

# Pedestrian Traffic Counting Using HoG Feature-Based Person Detection and Multi-Level Match Tracking

Sung-Wook Kang<sup>†</sup> · Jin-dong Jung<sup>\*\*</sup> · Hong-il Seo<sup>\*\*\*</sup> · Hae-Yeoun Lee<sup>\*\*\*\*</sup>

## ABSTRACT

Market analysis for a business plan is required for the success in the modern world. Most important part in this analysis is pedestrian traffic counting. A traditional way for this is counting it in person. However, it causes high labor costs and mistakes. This paper proposes an automatic algorithm to measure the pedestrian traffic count using images with webcam. The proposed algorithm is composed of two parts: pedestrian area detection and movement tracking. In pedestrian area detection, moving blobs are extracted and pedestrian areas are detected using HoG features and Adaboost algorithm. In movement tracking, multi-level matching and false positive removal are applied to track pedestrian areas and count the pedestrian traffic. Multi-level matching is composed of 3 steps: (1) the similarity calculation between HoG area, (2) the similarity calculation of the estimated position with Kalman filtering, and (3) the similarity calculation of moving blobs in the pedestrian area detection. False positive removal is to remove invalid pedestrian area. To analyze the performance of the proposed algorithm, a comparison is performed with the previous human area detection and tracking algorithm. The proposed algorithm achieves 83.6% accuracy in the pedestrian traffic counting, which is better than the previous algorithm over 11%.

**Keywords :** Traffic Counting, Pedestrian Detection and Tracking, Kalman Filter, HoG Feature, Multi-Level Matching

## HoG 특징 기반 사람 탐지와 멀티레벨 매칭 추적을 이용한 보행자 통행량 측정 알고리즘

강성욱<sup>†</sup> · 정진동<sup>\*\*</sup> · 서홍일<sup>\*\*\*</sup> · 이해연<sup>\*\*\*\*</sup>

## 요 약

현대 사회에서 상업적 성공을 위해서는 상권 분석이 필요하며, 상권 분석의 요소 중에서 핵심적인 부분은 통행량이다. 통행량을 측정하기 위해서 사람이 직접 측정하는 방법이 많이 사용되고 있으나 높은 인건비와 측정 실수를 유발할 가능성이 높다. 본 논문에서는 웹캠을 통해 촬영한 이미지를 이용하여 보행자의 통행량을 측정할 수 있는 알고리즘을 제안한다. 제안하는 알고리즘은 사람 영역 탐지와 움직임 추적으로 구성되어 있다. 사람 영역 탐지에서는 움직임 영역을 추출하고 HoG 특징과 Adaboost 분류기를 이용하여 사람 영역을 탐지한다. 움직임 추적에는 멀티 레벨 매칭과 거짓 양성 제거를 이용하여 추적 및 통행량을 측정한다. 멀티 레벨 매칭은 HoG 영역에 대해 유사도 계수를 구하여 판별하는 과정, 칼만 필터를 이용하여 추정한 위치의 이미지 유사도를 계산 과정, 사람 영역 탐지에서 추출한 움직임 영역을 이용해 유사도를 계산하는 3 단계 과정으로 구성되어 있다. 거짓 양성 제거는 사람 영역 탐지에서 잘못된 탐지 영역을 제거한다. 제안한 알고리즘의 성능을 분석하기 위하여 기존의 사람 영역 탐지 및 추적하는 방법과 비교 실험을 수행하였다. 그 결과 제안하는 방법은 사람 통행량 측정에서 83.6% 정확도를 보였으며, 기존 알고리즘에 비하여 11% 높은 성능을 달성하였다.

**키워드 :** 통행량 측정, 보행자 검출 및 추적, 칼만 필터, HoG 특징, 멀티레벨 매칭

## 1. 서 론

현대 사회에서 상업적으로 성공을 거두기 위해서는 상권

분석이 필요하다. 상권 분석에 있어서 가장 중요한 요소는 통행량이다. 실제로 많은 사람이 지나다니는 길은 그렇지 않은 길보다 통계적으로 높은 수익을 보여주는 경우가 많다. Fig. 1은 매출액과 유동인구 간의 상관관계를 보여주며, 유동인구가 많은 곳이 매출액 또한 높음을 볼 수 있다.

이와 같은 통행량을 측정하기 위해 많이 사용되는 전통적인 방법은 사람이 수동으로 직접 카운트해서 통행량을 측정한다. 이 방법은 단순하지만 높은 인건비와 측정실수를 유발할 가능성이 있다.

<sup>†</sup> 비 회 원 : 고려대학교 정보보호대학원 석사과정  
<sup>\*\*</sup> 비 회 원 : 웬즈튜디오 연구원  
<sup>\*\*\*</sup> 비 회 원 : TeamSF 연구원  
<sup>\*\*\*\*</sup> 정 회 원 : 금오공과대학교 컴퓨터소프트웨어공학과 교수  
Manuscript Received: January 14, 2016  
First Revision: April 15, 2016  
Accepted: April 18, 2016  
\* Corresponding Author: Hae-Yeoun Lee(haeyeoun.lee@kumoh.ac.kr)

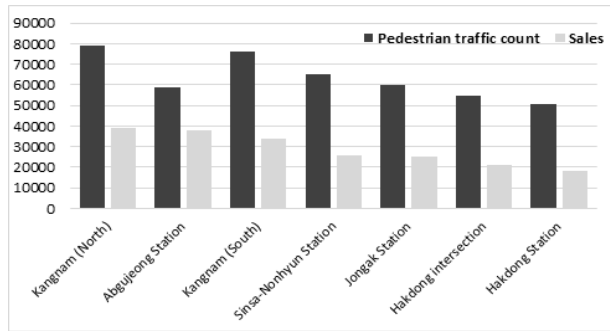


Fig. 1. Relation between pedestrian traffic count and sales in Korea major market places (Maeil business news at Mar. 18, 2012)

