

# 소형 이동 로봇의 사람 추적 성능 개선을 위한 휠 오도메트리 기반 실시간 보정에 관한 연구

## Real-Time Correction Based on wheel Odometry to Improve Pedestrian Tracking Performance in Small Mobile Robot

박재훈<sup>1</sup>·안민성<sup>2</sup>·한재권<sup>†</sup>

Jaehun Park<sup>1</sup>, Min Sung Ahn<sup>2</sup>, Jeakweon Han<sup>†</sup>

**Abstract:** With growth in intelligence of mobile robots, interaction with humans is emerging as a very important issue for mobile robots and the pedestrian tracking technique following the designated person is adopted in many cases in a way that interacts with humans. Among the existing multi-object tracking techniques for pedestrian tracking, Simple Online and Realtime Tracking (SORT) is suitable for small mobile robots that require real-time processing while having limited computational performance. However, SORT fails to reflect changes in object detection values caused by the movement of the mobile robot, resulting in poor tracking performance. In order to solve this performance degradation, this paper proposes a more stable pedestrian tracking algorithm by correcting object tracking errors caused by robot movement in real time using wheel odometry information of a mobile robot and dynamically managing the survival period of the tracker that tracks the object. In addition, the experimental results show that the proposed methodology using data collected from actual mobile robots maintains real-time and has improved tracking accuracy with resistance to the movement of the mobile robot.

**Keywords:** Multiple Object Tracking, Mobile Robot, Realtime

### 1. 서 론

물류 로봇, 배달 로봇으로서 산업적 성공<sup>[1]</sup>을 보인 이동 로봇은 산업적 사용을 넘어 다양한 모습과 방법으로 인간의 삶에 가까워 지고 있다. 한 사람만을 추종하는 반려 로봇의 제품화<sup>[2]</sup>, 사람 추종의 로봇 공연 콘텐츠화 등 많은 사례에서 로봇과 인간 간의 상호작용 방법으로 사람 추종 기법이 채택되고

있고, 이러한 경향에 맞춰 이동 로봇의 사람 추종에 관한 연구가 현재 활발히 진행되고 있다.

로봇이 사람을 추종하기 위해서는 다수의 사람이 검출되었을 때도 각 사람을 고유한 객체로 인식할 수 있어야 하고 이를 위해 다중 객체 추적 기술이 필수적이다. 하지만 다중 객체 추적을 소형 이동 로봇에 적용하는 데는 몇 가지 어려운 점이 존재한다. 첫번째 난관은 인식을 실시간으로 수행 해야한다는 점이다. 객체 추적 결과를 기반으로 제어 판단을 내리게 되는 이동 로봇에게 있어 실시간성은 최우선 지표이지만 기존 기법들은 대부분 실시간성을 보장하지 못한다는 문제점이 존재한다. 실제로 MOT Challenge<sup>[3]</sup>의 가장 최근 Benchmark인 MOT 20<sup>[4]</sup>의 결과를 살펴 보면, 정확도 성능 지표 상위 25 개의 기법들<sup>[5]</sup>은 모두 20 Hz 미만의 처리 속도를 보이며, 상위 32 개의 기법 중 20 Hz 이상의 속도를 보장하는 기법은 단 4개에 불과하다.

Received : Feb. 25. 2022; Revised : Mar. 22. 2022; Accepted : Mar. 22. 2022

※ This research was supported by the MOTIE (Ministry of Trade, Industry, and Energy) in Korea, under Global Talent Growth Support Project for Robot Industry Innovation (P0017311) supervised by the Korea Institute for Advancement of Technology (KIAT)

1. Principal Researcher, Hanyang University, Seoul, Korea (jh9277@hanyang.ac.kr)

2. Manager, UCLA, Los Angeles, CA (aminsung@ucla.edu)

† Associate Professor, Corresponding author: Hanyang University, Seoul, Korea (jkhan@hanyang.ac.kr)

