하는 경우 모델의 템플릿의 수는 보통 수백 개 근처에서 수렴하며, 템플릿 크기가 작기 때

문에 메모리 측면에서는 문제가 되지 않는다^[8]. 그러나 제안하는 시스템의 경우 여러 대의 차량을 고려하므로 전체 템플릿의 수는 훨씬 많아진다. 주행환경에 따라 템플릿의 수는 계속 증가할 수 있다. 또한 템플릿의 수가 늘어날수록 NN 분류시간은 증가하게 된다. 또한 잘못 추가된 긍정 템플릿에 의한 드리프트 발생을 막기 위해 사용되지 않는 긍정 템플릿을 대체하는 방법을 제안한다.

4.3.2 제안하는 모델 업데이트 방법

제안하는 방법에서는 긍정 템플릿의 수를 제한하고 템플릿마다 가중치(weight)를 부여한다. 차량 모델 M 초기화 시 긍정 템플릿의 가중치는 모두 동일하게 설

```

1:  $w_i \leftarrow w_i \times \exp(c_i)$ 
2: if ( $d^+(p, M) > \theta$ ) then
3:    $i_0 \leftarrow \operatorname{argmin}_{1 \leq i \leq m} w_i$ 
4:    $p_{i_0}^+ \leftarrow p$ ,
5:    $w_{i_0} \leftarrow \operatorname{median}(\mathbf{w})$ ,
6: endif
7: Normalize  $\mathbf{w}$  such that  $\|\mathbf{w}\| = 1$ .
8: Adjust  $\mathbf{w}$  such that  $\max(\mathbf{w}) = 0.3$ 
    
```

그림 6. 차량 모델의 긍정 템플릿 업데이트 규칙
Fig. 6. Positive template update rule for vehicle model.

정한다. 이후 P-experts에 의해 새로운 템플릿을 추가하는 경우 그림 6의 규칙을 따른다. 이 때 p 는 현재 프레임에서 차량 추적 결과, \mathbf{w} 는 긍정 템플릿의 현재 가중치 벡터를 나타낸다.

먼저 p 와 $p_{i_0}^+$ 의 NCC 계수에 따라 긍정 템플릿의 가중치 w_i 를 각각 업데이트 시킨다(1). 이 과정에서 현재 추적차량과 유사한 긍정 템플릿은 가중치가 커지고, 유사하지 않은 것은 가중치가 낮아진다. 다음 추적차량과 유사한 $p_{i_0}^+$ 가 없는 경우에만 p 를 긍정 템플릿에 추가한다(2). 이 경우 가중치가 가장 낮은 기존 템플릿 $p_{i_0}^+$ 가 대체된다(3, 4). 이 때 θ 의 값은 0.3 정도이며 실험적으로 정한다. 새로 추가된 긍정 템플릿의 가중치는 전체 가중치의 중앙값으로 초기화하여 다시 일찍 대체되는 것을 방지한다(5). 위 과정이 끝나면 가중치 벡터의 norm이 1이 되도록 \mathbf{w} 를 정규화한다(7). 추가적으로 연속된 프레임에서 특정 템플릿의 가중치가 과도하게 커지는 것을 막기 위해 가중치의 최대값을 제한한다. 따라서 특정 템플릿의 가중치가 0.3을 초과하면, 이를 0.3으로 고정시키고 $\|\mathbf{w}\| = 1$ 을 만족하도록 나머지 템플릿의 가중치를 증가시킨다(8). 차량 모델에서 차량에 비해 배경의 변화는 훨씬 빠르므로 부정 템플릿 업데이트는 기존 템플릿 중 하나를 무작위로 대체하는 방법을 사용한다.

V. 결과 통합

결과 통합은 모듈별 결과를 통합하여 차량의 현재 위치와 크기를 출력하는 과정이다. TLD에서는 추적 차량 패치와 첫 프레임의 차량패치의 NCC를 신뢰도로 사용하여 유효한 궤적 여부를 판단한다. 또한 추적기와 온라인 검출기의 신뢰도 비교에는 보존 유사성 S^c 를 이용한다^[8]. 이는 긍정 최근접 이웃 거리를

계산하는데 차량 모델 M 에 추가된 시간 순으로 50% 이내인 긍정 템플릿만 사용하는 것이다. 즉, TLD는 첫 프레임에서 수동으로 지정해준 경계상자가 원래의 타겟 외형을 잘 표현한다는 것을 전제한다.