본 논문에서는 웹캠을 통해 받은 이미지를 이용하여 보행자의 통행량을 측정할 수 있는 알고리즘을 제안한다. 영상에서 보행자의 통행량을 측정하기 위해서는 보행자의 인식과 추적이 중요하다. 제안하는 알고리즘은 크게 두 부분으로 나눌 수 있다. 사람 영역 탐지와 움직임 추적이다. 사람 영역 탐지에서는 차영상으로 움직임 영역을 먼저 추출하고 그 영역에 대해서 HoG 특징과 Adaboost 분류기를 이용하여 사람인지 아닌지를 판별한다. 움직임 추적에서 멀티 레벨 매칭과 거트 양성 제거를 사용하여 이전 프레임과 다음 프레임의 사람이 동일하지 아닌지를 추적하고 관심 영역(RoI)을 넘어 가면 통행량을 측정한다. 추적을 위한 멀티 레벨 매칭은 총 3단계로 이루어지며, 1단계는 색상과 거리 및 겹침 처리를 통해 최종 유사도 계수를 구하여 판별하고, 2단계는 칼만 필터를 이용하여 추정한 위치에 대해 이미지 유사도를 계산하며, 3단계는 사람 영역 탐지에서 추출한 움직임 영역을 이용해 유사도를 계산한다. 그리고 거트 양성 제거를 통해 사람 영역 탐지에서 잘못된 탐지 영역을 제거하여 정확도를 높였다.

본 논문의 구성은 다음과 같다. 2절에서는 사람 영역 탐지를 위한 기존 연구들에서 사용하는 대표적인 방법에 대해서 소개하고, 3절에서는 제안하는 사람 영역 탐지와 움직임 추적을 통한 통행량 측정 알고리즘을 설명한다. 4절에서는 기존 방법을 이용한 알고리즘과 비교 실험을 통하여 성능을 분석하고, 5절에서는 결론을 짓도록 하겠다.

## 2. 관련 연구

통행량을 측정하기 위해서는 사람 영역 탐지와 움직임 추적이 중요하며, 사람 영역 탐지하기 위해서 널리 사용되는 HoG 특징과 Adaboost 분류기를 간략하게 설명한다. 그 후에 통행량 및 교통량 측정에 대한 기존의 자동화 기술 연구들에 대해서 설명한다.

### 2.1 HoG 특징

Histogram of Oriented Gradients(HoG)는 물체의 경계선

에 있어서 방향 성분의 히스토그램을 사용하는 방법으로 특정 물체를 검출하기 위해서 컴퓨터 비전이나 이미지 처리에서 많이 사용된다. HoG는 Dala et al.에 의해 제안되었다[1, 2]. HoG 특징 추출은 이미지를 일정크기의 작은 구역으로 나누고 그 지역별 영역의 기울기 방향 성분의 발생 빈도를 더하여 히스토그램의 집합으로 나타낸다. HoG는 이미지의 밝기나, 조명변화에 덜 민감한 특성을 갖고 있다. 또한 물체의 경계선 정보를 이용하므로 패턴이 비교적 다른 물체보다 단순한 사람과 같은 물체를 식별하는데 강인한 특징이다.

### 2.2 Adaboost 분류기

Adaboost 분류기는 얼굴인식을 위한 Haar-like 특징과 함께 널리 쓰이는 알고리즘이다[3]. 그러나 분류기의 학습 시간은 상대적으로 많이 걸린다. 하지만 충분한 훈련 데이터와 입력 데이터라면 빠른 인식 성능을 보이며 Support Vector Machine 분류기에 뒤지지 않는 성능을 보여 준다. Adaboost 분류기는 분류를 위한 규칙을 만들고 이전의 규칙을 적용하여 얻은 관측값과 표본 데이터의 분포를 재조정한다. 구별 기능이 약한 분류기들을 성능이 향상된 강한 분류기를 만들기 위해서 사용된다. 검출하기 위한 물체의 양성(positive)와 음성(negative) 영상을 분류하기 위한 약 분류기를 찾아 강 분류기로 만들어 내는 과정이다. 정확히 인식된 영상과 잘못 인식된 영상의 가중치를 조절하여 다음 약한 분류기에 반영한다. 강한 분류기는 약 분류기들의 그룹으로 구성된다. 즉 약 분류기를 묶어 하나의 강 분류기를 생성한다.

### 2.3 통행량 및 교통량 측정 연구

보행자 검출은 컴퓨터 비전 분야에 있어서 주요한 관심 문제 중에 하나로서, Dollar et. al은 최신 연구 동향에 대하여 분석하고, 이 중에 MIT, USC, INRIA, ETH, CalTech 등에서 개발한 대표적인 16개의 훈련에 기반한 보행자 검출 알고리즘에 대한 분석을 수행하였다[4].

Seo and Ji는 다중 프레임에서 HoG 특징을 이용하여 보행자 탐지 속도를 향상한 알고리즘을 제안하였다. 특히, 차영상을 이용한 다중 프레임 기법에서는 보행자 움직임이 없는 경우에 검출이 용이하지 않지만, 배경을 생성하여 활용함으로써 이를 해결하기 위한 방법을 제시하였다[5]. Yang and Jo는 교통 신호 제어를 위하여 HoG 특징을 이용하여 외부 환경에서 조명 변화에 강인한 보행자 검출 알고리즘을 사용하였다. HoG 특징의 강인성을 위한 알고리즘 향상은 없지만 처리 속도를 향상하기 위하여 GPU를 이용한 CUDA 기법을 사용하였다[6].