이러한 결과를 고려했을 때 기존 다중 객체 추적 방법론 중 이동 로봇에 적합한 기법으로 SORT 알고리즘<sup>[7]</sup>이 존재한다. SORT는 2016년 처음 발표되어 가장 최근 MOT Challenge인 MOT 20 Benchmark 결과 정확도 측면에서 상위 32 위를 기록하였고, 처리 속도는 MOT Benchmark에서 가장 빠른 속도인 57.3 Hz를 기록 하였다. SORT는 빠른 처리 속도와 구조의 단순함으로 인한 높은 이식성을 강점으로 이동 로봇과 같은 임베디드 시스템에 많이 사용된다<sup>[8]</sup>. 하지만 SORT를 이동 로봇에 적용함에 있어 객체 추적 정확도 성능에 저하가 발생하는데, 그 이유는 아래 두번째 문제점과 같다.

이동 로봇이 움직이면 카메라의 시점 변화가 발생하고 이로 인한 입력 프레임 변화는 SORT와 같은 IOU tracker<sup>[9]</sup>기반의 다중 객체 추적 기법들의 성능을 현저히 저하시키게 된다. 특히 SORT는 일시적인 객체 검출 실패에 대처하지 않기 때문에 입력 프레임의 일시적인 변화로 인한 영향을 크게 받게 된다. 이러한 문제점을 보완 해내기 위해 객체의 특징을 고려하는 방법도 존재하지만 이로 인한 연산 시간 증가는 피할 수 없는 문제이다<sup>[10]</sup>.

이러한 문제들 때문에 기존의 다중 객체 추적 기법들은 실시간성을 유지하고 안정성 있게 사람을 추적해야 하는 사람 추종 이동 로봇에 적합하지 않다. 본 논문에서는 이러한 기존 기법들의 한계를 극복하고 제한적 연산 능력을 지닌 소형 이동 로봇에 적용 가능한 방법론을 제시한다. 이동 로봇의 휠 오도메트리를 이용하여 로봇의 회전 정도를 추정된 후 이를 통해 실시간으로 사람 추적 알고리즘을 보정하는 방법에 대해 소개하며, 더 나아가 안정적인 추적 유지를 위한 객체 추적기의 생존 기간 동적 설정 방법을 제시한다.

## 2. 관련 연구

### 2.1 Pedestrian Tracking for Mobile Robot

이동 로봇에 사람 추적 기법을 적용시키기 위한 고전적인 방법으로서 거리 측정 센서를 사용한 연구들이 주를 이루어왔다. 레이저 센서 기반의 연구는 사람의 다리를 인식하여 이를 통해 사람의 이동 궤적을 유추하는 방법론<sup>[11]</sup>과 다수의 레이저 센서 기반의 Multiple Hypothesis Tracking (MHT)<sup>[12]</sup>를 사용하여 사람 추적을 수행하는 연구<sup>[13]</sup>가 수행되어 왔다.

카메라 센서를 사용한 방법론으로는 스테레오 카메라를 사용하여 사람 추적을 수행한 연구<sup>[14]</sup>와 RGB-D 카메라를 사용한 연구<sup>[15,16]</sup>가 존재하는데 처리 속도 면에서 높은 성능을 보여주고 있다. 하지만 이러한 거리 측정을 이용한 기법들은 거리 측정 센서의 인식 한계를 넘는 객체는 추적하지 못하는 한

계가 존재하고, 해당 센서에 대한 의존성으로 인해 소형 이동 로봇에 적용하기에 어려움이 존재한다.

현재는 딥러닝 특히 convolutional neural network 기반 비전 인식 기술의 발전으로 visual object tracking 기법들이 고전적 방법론들보다 높은 정확도 성능을 보여주고 있다<sup>[17]</sup>. 이러한 기법들은 카메라의 이미지만을 입력으로 사용하므로 특정 거리 측정 센서에 대한 의존성이 없기 때문에 소형 로봇에 일반적으로 적용 가능하고 이에 관련된 많은 연구가 이루어지고 있다<sup>[18-20]</sup>.