제안하는 시스템에서는 차량 검출 과정을 통해 자동으로 새로운 차량이 추가되므로 위의 전제가 항상 적용되지 않는다. 따라서 신뢰도 계산에는 모두 수식 (8)의 상대 거리 d^r 을 이용한다. 추적기와 온라인 차량 검출기 모두 결과 없는 경우는 전방에 차량이 보이지 않는 상태로 판단한다.

특정 프레임에서 차량 검출 과정을 통해 발견된 차량이 이미 추적중인 차량인지 판단하기 위해 수식 (2)의 오버랩과 추가적으로 중심 유클리디언 거리 (Euclidean distance)를 사용한다. 이 때 두 경계상자의 오버랩이 0.6이상이고, 중심 거리가 10화소 이내인 경우를 정합기준으로 한다. 추적중인 차량 중에서 차량 검출을 통해 발견된 차량과 정합되는 것이 없으면 새로운 차량으로 판단한다. 따라서 추적기와 온라인 차량 검출기를 초기화하고 이후 프레임부터 추적과 재검출을 수행한다.

매 프레임마다 추적중인 차량의 경계상자 간 오버랩을 계산하여 차량 간 가림 발생을 판단한다. 특정 차량에 50% 이상 오버랩이 발생한 경우 차량 추적 및 학습을 중지하고 온라인 차량 검출기로 재검출만을 수행한다. 일시적인 가림의 경우 이러한 방법으로 차량 모델이 배경 클러스터를 함께 학습하는 현상을 막는다. 장기간 가림의 경우는 온라인 차량 검출기의 검출 실패횟수에 따라 일정횟수를 초과하면 추적중인 차량에서 삭제하는 방법을 사용한다.

추가적으로 연속적인 프레임에서 차량의 위치와 크기를 예측하기 위해 칼만 필터를 이용한다. 예측된 차량의 위치와 크기는 온라인 차량 검출기의 관심영역으로 사용되어 검출 수행시간을 단축시키는 역할을 한다.

VI. 실험

본 장에서는 제안하는 차량 모델 업데이트 방법이 차량 검출 및 추적 성능에 미치는 영향을 비교하고 그 결과를 분석한다. 이를 위해 다양한 도로환경에서 자체적으로 획득한 영상 및 일반 공개된 데이터셋을 사용하였다. 각 절의 실험마다 정량 및 정성적인 평가를 제시한다. 실험은 Intel Core i7 950 3.06GHz CPU의 PC에서 수행하였다.

6.1 HD 영상 데이터셋

HD 영상 데이터셋은 15fps(frame per second)로 취득한 1280×800 크기의 영상으로 구성된다. 이 데이터셋에서는 온라인 차량 검출기의 도입이 차량 검출 및 추적 성능에 미치는 영향을 분석한다. 정량적인 성능 평가를 위해 프레임 당 차량 수 NV/Frame (number of vehicles per frame), 전체 차량 중 검출한 차량의 비율 TPR(true positive rate), 검출된 차량에서 오 검출의 비율 FDR(false detection rate), 프레임 당 오 검출의 비율 FP/Frame(false positive per frame)을 사용한다.

표 1은 실험에 사용된 HD 영상 데이터셋의 주행 환경에 대한 설명이다. 표 2는 표 1의 데이터셋에 대한 제안하는 시스템의 성능평가 결과이다. (a-c)는 온라인 차량 검출기를 사용하지 않는 경우, (d-f)는 사용하는 경우이다.