걸어가는 사람을 검출하기 위하여 Viola et. al은 연속적인 두 프레임에 대하여 움직임이 외형에 대한 정보를 사용하여 AdaBoost 알고리즘을 훈련하고, 이를 이용하여 보행자를 검출하기 위한 알고리즘을 제안하였다[7]. Kong and Hong은 AdaBoost 프레임 워크에서 강 분류기를 통하여 보행자를 검출하기 위한 연구를 수행하였다[8].

Lee et. al은 보행자 검출을 위하여 에지 정보의 기하학적인 특성을 이용하여 보행자 후보 지역을 검출하고, 대칭 히스토그램 방법을 이용하여 보행 후보자의 수를 추정하며, 유전자 알고리즘에 기반한 Support Vector Machine 분류기를 사용하여 보행자를 검출하는 알고리즘을 제안하였다[9].

보행자 검출에 대한 연구는 다양하게 진행 중이며, 본 논문에서는 HoG 특징과 Adaboost 분류기를 이용하고 멀티레벨 매칭을 통하여 보행자 검출 성능을 향상하며, 통행량 분석을 위한 응용에 대하여 제안하였다.

### 3. 제안하는 보행자 추적을 통한 통행량 측정 알고리즘

본 논문에서 제안하는 보행자 추적을 통한 통행량 측정 알고리즘은 사람 영역 탐지와 움직임 추적의 과정으로 구성된다. 제안하는 알고리즘의 대략적인 수행 과정은 Fig. 2에 도식화되어 있다.

웹캠에서 입력되는 영상에 대하여 Tracking ROI는 실제 통행량을 측정하기 위해 사람 영역 탐지와 움직임 추적을 수행하고 있는 관심 영역이며, Candidate ROI는 보행자를 탐지는 하지만 추적은 하지 않는 후보 영역이다. 사람 영역 탐지에서는 웹캠으로부터 이미지를 받아와 움직임 영역을 추출하고, 이에 대하여 슬라이딩 윈도우 탐색과 HoG 특징을 통하여 학습된 Adaboost 분류기를 반복적으로 수행하여 보행자를 탐지한다. 움직임 추적에서는 탐지된 보행자를 Tracking ROI 보행자 리스트와 멀티 레벨 매칭을 시작한다. 매칭이 성공한 경우 각 보행자마다 가지고 있는 위치 정보를 갱신하고 실패한 경우에는 새로운 사람이라고 판단하여

Tracking ROI 보행자 리스트에 추가한다. 마지막으로 통행량 측정은 추적된 위치 정보를 이용하여 Tracking ROI에서 빠져나간 경우 보행자 카운팅을 수행한다.

본 논문에서 제안하는 통행량을 측정하기 위한 알고리즘의 사람 영역 탐지는 3.1절에서 설명하고 3.2절에서는 움직임 추적 및 3.3절에서는 통행량 측정에 대해 설명한다.

#### 3.1 사람 영역 탐지

사람 영역 탐지를 위해서 순서대로 들어오는 두 개의 프레임으로부터 움직임 영역을 먼저 추출한다. 그 영역에 대해 슬라이딩 윈도우 탐색과 HoG 특징을 이용해 학습된 Adaboost 분류기를 통하여 사람인지 아닌지를 탐지한다. 설명의 편의성을 위하여 움직임 영역 추출과 슬라이딩 윈도우 탐색, 보행자 탐지로 나누어 설명한다.

##### 3.1.1 움직임 영역 추출

웹캠으로부터 받아온 n-1번째 프레임과 n번째의 프레임에 대해 전처리 과정으로 회색조 영상 변환 및 히스토그램 평준화를 수행한다. 그 후에 임계치 필터링을 수행하여 이진 영상을 획득한 후에 차영상을 구한다(Fig. 3b 참조). 그 다음에 Fig. 3c와 같이 차영상에 대해 이어져 있는 모든 경계선을 구한 후에 경계선의 중심점을 추출한다. 이를 통하여 일정 임계치 이상 움직인 부분의 중심점 위치를 구할 수 있다. 이 픽셀들을 군집화하여 일정 크기의 영역들로 추출한다. 군집화의 과정은 Fig. 3d와 같이 움직임 중심점들에 대하여 일정 픽셀 거리를 기준으로 인접 행렬을 생성한다. 생성한 인접 행렬에 대하여 깊이 우선 그래프 탐색을 통하여 일정 그룹 단위로 묶어 Fig. 3d와 같이 사각형 형태의 영역을 생성한다.