### 2.2 Visual Object Tracking

Visual Object Tracking은 주어진 프레임 안에서 객체를 검출하고 각 객체마다 식별 가능한 상태를 유지해야 하는 태스크이다. 추적을 위해선 먼저 추적하고자 하는 객체의 정보를 제공받아야 하고 이를 검출 모델을 통해 얻으면 Detection-based trackers, 검출 모델을 사용하지 않으면 Detection-free trackers로 구분된다. Detection-free trackers의 경우 처음 시점에 추적해야 할 목표물을 정해주고 이를 기반으로 다음 프레임에서의 추적을 수행하기 때문에 학습되지 않은 객체를 추적할 수 있다는 장점이 있다. 또한 일반적으로 많은 연산량을 요구하는 Detection 모델을 적용하지 않으므로 계산적 이점 또한 존재한다. 하지만 Detection-free trackers는 짧은 시퀀스에 한해서만 높은 성능이 보장되고, 단 하나의 객체를 추적하는 SOT (Single Object Tracking)에 적합한 방법론이다.

CNN 기법의 발전과 이를 기반한 객체 검출 모델의 발전으로 많은 객체 검출 모델이 높은 수준의 정확도를 가지게 되었다. 이를 기반으로 Detection-based trackers는 MOT (Multiple Object Tracking) 영역에서 높은 성능을 거두고 있다. MOT를 수행하는 Detection-based trackers는 Detection-free trackers에 비해 긴 시퀀스에서도 높은 성능을 보이며, 다수의 객체를 추적하고 있기 때문에 알고리즘의 확장성도 높아 사람 추적에 적합하다. 한편 MOT 분야에서는 SOT 분야에서 Siam tracker<sup>[21]</sup>나 CF tracker<sup>[22]</sup>의 계열의 방법론이 최고의 성능을 내는 한편 MOT 분야는 최고의 성능을 내는 명백한 알고리즘이 존재하지 않고 있다. 특히 알고리즘의 복잡도가 성능 향상으로 이어지지 않는 경향을 보이는데 성능 좋은 객체 검출기를 사용한 간단한 IOU tracker<sup>[9]</sup>가 좀 더 복잡한 알고리즘보다 좋은 성능을 내곤 한다.

### 2.3 SORT (Simple Online Real-time Tracking)

SORT는 Kalman Filter와 Hungarian 알고리즘을 기반으로

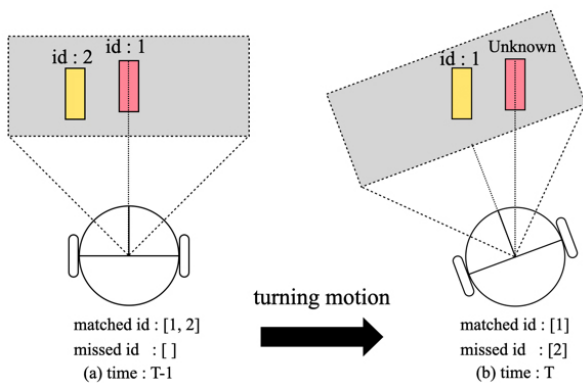
하는 온라인 실시간 추적 기법으로 추적 중인 각 객체의 움직임을 선형 등속 모델로 근사화하여 예측한다. SORT의 객체 추적 과정은 아래와 같다.

1. 객체 검출 모델을 통해 프레임 상 모든 객체의 위치 검출
2. Kalman Filter를 통해 추적 중인 객체의 다음 위치값 예측
3. 객체 검출값과 예측값 간의 IoU 점수를 기반으로 Hungarian 알고리즘을 사용하여 매칭
4. 매칭 결과를 통해 각 객체에 고유의 id 값을 부여하고, Kalman Filter 업데이트

SORT는 현재 프레임과 바로 직전 프레임 간의 매칭에 바운딩 박스 위치 정보 외 다른 부수적인 정보들은 무시하고 오직 객체 검출 결과값만 사용한다. 추적에 객체의 특징을 사용하지 않기 때문에, 목표물이 가려지거나 겹쳐지는 상황으로 인한 일시적인 검출 실패가 추적 실패로 이어진다.