맑은 날씨의 경우 날씨가 조명에 따른 차량 외형 변화가 거의 발생하지 않는다. 따라서 그림 7 (a), (d)와 같이 온라인 차량 검출기의 사용 여부와 관계없이 높은 TPR과 낮은 FDR을 나타낸다. 비오는 날씨의 경우 물안개에 의해 차량의 외형이 흐려지는 현상 (blurring)이 자주 발생한다. 따라서 그림 7 (b)와 같이 단순히 차량 검출과 추적 과정만을 수행하는 경우에는 TPR이 낮다. 그림 7 (e)는 차량 모델을 이용해 외형이 흐려졌음에도 검출된 차량을 확인할 수 있다. 터널 진입 시에는 조명 변화에 의해 차량의 외형이 달라진다. 특히 터널의 조명이 차량에 반사되면서 차량의 외형 변화로 작용하며, 수 프레임 동안 차량 추적이

표 1. HD 영상 데이터셋에 대한 설명
Table 1. A description of the HD image dataset.

	# of frames	NV/Frame	Place	Weather
1	900	3.4	Highway	Sunny
2	500	2.9	Urban road	Rainy
3	900	4.1	Urban road	Tunnel

표 2. HD 영상 데이터셋에서 제안하는 시스템의 온라인 차량 검출기 사용여부에 따른 성능
Table 2. Performance of the proposed system with/without online vehicle detector under HD image dataset.

		TPR(%)	FDR(%)	FP/Frame
1	(a)	96.0	0.4	0.013
	(d)	98.1	0.3	0.010
2	(b)	78.8	1.5	0.034
	(e)	90.5	1.2	0.031
3	(c)	86.7	1.7	0.060
	(f)	92.1	0.4	0.015

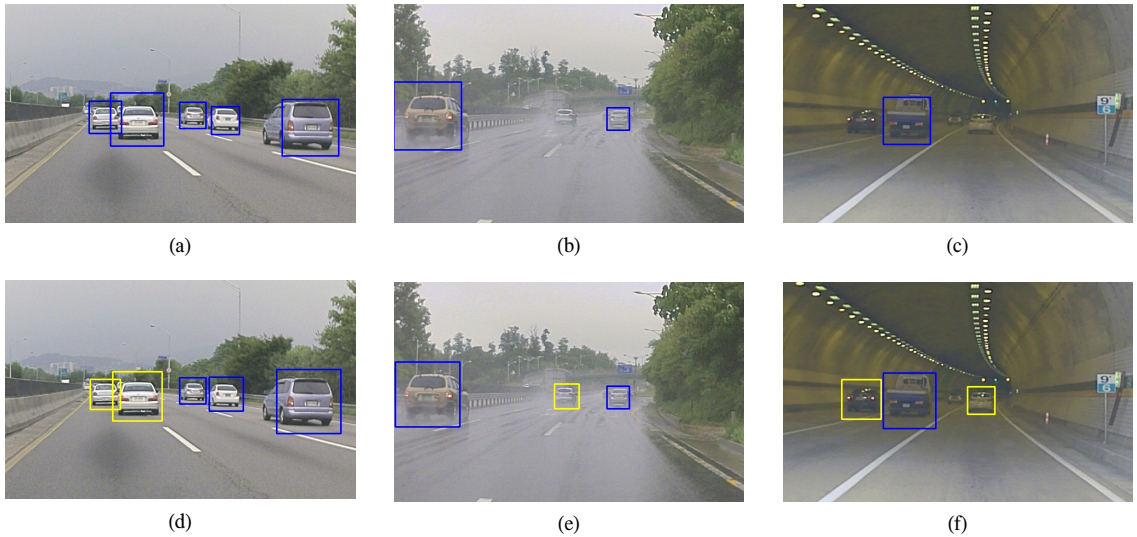


그림 7. 다양한 주행환경에서 온라인 차량 검출기 사용 여부에 따른 결과 비교: (a-c) 온라인 차량 검출기 사용, 노란색 경계상자는 occlusion, 흐려짐, 조명 변화에 의해 최종 결과의 신뢰도가 낮은 것을 의미하며, 이 경우에는 차량의 외형을 학습하지 않는다.

Fig. 7. Comparison of the results of the proposed system on HD image datasets: (a-c) without online detector, (d-f) with online detector. The yellow bounding boxes mean the results with low confidence due to occlusion, blurring and illumination change. In this case, the online detector of those vehicles is not trained.