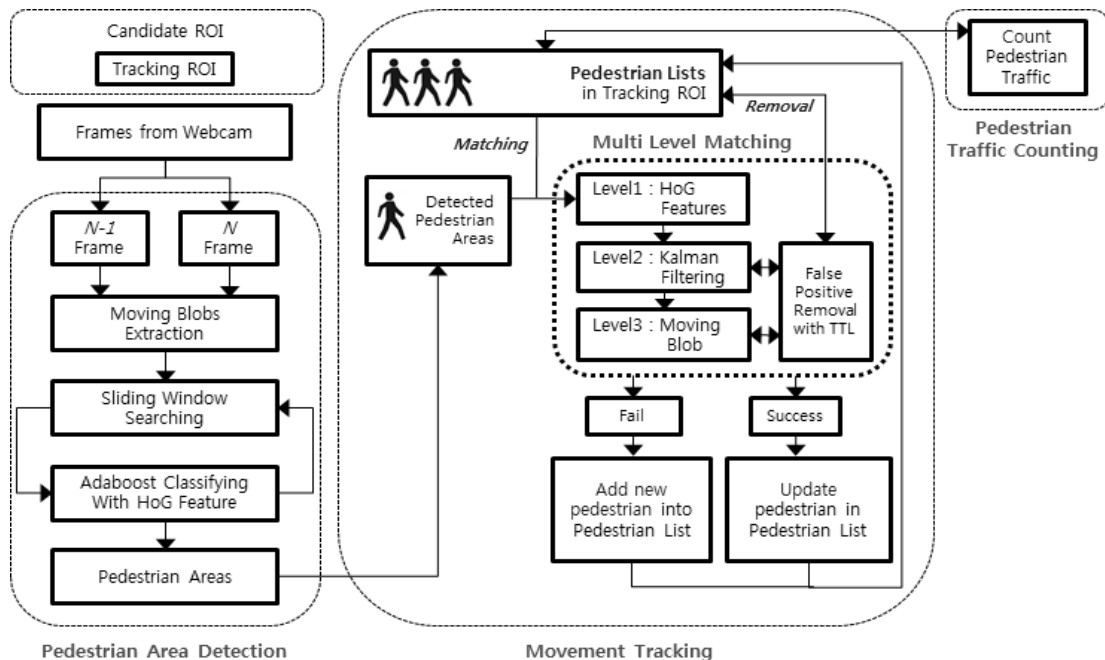


Fig. 2. The overall process of the proposed algorithm

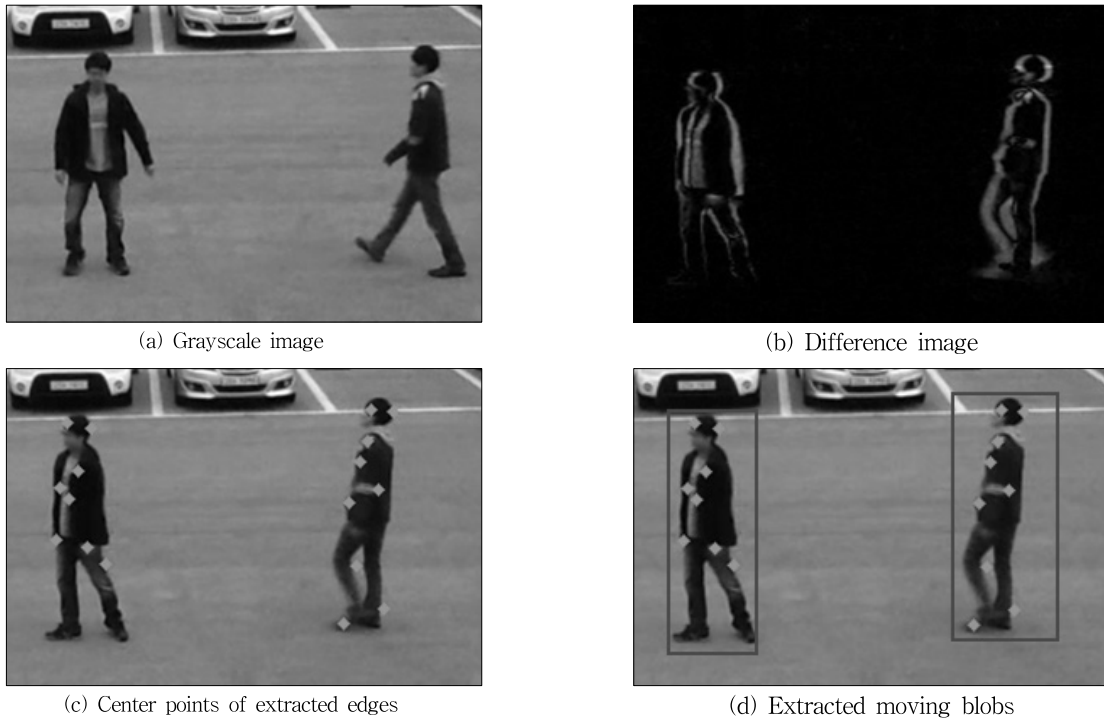


Fig. 3. Moving blob extraction from images

### 3.1.2 슬라이딩 윈도우 탐색

제안하는 알고리즘에서 사용한 사람에 대한 학습 영상은 크기가 64×128 픽셀 크기이며, 목적 카메라 영상은 640×480 픽셀이므로 입력 받은 영상에 대하여 선형 검색을 해야 할 필요가 있다. 선형 검색의 경우 한쪽 방향에서 이미지의 끝 부분까지 진행한다. 또한 카메라의 시야각이 있으므로, 카메라의 위쪽에서 나타나는 사람은 작게 보일 수 있어서 입력 받은 영상을 70%~130%까지 다른 비율로 스케일링한 7장 영상에 대하여 64×128 픽셀 크기의 윈도우로 사람 영역 탐색을 수행한다. 영상에 대하여 순차적으로 HoG 특징을 구하여 분류하는 과정에 POSIX 스레드를 사용하여 병렬 처리를 수행하였다.