또한 로봇의 움직임으로 인해 카메라 시점이 변경되어 객체 검출값이 이동했을 경우 추적 예측값과의 매칭이 어려워진다. [Fig. 1]은 로봇의 움직임에 의해 객체 추적에 실패하는 예시이다. 이러한 현상은 낮은 연산 처리 속도로 인해 연속된 프레임 사이의 시간 간격이 비교적 긴 소형 이동 로봇에게는 치명적인 문제가 될 수 있다.

객체의 특징 정보를 포함하지 못해 생기는 위와 같은 문제들을 보완한 방법론으로 객체 검출 모델 단의 feature map에 기록된 목표물의 특징점을 매칭에 반영하는 DeepSORT<sup>[23]</sup>가 존재하지만 DeepSORT는 SORT와 비교해 처리 시간이 3 배 정도 증가하였지만, 정확도면에서는 SORT의 59.8에서 61.4로 눈에 띄는 성능 향상을 이뤄내지는 못했다<sup>[23]</sup>.



[Fig. 1] Example of object tracking failure due to rotation of mobile robot: when there are two objects that remain stationary, the objects tracked in T-1 (a) are assigned ids 1 and 2. The robot rotates between T-1 and T (b), and the object assigned 1 fails to track in T, and object 2 is incorrectly tracked as object 1 in T

## 2.4 카메라 움직임 측정을 위한 연구

움직이는 카메라로 인한 객체 추정 성능 저하 문제를 해결하기 위해선 카메라의 움직임 정도를 추정해야 한다. 카메라의 움직임을 비전 정보를 통해 추정하는 방식인 Visual Odometry (VO)<sup>[24]</sup>는 스테레오 카메라를 사용하는지 단안 카메라를 사용하는지에 따라 Stereo Visual Odometry (SVO)와 Monocular Visual Odometry (MVO)로 나눌 수 있다.

VO기법들은 주변 환경이 어둡거나 인식에 방해가 되는 요소에 의해 성능 저하가 발생할 수 있으며, 프레임 상 다수의 특징점들을 연속된 프레임끼리 비교하는 방식으로 작동하기 때문에 필요한 연산량이 많다. 실제로 소형 시스템 적용을 위해 제안된 ARM-VO<sup>[25]</sup> 기법도 Raspberry Pi 3에 적용시킬 경우의 처리 속도는 8 Hz밖에 되지 못한다.

이처럼 높은 연산량을 필요로 하는 VO방식보다 휠 엔코더를 기반으로 한 오도메트리 계산 방법이 소형 이동 로봇에 더 적합하다. 휠 엔코더를 기반으로 계산하여 이를 통해 카메라의 움직임을 추정하는 방법은 VO처럼 모든 이동 카메라 사례에 범용적으로 적용할 수 없고, 이동 로봇 바퀴의 슬립으로 인한 측정 오차가 발생할 수 있지만, VO에 비해 적은 연산량을 통해 카메라 움직임 추정 결과를 얻어낼 수 있다.

## 3. 방법론

### 3.1 이동 로봇의 하드웨어 구성 및 휠 오도메트리 측정

본 연구에서 사용된 이동 로봇 EDIE의 하드웨어는 [Fig. 2]와 같다.

EDIE는 2 개의 휠을 기반으로 한 differential drive robot으로 로봇의 휠 반지름은 0.035 m, 휠 간 거리는 0.108 m, 엔코더 분



[Fig. 2] EDIE hardware CAD model and physical hardware

해능은 374 pulse per rotation이다. 사용된 MCU 는 Raspberry Pi 4이며, 병렬 연산 처리를 위한 VPU인 Intel 사의 Neural Compute Stick 2이 사용된다.