실패한다. 그림 7 (f)는 추적기가 수 프레임 실패한 후에도 온라인 차량 검출기가 두 대의 차량을 다시 검출하여 추적기를 재초기화한 것을 나타내며, (c)는 차량 검출 과정에 의해 다시 검출될 때까지 추적에 실패한다. 제안하는 시스템은 프레임 당 3~4대의 차량이 나타나는 HD 영상 데이터셋에서 20~25 fps의 처리 속도로 동작하였다.

추가적으로 제안하는 차량 모델 업데이트 방법이 차량 추적 성능에 미치는 영향을 분석하기 위해 TLD 추적기와 성능을 비교한다. TLD 추적기의 결과는 저자가 공개한 MATLAB 코드^[8]를 이용하였다. 각각 500 프레임으로 구성된 우천, 터널 통과 HD 영상 데이터셋에서 결과를 비교하였다. 우천의 경우 전방차량에 비에 의한 흐려짐과 와이퍼에 의한 가려짐이 빈번히 발생하여 차량 추적과 재검출에 제약으로 작용한다. 특히 그림 8 (a), (c)에서 전방차량이 우회전하면서 자세 변화가 일어날 때 TLD 추적기는 보존 유사성에 의해 외형 변화를 따라가지 못하고 추적 및 재검출에 실패한다. 제안하는 시스템은 첫 프레임에서 차량의 외형과 관계없이 차량 모델을 업데이트하므로, 차량의 우회전시에도 재검출에 성공하여 프레임이 끝날 때까지 강건하게 추적한다. 터널 통과와 조명 변화와 함께 차량의 움직임이 빠른 특징이 있다. 그림 8 (b), (d)는 두 시스템의 최종출력에서 배경이 50%

이상을 차지하기 시작하는 프레임을 보여준다. 두 시스템 모두 차량 모델의 긍정 템플릿이 배경을 포함하면서 최종적으로는 추적과 재검출 모두 실패하게 된다. 이 과정에서 TLD 추적기의 차량 모델 업데이트는 단순히 긍정 템플릿을 추가하는 것이기 때문에 배경으로 인한 잘못된 패치의 수가 급격히 늘어나 최종출력의 드리프트가 더 빨리 발생한 것을 그림 8 (b), (d)에서 확인할 수 있다.

6.2 TME motorway 데이터셋

TME motorway 데이터셋은 Daylight와 Sunset 데이터셋으로 구성된다^[4]. 본 실험에서는 Sunset 데이터셋(10,894 프레임)를 이용하여 제안하는 시스템과 Caraffi 등^[4]의 시스템의 성능을 비교하였다. 해당 데이터셋에서 제공하는 정량적 지표는 다음과 같다.

$$recallrate : \frac{TP}{TP + FN} \quad (9)$$

$$precision : \frac{TP}{TP + FP} \quad (10)$$

이 때 TP, FN, FP는 각각 정 검출(true positives), 미 검출(false negatives), 오 검출(false positives)의 수를 나타낸다. TME motorway 데이터