### 3.1.3 사람 영역 탐지

추출한 움직임 영역에 대해 Fig. 4와 같이 HoG 특징과 Adaboost 분류기를 이용해 보행자를 탐지한다. HoG 특징은 경계선 검출에 강인한 성질을 보여 사람을 탐지하는데 적합하다. HoG 특징을 추출하기 위해서 목적 영상에 대해 광원에 의한 영상의 잡음을 감소시키기 위해 회색조 변환 및 히스토그램 평준화를 거친다. 그 후에 목적 영상을 가로 3픽셀, 세로 3픽셀 크기의 블록으로 나눈 후에 각 블록 내부의 픽셀이 대하여 Gradient의 방향성을 구한다. 분할한 각 블록에 대해 기울기를 기준으로 히스토그램을 생성하고 가장 큰 기울기 분포를 가진 기울기를 해당 블록의 평균 기울기로 선택한다. 생성된 각 블록들의 방향성 값과 크기 값을 선형으로 결합하면 HoG 특징 벡터가 된다. 이와 같은 특징 벡터를 Adaboost 분류기를 이용하여 학습시키고 입력된 목적

영상에 대하여 사람 영역과 비사람 영역의 두 가지 형태로 분류를 수행하였다. 학습을 위한 영상은 INRIA의 Pedestrian 이미지 데이터 셋에 포함된 약 2,000장의 보행자 영상을 사용하였다.

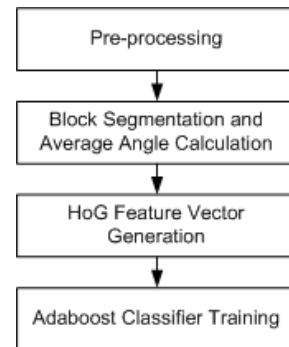


Fig. 4. HoG feature extraction and Adaboost classifier learning process

### 3.2 움직임 추적

움직임 추적은 탐지된 보행자의 영역에 대하여 멀티 레벨 매칭을 실시하여 보행자의 움직임을 추적하고 거짓 양성 제거를 수행하여 잘못 탐지된 보행자를 리스트에서 제거한다. 멀티 레벨 매칭에 있어서 가장 주요한 부분인 겹침 상황을 고려한 향상된 유사도 계산 방법은 3.2.1절에서 설명하고, 칼만 필터를 이용한 움직임 추정 방법은 3.2.2절에서 설명하며, 이와 같은 방법들의 결과를 이용한 멀티 레벨 매칭 과정을 3.2.3절에서 설명한다.

### 3.2.1 겹침 상황을 고려한 향상된 유사도 계산 과정

사람 영역 추적은 탐지된 보행자와 기존에 이미 탐지된 보행자와의 유사도를 측정한다. 유사도를 측정하여 동일임으로 판명되면 보행자의 위치 정보를 갱신한다. 보행자 위치 정보를 연결하면 보행자의 움직임 동선을 파악할 수 있다. 하지만 유사도 매칭이 연속적으로 잘못되면 제대로 된 탐지를 할 수 없다. 따라서 매칭 과정은 추적에서 아주 중요하다.

제안하는 알고리즘은 잘못된 탐지를 줄이기 위해서 멀티레벨 매칭을 수행한다. 멀티레벨 매칭의 1단계는 겹침 상황을 고려한 향상된 유사도 계산 과정이다. 향상된 유사도를 계산하기 위해 사용된 방법은 거리 변화율 유사도, 색상 히스토그램 유사도 및 겹침 처리이다[10, 11, 12].

거리 변화율 유사도는 사람이 매 프레임마다 이동할 수 있는 최대 거리는 제한적이기 때문에 유용하다. 한 명의 보행자에 대한 정보를 태그라고 표현하면, 이전 태그는  $T_{i_{prev}} = (x_{ip}, y_{ip})$ 라 하고, 다음 태그  $T_{i_{next}} = (x_{in}, y_{in})$ 라 할 때, 이전 프레임 및 다음 프레임  $i$ 번째 태그 간의 거리  $d_i$ 는 다음과 같이 구할 수 있다.

$$d_i = \sqrt{(x_{ip} - x_{in})^2 + (y_{ip} - y_{in})^2}$$

최소값을 갖는  $d_i$ 를  $d_{\min}$ 이라고 하고, 최종적으로 구하고자 하는 거리 변화율 유사도(거리 비율 계수)를  $D$ 라고 할 때  $D = d_{\min}/d_i$ 이다.

색상 히스토그램 유사도 계산 방법은 다음과 같다. 먼저 히스토그램 생성시 R, G, B 각 채널에 대해 60개의 히스토그램 빈을 생성하고 해당 빈의 개수를  $N$ 이라 하자.  $H_i$ 를  $i$ 번째 히스토그램이라 하고,  $\overline{H}_k = \frac{1}{N} \sum H_k(j)$ 라 하면  $d(H_1, H_2)$ 는 다음과 같다.

$$d(H_1, H_2) = \frac{\sum (H_1(l) - \overline{H}_1) \sum (H_2(l) - \overline{H}_2)}{\sqrt{\sum (H_1(l) - \overline{H}_1)^2 \sum (H_2(l) - \overline{H}_2)^2}}$$

물체 트래킹에서 가장 실패를 많이 하는 요인은 3차원 공간의 물체를 2차원으로 투영한 영상을 사용하여 트래킹하기 때문에 뒤에 있는 물체가 앞에 있는 물체나 장애물에 의해 겹칠 수 있다는 점이다. 겹침 상황을 감지하기 위하여 겹침 정도를 다음과 같이 수치화하였다[11, 12].  $i$ 개의 사람 윈도우  $W_i$ 에 대하여  $W_i = (x_i, y_i, w_i, h_i)$ 라 할 때, 겹침 계수  $C$ 는 다음과 같다.

$$C = \frac{(x_a + w_a - x_b)(y_a + h_a - y_b)}{w_a h_a + w_b h_b}$$

제안하는 알고리즘에는 총 4가지의 경우에 대해 겹침 계수를 계산하였으며 위 식은 그 중 한가지의 경우이다. 겹침

계수식의 분모는 겹치는 면적이고 분자는  $W_1$ 과  $W_2$ 의 전체 면적의 합을 나타낸다. 이 겹침 계수값이 임계치를 넘으면  $W_a$ 와  $W_b$ 는 겹침 상황에 있다고 판단한다. 제안하는 알고리즘에서는 임계값로 0.2를 사용하였다.