로봇의 오도메트리는 출발 지점과 로봇 간의 상대적 위치를 일반적으로  $x, y, \theta$  정보로 표현하지만, 카메라 프레임 기준 1 Hz 사이의  $x, y$ 의 변화량은  $\theta$ 의 변화량에 비해 미세하므로 본 연구에선  $x, y$ 에 대한 보정은 고려하지 않았다. 실제 연구에 사용된 로봇의 경우 각 모터의 속도가 1 m/s 일 때 20 Hz 주기로 카메라 프레임을 업데이트 할 경우  $x$ 는 최대 0.05 m의 변화량을 가질 수 있지만 회전의 경우 좌우 휠을 반대로 회전한다면 최대 53.05°의 변화량을 가질 수 있다.

카메라가 로봇에 고정되어 있을 경우 카메라의 움직임은 로봇의 오도메트리에 변환 행렬을 적용하여 구할 수 있고 로봇 오도메트리의 회전 각도는 휠 엔코더 값을 사용하여 식 (1) 과 같이 계산 되었다.

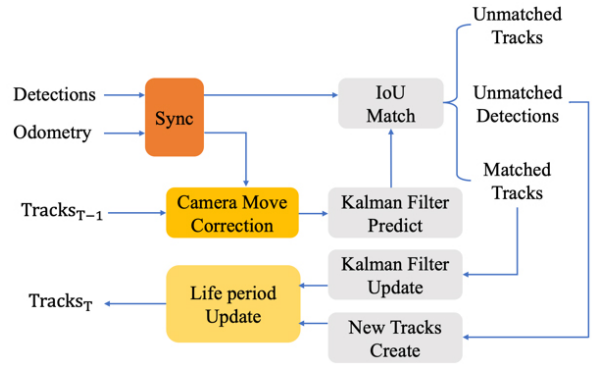
$$\Delta\theta = \frac{\Delta Distance_{Right} - \Delta Distance_{Left}}{wheel Track} \quad (1)$$

휠 엔코더를 통해 계산된 오도메트리를 사용해 카메라 프레임을 보정하기 위해서는 오도메트리와 카메라 간의 동기화 작업이 필요하다. 이를 위해 카메라에서 실시간 화면만을 가져올 수 있도록 카메라 버퍼 사이즈를 1로 변경하였으며 멀티 쓰레드를 사용해 메인 프로그램과 비동기적으로 작동시켰다. 또한 오도메트리 정보와 측정 시 time stamp를 버퍼에 저장하여 카메라 프레임과의 동기화를 진행하였다.

### 3.2 카메라 회전을 반영한 객체 추적의 실시간 보정

본 보정의 주요 목표는 객체의 다음 위치를 예측하는 Kalman Filter의 선형 예측 모델이 카메라의 움직임을 제외한 객체의 움직임만을 인지하도록 하기 위해서다. 이는 카메라의 움직임에도 추적을 유지하고, 일시적인 객체 검출 실패에 대한 저항력을 가질 수 있는 토대가 되어줄 수 있다. 보정 과정을 포함한 사람 추적 기법의 흐름도는 [Fig. 3]와 같다.

[Fig. 3]에서 확인할 수 있듯이 카메라 움직임에 대한 보정은 동기화된 오도메트리를 기반으로 Kalman Filter의 Predict 단계 직전에 수행된다. Kalman Filter는 Predict단계에서 선형 등속 시스템에 따라 객체의 다음 상태를 예측하게 하고, Update 단계에서 매칭된 실제 객체 검출값과 예측값 사이의 오차를 기반으로 예측 오차 공분산과 상태 변수를 보정하게 된다. 카메라 움직임에 대한 보정 이후에 Kalman Filter의 Update가 이뤄지므로 예측 오차 공분산과 상태 변수는 객체의 온전한 움직임만을 가지고 갱신될 수 있다. 이 과정에서 사용되는 Kalman