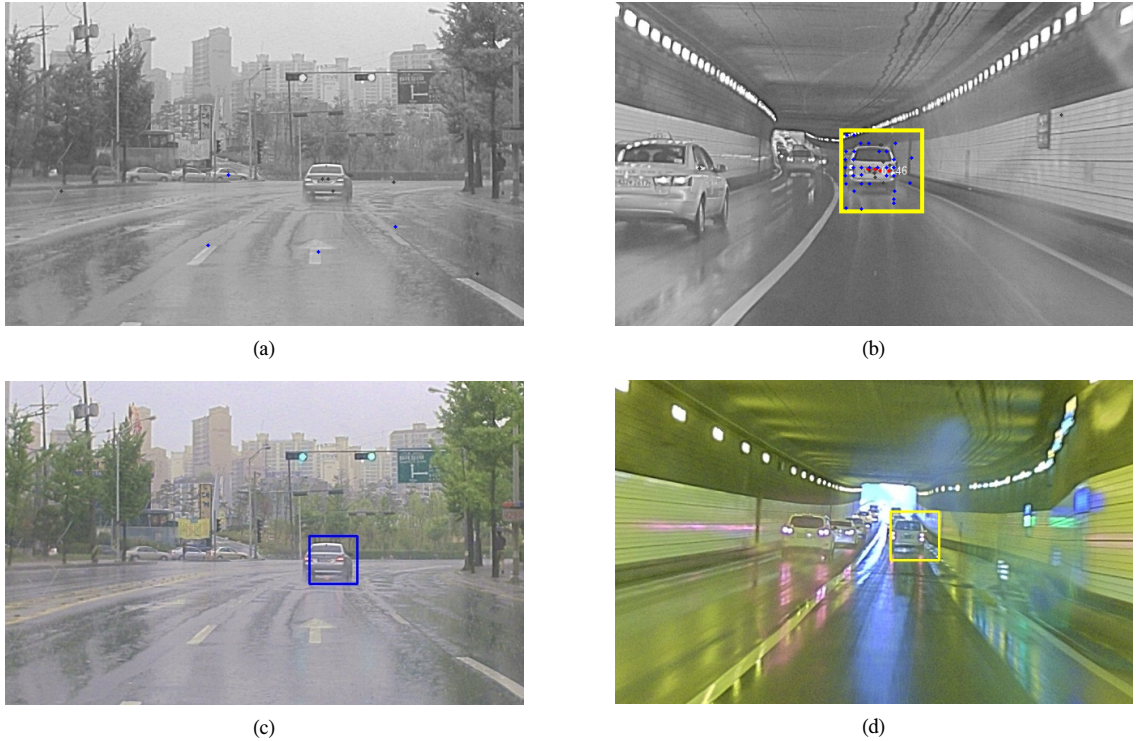


그림 8. HD 영상 데이터셋에서 제안하는 시스템과 TLD 추적기의 성능 비교: (a-b) TLD 추적기의 결과 (c-d) 제안하는 시스템의 결과. (b), (d)에서 노란색 경계상자는 차량의 빠른 움직임과 조명 변화에 의해 최종 결과의 신뢰도가 낮은 것을 의미하며, 이 경우에는 차량의 외형을 학습하지 않는다.

Fig. 8. Comparison of the results of the proposed system on HD image datasets: (a-b) results of the TLD tracker, (c-d) result of the proposed system. The yellow bounding box in (b), (d) means the results with low confidence due to rapid motion of vehicles and illumination change. In this case, the online detector of those vehicles is not trained.

세트는 레이저 스캐너로 측정된 실제 차량 기준 (ground truth)과 이를 바탕으로 임의의 시스템의 출력의 성능을 평가할 수 있는 툴(evaluator)을 제공한다.

표 3은 위의 툴을 이용하여 Caraffi 등^[4]의 시스템과 제안하는 시스템의 성능을 비교한 결과이다. 제안하는 시스템의 재현율(recall rate)은 승용차 70.3%, 트럭 58.4%로 Caraffi 등^[4]의 시스템에 비해 각각 7%, 9% 정도 낮다. 재현율이 낮은 이유로 추적중인 차량의 추적 실패에 따른 미 검출보다는, 오프라인 차량 검출기가 검출하지 못하는 차량에 의한 미 검출 비중

이 컸다. 하지만 일단 검출된 차량에 대해 트럭의 정확도(precision)는 제안하는 시스템과 Caraffi 등^[4]의 시스템이 각각 85.8%, 86.5%로 거의 비슷한 성능을 보였다. 승용차의 정확도는 Caraffi 등^[4]의 시스템이 95.3%, 제안하는 시스템이 97.1%였다. 특히 재현율의 미 검출과 마찬가지로 대부분의 오 검출이 오프라인 차량 검출기에 기인한 것에서, 제안하는 시스템의 차량 추적 성능이 더 강건한 것을 확인할 수 있다. 제안하는 시스템의 재현율 성능을 향상시키기 위해서는 오프라인 차량 검출기의 추가적인 훈련을 통해 검출율을 높일 필요가 있다. 특히 트럭의 경우 승용차와는 외형이 다소 다르기 때문에, 별도의 검출기를 훈련시켜 사용하는 것이 검출 성능 면에서 효과적일 것으로 판단된다.

표 3. Caraffi 등^[4]의 시스템과 제안하는 시스템의 성능 비교
Table 3. Performance comparison between Caraffi et al. system^[4] and the proposed system.