이와 같은 색상 히스토그램 상관계수  $C$ 와 거리변화율 계수  $D$ 를 가지고 다음과 같이 가중치를 주어 최종 유사도 계수  $S$ 를 계산한다.

$$S = (1 - \alpha)C + \alpha D$$

최종 유사도 계수  $S$ 는 0에서 1사이의 실수값을 갖는다. 알파값은 겹침 상황에 따라 다르며 상세한 값을 변화는 겹치는 경우 0.5를, 겹치지 않는 경우 0.3을 지정하였다. 제안한 방법에서는 알파값은 자체적으로 실험한 결과에 따라서 선택한 값이다. 이렇게 적용함으로써 겹치는 경우에는 거리 유사도 계수보다 컬러 히스토그램 유사도 계수의 가중치가 높아져 윈도우가 섞이는 빈도를 최소화 할 수 있다.

### 3.2.2 칼만 필터를 이용한 움직임 추정

멀티레벨 매칭 1단계의 최종 유사도 계수  $S$ 가 일정 수치 이하일 경우 매칭 실패로 간주하고 2단계로 넘어간다. 2단계에서는 칼만 필터를 이용해 움직임을 예측하여 그 위치에 대해 이미지 유사도 비교를 실시한다.

본 논문에서는 보행자들의 움직임은 일정속도로 움직이는 등속도 운동을 한다고 가정하였다. 보행자 운동 방정식은 2차원 평면상에서 운동이므로  $x_{i+1} = x_i + \Delta t v_x$ ,  $y_{i+1} = y_i + \Delta t v_y$ 이다. 본 논문에서 설정한 운동 모델에 따른 선형 칼만 필터 상태 벡터  $v_i$ 와 상태 전이행렬  $A$ 는 다음과 같다[13].

$$v_i = \begin{pmatrix} x_i \\ y_i \\ dx \\ dy \end{pmatrix} \quad A = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

제안한 방법에서는 이와 같은 특성을 갖는 선형 칼만 필터를 사용하였으나, 측정값이 일정 수준 이상 들어가지 않으면 예측값의 오차가 매우 크게 되므로 최초 5번의 단계까지는 예측값을 무시하였다.

### 3.2.3 멀티레벨 매칭

멀티레벨 매칭은 3단계로 이루어지며 앞서 계산한 최종 유사도 계수  $S$ 를 이용해 HoG 특징을 이용한 검출 윈도우 영역과 칼만 필터 예측 윈도우 영역, 인접 움직임 영역의 윈도우에 대하여 유사도를 단계적으로 측정한다.

1단계인 HoG 특징을 이용한 검출 윈도우 영역은 사람 영역 탐지에서 탐지된 보행자에 대한 영역에 기존에 보유하고 있는 보행자 리스트와 비교하여 최종 유사도 계수  $S$ 가 일정 임계치를 초과하면 매칭 성공으로 판단하고 위치 정보를 갱신한다. 제안한 방법에서는 실험적으로 결정된 임계치

0.8을 적용하였다.

2단계는 1단계에서 임계치를 넘지 못하는 영역에 대해서 적용한다. 먼저 선형 칼만 필터의 예측점을 중심으로 하여 윈도우를 생성한다. 생성된 윈도우의 이미지와 기준에 저장되어 있는 보행자 이미지와의 유사도를 계산하여 임계치를 초과하면 매칭 성공으로 판단한다.

3단계인 인접 움직임 영역을 이용한 과정에서는 2단계를 실패한 영역에 대하여 수행하며, 가장 가까운 움직임 영역의 중심점에 대해 윈도우를 생성하고 해당 위치의 이미지와 기준에 저장된 이미지와의 유사도를 계산한다.

3.2.4 거짓 양성 제거

상기의 알고리즘들을 사용하여 추적을 시도하지만 잘못된 HoG 특징 검출 영역으로 인하여 잘못된 윈도우들이 생기는 문제가 여전히 존재한다. 이러한 잘못된 윈도우를 제거하기 위해 Time To Live (TTL) 특성을 적용한 거짓 양성 제거 알고리즘을 적용한다. 즉 각 윈도우에 TTL 특성을 적용한 후에 값을 이미지 처리 횟수에 의존하여 감소시키며 0인 경우 윈도우를 삭제한다. HoG 검출 영역과 칼만 필터 예측 윈도우에 대하여 TTL을 적용하였다. 실제 사람인 영역을 연속해서 일정 횟수 동안 2단계 매칭에서 임계치 이하로 나타는 가능성이 아주 낮기 때문에 TTL를 감소할 수 있다. 이 TTL 정보는 각 태그마다 가지고 있으며 초기값은 15로 설정했다.

3.3 통행량 측정

Fig. 5와 같이 입력받은 웹캠 영상에 대하여 통행량을 측정할 영역을 Tracking ROI로 설정하고 그 외의 영역은 Candidate ROI라 하자. 통행량을 측정하는 위치는 Tracking ROI내의 보행자이다. Candidate ROI는 보행자 탐지를 하지만 추적을 하지 않는 후보 영역이다.

통행량 측정은 멀티 레벨 매칭이 성공한 후 Tracking ROI 보행자 리스트 안에 있는 각각의 태그들의 위치정보를 Tracking ROI 영역에서 Candidate ROI로 넘어갔는지 검사를 수행한다. Tracking ROI에서 Candidate ROI로 넘어간 태그들은 통행량으로 인지하여 카운팅을 하고, 리스트에서 태그 정보를 삭제한다.