[Fig. 3] Flowchart of person tracking algorithm including real-time rotation correction

Filter 상태 변수는 식 (2) [7]와 같다.

$$State = [x, y, s, r, \dot{x}, \dot{y}, \dot{s}]^T \quad (2)$$

이 중 주목해야할 값은  $x$ 와  $\dot{x}$ 로  $x$ 는 객체 중심 좌표의 가로 픽셀 값을 나타내며,  $\dot{x}$ 는 객체의 속도값을 나타낸다. 회전 움직임 보정은 프레임 상으로 가로 축의 보정에 해당하며 현재 시점과 이전 시점의 카메라 회전 각 차이만큼 식 (3)으로  $x$ 를 보정해 준다.

$$x' = x + (\theta_T - \theta_{T-1}) * \frac{frame\_width}{fov} \quad (3)$$

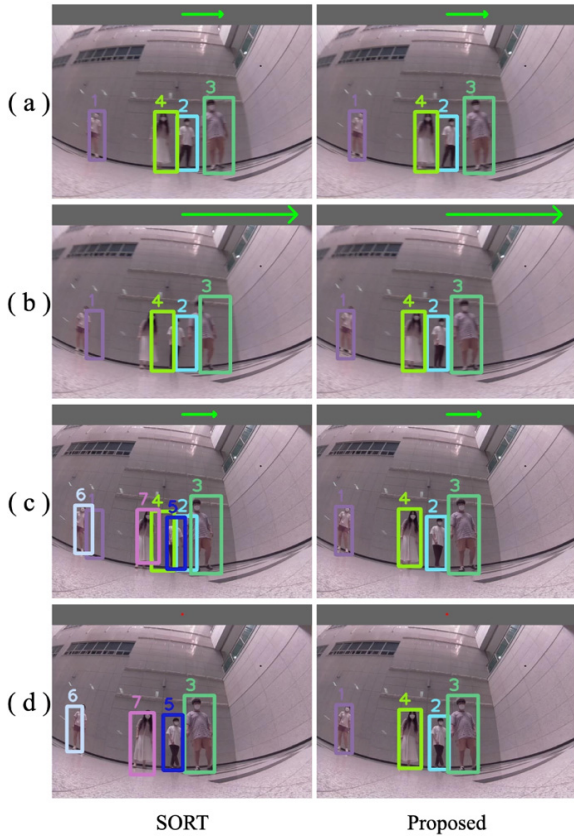
frame\_width는 프레임의 가로 픽셀 너비를 fov는 카메라의 가로 화각을 나타낸다. 보정된  $x$  값을 기반으로  $\dot{x}$ 가 계산되므로 객체의 온전한 움직임만이 속도 계산에 포함된다. 이러한 실시간 보정의 적용 효과는 [Fig. 4]에서 확인할 수 있다.

[Fig. 4]는 이동 로봇이 시계 방향으로 회전하고 있는 상황으로 연속된 4개의 프레임에서 4명의 사람이 지속적으로 검출되고 있다. SORT의 경우 객체 검출이 모든 프레임에서 성공했음에도 회전으로 인한 오차로 인해 (b)에서부터 예측값과 검출값 매칭에 실패해 결국 (d)에선 4개의 객체 중 3개의 객체 id가 전환되었다. 하지만 제안된 기법의 경우 이동 로봇의 움직임에도 객체의 다음 위치를 안정적으로 예측하여 추적 유지에 성공하는 모습을 보여주었다.

### 3.3 객체 추적기의 생존 기간 동적 설정

다중 객체 추적기는 객체 검출 결과와 예측값의 매칭이 실패할 경우 객체의 추적과 예측을 얼마나 오래 유지할지 결정해야하고, 이러한 객체 추적기의 생존 기간 설정은 연산의 효율성과 추적 안정성을 고려해서 설정되어야 한다.