Types	The system of Caraffi et al. ^[4]		Proposed system	
	Car	Truck	Car	Truck
Precision	0.953	0.865	0.971	0.858
Recall	0.773	0.679	0.703	0.584

VII. 결 론

본 논문에서는 차량의 외형 변화를 온라인으로 학습하는 차량 검출 및 추적 시스템을 제안하였다. 시스

템에 도입된 온라인 차량 검출기는 추적기와 동시에 수행되면서, 차량 외형 모델을 이용하여 추적중인 차량을 검출한다. 특히 차량 외형 모델의 업데이트 방법을 개선하여 온라인 차량 검출기의 검출 성능을 높이고 처리시간을 단축시켰다. 실험을 통해 제안하는 시스템이 맑은 날씨는 물론 우천, 터널 등의 악조건에서도 강건하게 차량을 검출 및 추적하는 것을 입증하였다. 기존의 방법과 승용차와 트럭 차종을 분리하여 성능을 비교하였을 때에는 특히 승용차 차종에 대한 차량 추적 성능이 개선된 것을 확인하였다.

제안하는 시스템은 별도의 모듈 간 스케줄링 없이 실시간 동작이 가능하다. 또한 단일 카메라를 사용하므로 시스템 비용이 저렴하다. 따라서 차량 검출 및 추적 성능을 향상시키기 위해 여러 대의 시스템으로 확장하거나 다른 종류의 센서와 융합하는데 용이할 것으로 기대한다.

References

- [1] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 5, pp. 694-711, May 2006.
- [2] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. CVPR Conf.*, vol. I, pp. 511-518, 2001.
- [3] J. R. Kim, S. J. Yu, K. A. Toh, D. H. Kim and S. Y. Lee, "Fast on-road vehicle detection using reduced multivariate polynomial classifier," *J. KICS*, vol. 37, no. 8, pp. 639-647, Aug. 2012.
- [4] C. Caraffi, T. Vojir, J. Trefny, J. Sochman and J. Matas, "A system for real-time detection and tracking of vehicles from a single car-mounted camera," in *Proc. ITSC Conf.*, pp. 975-982, Sept. 2012.
- [5] S. Sivaraman and M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Trans. Intelligent Transportation Systems*, pp. 1773-1795, Dec. 2013.
- [6] W. Chang and C. Cho, "Online boosting for vehicle detection," *IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 40, no. 3, pp. 892-902, Jun. 2010.
- [7] X. Mei and H. Ling, "Robust visual tracking and vehicle classification via sparse representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 11, pp. 2259-2272, Nov. 2011.
- [8] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-Learning-Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, Jul. 2012.
- [9] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints," in *Proc. CVPR Conf.*, pp. 49-56, Jun. 2010.
- [10] J. Y. Bouguet, "Pyramidal implementation of the Lucas Kanade feature tracker description of the algorithm," *Technical Report, Intel Microprocessor Research Labs*, 1999.
- [11] J. Shi and C. Tomasi, "Good features to track," in *Proc. CVPR Conf.*, pp. 593-600, Jun. 1994.
- [12] F. Zheng and G. Webb, "A comparative study of semi-naive bayes methods in classification learning," in *Proc. of the Fourth Australasian Data Mining Conference (AusDM05)*, pp. 141-156, Sydney, Dec. 2005.
- [13] M. Ozuysal, P. Fua, and V. Lepetit, "Fast keypoint recognition in ten lines of code," in *Proc. CVPR Conf.*, pp. 1-8, USA, Jun. 2007.

길 성 호 (Sung-Ho Gil)



2012년 2월 : 서강대학교 전자
공학과 졸업
2012년 3월~현재 : 서강대학교
전자공학과 석사과정
<관심분야> 컴퓨터 비전, 패턴
인식

김 경 환 (Gyeong-Hwan Kim)



1984년 2월 : 서강대학교 전자
공학과 졸업
1986년 2월 : 서강대학교 전자
공학과 석사
1996년 2월 : State University
of New York at Buffalo 전기
및 컴퓨터 공학과 박사
1997년 3월~현재 : 서강대학교 전자공학과 교수
<관심분야> 영상신호해석, 패턴인식, 임베디드 시스템
디자인