Fig. 5. Tracking ROI and Candidate ROI in an image

4. 실험 결과

제안한 알고리즘의 성능 평가를 위하여 기존 연구들에서 범용적으로 사용하는 HoG 특징과 칼만 필터를 이용하는 사람 탐지 및 추적 알고리즘[1, 2]을 구현하였고 통행량을 측정하여 비교 분석을 수행하였다.

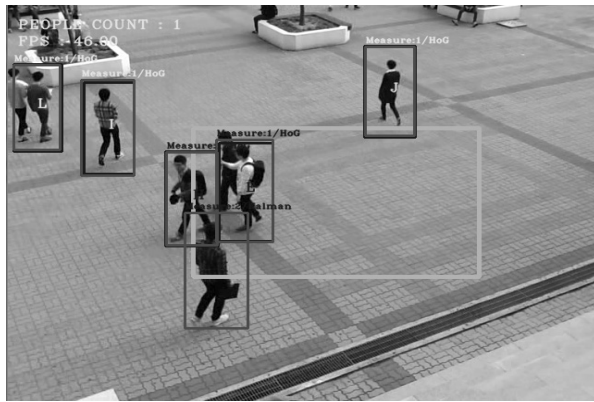
Fig. 6에서는 기존 연구들에서 범용적으로 사용하는 알고리즘과 제안하는 알고리즘의 차이점을 차별화하여 도시하였다. 기존 알고리즘의 경우 HoG 특징을 이용하여 사람을 탐지하고, 탐지된 영역에 보행자의 위치를 추적하기 위해서 단순 색상 유사도와 위치 보정을 위한 칼만 필터를 사용하여 추적 및 통행량 측정을 수행한다. 칼만 필터를 이용한 위치 보정의 경우 어느 정도의 데이터의 값이 축적되어 있어야 효과를 볼 수 있기 때문에 초기에는 색상 유사도만을 가지고 추적을 진행하게 된다. 따라서 초기에 비슷한 색깔의 사람들이 많을 경우 제대로 추적이 되기 힘들다. 그러나 제안하는 알고리즘은 색상과 거리 및 겹침 정도를 수치화한 최종 유사도 계수를 이용하여 멀티 매칭을 진행하기 때문에 이러한 문제점을 보완할 수 있다. 또한 겹침 상황에서 겹침의 정도를 수치화하여 최종 유사도에 활용한다.

Previous Algorithm		Proposed Algorithm	
Webcam Image		Webcam Image	
Pedestrian Detection using HoG Feature	Detection	Moving Blob Detection	Pedestrian Detection using HoG Feature
Tracking using Color Similarity and Kalman Filtering	Tracking	Multi-Level Matching with Occlusion, Color Histogram, Distance and Kalman Filtering	False Positive Removal
Pedestrian Traffic Counting	Counting	Pedestrian Traffic Counting	

Fig. 6. Comparison between previous algorithm and proposed algorithm

제안한 알고리즘의 성능 평가를 위해 사용된 보행자 영상은 3m 높이에 웹캠을 설치하여 2개의 서로 다른 장소에 대해 5분 간격으로 8회씩 촬영된 영상을 사용하였다. 장소 1은 장소 2보다 상대적으로 겹침 상황이 많이 발생하는 영상을 촬영하였다. 촬영된 2곳의 장소에 대한 영상의 예는 Fig. 7과 같다. 통행량 측정은 설정된 Candidate ROI에서 Tracking ROI로 진입한 후 빠져나올 때 측정된다.

Table 1은 각 장소에서 촬영한 영상에 대하여 실제 사람이 지나간 횟수를 사람이 일일이 체크한 실제 통행량 테이블이다. Table 2 및 Table 3에는 촬영된 장소 1과 장소 2에서의 기존 알고리즘과 제안하는 알고리즘의 정확도를 비교하여 제시하였다.



(a) Location 1



(a) Location 2

Fig. 7. 2 Locations for the experiment

Table 1. True pedestrian traffic count on location 1 and location 2

Number	Pedestrian traffic count	
	Location 1	Location 2
1	42	34
2	43	19
3	33	30
4	26	14
5	32	26
6	40	20
7	35	34
8	33	33
Total	284	210

Table 2. Pedestrian traffic counting result on location 1

Number	Previous algorithm	Proposed algorithm
1	25	30
2	34	34
3	24	29
4	21	22
5	28	30
6	26	32
7	22	28
8	19	26
Total	199	232

Table 3. Pedestrian traffic counting result on location 2

Number	Previous algorithm	Proposed algorithm
1	28	30
2	15	11
3	16	25
4	13	15
5	16	23
6	26	22
7	20	27
8	23	27
Total	157	180

Table 4에는 각 장소 1과 장소 2에 대하여 전체적인 평균을 구하여 나타낸 통행량 정확도이다. 평균적으로 기존 알고리즘의 경우 72.35% 정확도를 나타냈으나 제안하는 알고리즘은 83.65% 정확도를 나타내었고 약 11% 향상된 것을 확인할 수 있다.

Table 4. Detection accuracy in each location

	Previous algorithm		Proposed algorithm	
	Location 1	Location 2	Location 1	Location 2
Accuracy	70.0%	74.7%	81.6%	85.7%
Average	72.35%		83.65%	

### 5. 결론

현대 사회에서 창업 및 사업에 대한 성공을 위해서 상권 분석이 중요하며, 상권 분석의 요소 중에서 핵심적인 부분은 통행량이다. 이와 같은 통행량을 측정하기 위해서 전통적인 방법인 사람이 직접 측정하는 방법이 많이 사용되고 있으나 높은 인건비와 측정 실수를 유발할 가능성이 높다.