[Fig. 4] Object tracking prediction results during clockwise rotation

SORT는 추적하던 객체가  $T_{Lost}$  동안 검출되지 않으면 추적을 유지할 필요가 없다 판단하고 추적을 중지한다. 또한 모든 실험에서  $T_{Lost}$  를 1로 설정됐다. 이러한 설정의 이유는 SORT가 선형 모델의 예측을 신뢰하지 않고 재식별 문제는 다루지 않기 때문이다. 이 때문에 매칭에 실패한 객체 검출기를 빠르게 삭제하면서 계산적 효율성은 증가 할 수 있지만 일시적인 검출 실패에도 추적 유지에 실패할 수 밖에 없게 된다. 실시간성을 보장하기 위해 비교적 낮은 검출 성능을 가진 객체 검출 모델을 사용하는 경우 이러한 설정은 잦은 객체 추적 실패로 이어질 수 있다.

하지만 제안된 기법의 경우 카메라의 움직임 보정을 통해 예측값의 신뢰도를 높였고, 이를 기반으로 객체 추적기의 생존 기간을 늘릴 수 있다. 또한 객체 추적을 오래 유지할수록 선형 모델의 예측 성능이 높아짐을 반영해주기 위해 기존 정적이던 객체 추적기의 생존 기간을 추적 유지 기간에 비례해 증가하도록 설정해주었다.

$T_{Hits}$  동안 추적을 유지했을 경우 추적기의 생존 기간  $T_{Age}$  는 식 (4)으로 계산된다.

$$T_{Age} = \min \left( T_{Lost} + \frac{T_{Hits}}{R}, \max\_age \right) \quad (4)$$

$\max\_age$ 는 생존 기간 제한을 위한 최대치를 나타내며,  $R$ 은  $T_{Hits}$  를 어느정도 반영할 지를 결정하는 변수이다.

객체 추적기의 생존 기간을 예측 신뢰도에 따라 동적으로 설정함으로 얻을 수 있는 성능 향상은 모션 블러 현상이 발생할 경우에 더 두드러진다. 이동 로봇에서 모션블러 현상은 빈도 높게 벌어지며, 모션 블러로 인한 객체 추적 성능 저하 현상은 [Fig. 5]에서 확인할 수 있다.

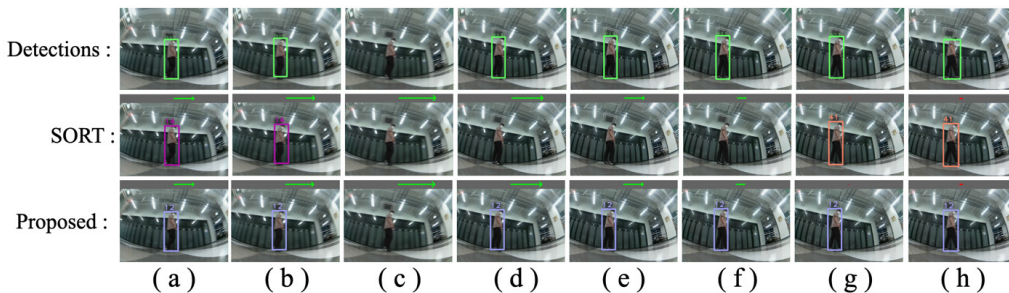
[Fig. 5]의 (c) 시점에서 객체 검출에 실패하게 되는데, 이때 객체 추적기는 검출값과의 매칭을 통한 업데이트를 하지 못하고, 오직 Kalman Filter로 객체의 다음 위치를 예측하는 과정만 진행하게 된다. 예측 과정만을 진행하는 상황에서 또다시 움직임으로 인해 카메라 시점이 변경되고, 이 때문에 SORT 는 추적 유지에 실패해 (g) 시점에 객체의 id가 바뀌는 모습을 확인할 수 있다.