본 논문에서는 자동화된 알고리즘을 통하여 통행량을 자동 측정하는 알고리즘을 제안하였다. 제안하는 알고리즘은 기존 알고리즘과는 다르게 멀티 레벨 매칭과 거짓 양성 제거 기법을 추가로 사용하여 사람을 추적하고 통행량을 측정하였다. 제안한 알고리즘은 기존의 색상 유사도와 칼만 필터를 사용한 추적과 통행량 측정보다 약 11% 더 나은 성능을 보였으며 전체적으로 83.6%의 정확도를 보였다. 또한 겹침 계수를 이용한 최종 유사도 계산 방법을 사용함으로써 겹침 상황이 많이 발생하는 곳에서의 더 향상된 성능 보였다.

겹침 상황을 수치화하여 최종 유사도 계수에 반영하였지만 아주 혼잡한 거리나 겹침 상황이 50% 이상 일어나는 영역에 대해서는 HoG 탐지나 추적이 힘든 경우가 있었다. 겹침 상황에서도 정상적으로 보행자 탐지나 추적이 가능하도록 하는 추가적인 연구가 필요하다.

제안하는 알고리즘은 통행량 측정에서 사용되었지만 사람 영역 탐지나 추적에 대한 다른 응용 분야에서도 널리 사용될 수 있다.

### References

[1] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Vol.2, pp.886-893, 2005.

[2] N. Dalal, "Finding People in Images and Videos," Institut National Polytechnique de Grenoble, 2006.

[3] Y. Freund and R. E. Schapire, "A Short Introduction to Boosting," *Journal of Japanese Society for Artificial Intelligence*, Vol.14, No.5, pp.771-780, 1999.

[4] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian Detection: An Evaluation of the State of the Art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.34, No.4, pp.743-761, 2011.

[5] C.-J. Seo and H.-I. Ji, "Pedestrian Detection using HOG Feature and Multi-Frame Operation," *Journal of Korean Institute of Electrical Engineers (in Korean)*, Vol.64P, No.3, pp.193-198, 2015.

[6] S.-M. Yang and K.-H. Jo, "HOG based Pedestrian Detection and Behavior Pattern Recognition for Traffic Signal Control," *Journal of Institute of Control, Robotics and Systems (in Korean)*, Vol.19, No.11, pp.1017-1021, 2013.

[7] P. Viola, M. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," *International Journal of Computer Vision*, Vol.63, No.2, pp.153-161, 2005.

[8] K.-K. Kong and K.-S. Hong, "Design of coupled strong classifiers in AdaBoost framework and its application to pedestrian detection," *Pattern Recognition Letters*, Vol.68, No.1, pp.63-69, 2015.

[9] B.-R. Lee, Q.-B. Truong, H.-S. Kim, and Y.-H. Bae, "A study on the pedestrian detection on the road using machine vision," *Journal of Institute of Control, Robotics and Systems (in Korean)*, Vol.17, No.5, pp.490-498, 2011.

[10] Z. Jiang, D. Q. Huynh, W. Moran, S. Challa, and N. Spadaccini, "Multiple Pedestrian Tracking Using Colour and Motion Models," *Proceedings of International Conference on Digital Image Computing: Techniques and Applications*, pp.328-334, 2010.

[11] X. Wang, "An HoG-LBP human detector with partial occlusion handling," *Proceedings of International Conference on Computer Vision*, pp.32-39, 2009.

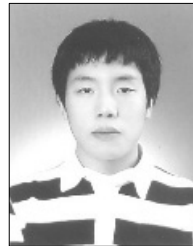
[12] A. M. Elgammal and L. S. Davis, "Probabilistic Framework for Segmenting People under Occlusion," *Proceedings of IEEE International Conference on Computer Vision*, Vol.2, pp.145-152, 2001.

[13] Wikipedia, "Kalman filter" [Internet], [http://en.wikipedia.org/wiki/kalman\\_filter](http://en.wikipedia.org/wiki/kalman_filter).



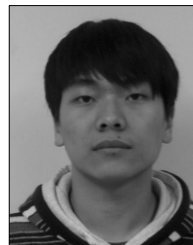
#### 강성욱

e-mail : hoingk@naver.com  
 2009년~2013년 금오공과대학교  
 컴퓨터소프트웨어공학과(학사)  
 2014년~현 재 고려대학교 정보보호대학원  
 석사과정  
 관심분야: 영상처리, 데이터 마이닝



#### 정진동

e-mail : jindongp@gmail.com  
 2007년~2015년 금오공과대학교  
 컴퓨터소프트웨어공학과(학사)  
 2016년~현 재 헨즈스튜디오 연구원  
 관심분야: 영상처리, 게임개발



#### 서흥일

e-mail : home1201@gmail.com  
 2010년~2013년 금오공과대학교  
 컴퓨터소프트웨어공학과(학사)  
 2014년~현 재 TeamSF 연구원  
 관심분야: 영상처리, 리눅스, 임베디드  
 프로그래밍



#### 이해연

e-mail : haeyoun.lee@kumoh.ac.kr  
 1997년 성균관대학교 정보공학과(학사)  
 1999년 한국과학기술원 전산학과(공학석사)  
 2006년 한국과학기술원 전자전산학과  
 전산학전공(공학박사)  
 2001년~2006년 (주)세트랙아이 선임연구원  
 2006년~2007년 코벨대학교 박사후연구원  
 2008년~현 재 국립금오공과대학교 컴퓨터소프트웨어공학과  
 교수  
 관심분야: 멀티미디어, 영상처리, 콘텐츠보안, 디지털위터마킹