이에 반해 제안된 기법을 통한 추적기는 안정적으로 다음 위치를 예측함으로써 객체가 다시 검출되는 (d) 시점에 추적을 성공적으로 유지하는 모습을 확인할 수 있다.

## 4. 실험

### 4.1 테스트 데이터

다중 객체 추적 기법의 성능 평가 기준으로 MOT Challenge



[Fig. 5] Object tracking result when object detection fails due to motion blur

[Table 1] Overview of the PxRx sequences

Sequence	FPS	Resolution	Length	Boxes	$th_{avg}$
PmRr	15	640x480	266 (00:18)	899	3.076°
PsRm	15	640x480	366 (00:24)	1169	2.42°
PmRm1	15	640x480	460 (00:30)	1442	2.369°
PmRm2	20	640x480	449 (00:22)	1461	2.091°
PoRt1	20	640x480	739 (00:37)	2698	0.858°
PoRt2	20	640x480	563 (00:28)	1712	1.034°

데이터가 표준적으로 쓰이고 있다. 하지만 해당 데이터는 카메라 이동에 대한 정보를 포함하고 있지 않아, 논문에서 제안하는 방법론의 성능 평가를 위해선 오도메트리 정보를 포함한 테스트 데이터를 구축해야 했다.

데이터는 이동 로봇에 부착된 카메라에서 수집 되었으며, 각 프레임마다의 카메라 오도메트리 정보와 객체들의 위치 정보를 담고 있다. 객체들의 ground truth 위치 데이터는 MOT 20 Challenge 형식으로 수집되었으며, 오도메트리 데이터는 x, y, theta와 해당하는 frame id를 포함하고 있고 frame id를 기준으로 ground truth 위치 데이터와 동기화된다.

테스트 데이터는 총 6시퀀스로 수집되었으며 각 시퀀스에 대한 정보는 [Table 1]에서 확인할 수 있다. [Table 1]의  $th_{avg}$  는 이동로봇의 프레임당 평균 회전 각을 나타내며 이를 통해 시퀀스 동안 로봇이 한번에 얼마나 많이 회전했는지 경향성을 확인할 수 있다.

## 4.2 평가 지표

다중 객체 추적 기법의 성능 평가를 위해 아래와 같은 지표를 사용하였다.

- $MOTA(\uparrow)$  : Multi-Object Tracking Accuracy.<sup>[3]</sup>
- $MOTP(\uparrow)$  : Summary of overall tracking precision.<sup>[3]</sup>
- $IDF_1(\uparrow)$  : The ratio of correctly identified detections over the average number of ground-truth and computed detections.<sup>[26]</sup>
- $IDP(\uparrow)$  : Identification precision.<sup>[26]</sup>
- $IDR(\uparrow)$  : Identification recall.<sup>[26]</sup>
- $FP(\downarrow)$  : number of false detections.<sup>[3]</sup>
- $FN(\downarrow)$  : number of missed detections.<sup>[3]</sup>
- $IDsw(\downarrow)$  : Number of times the reported identity of a ground-truth track changes.<sup>[3]</sup>
- $Hz(\uparrow)$  : Processing speed (in frames per second excluding the detector)

( $\uparrow$ )로 표시된 지표는 높은 값일수록 성능이 높다는 의미이고 반대로 ( $\downarrow$ )로 표시된 지표는 낮은 값일수록 높은 성능을 의미한다.

## 4.3 객체 검출 모델

객체 추적은 객체 검출 결과값을 토대로 이뤄지므로 객체 검출 성능에 큰 의존성을 가지고 있고, 이 때문에 적절한 객체 검출 모델 선정은 객체 추적이 있어 매우 중요한 문제이다.

CNN 기반의 다양한 객체 검출 모델이 존재하지만 제한된 연산 능력으로도 실시간으로 추론이 가능한 1-stage 기법의 객체 검출 모델인 YOLO<sup>[27]</sup>와 SSD<sup>[28]</sup> 계열의 모델이 이동 로봇에 적합하다.

YOLO와 SSD 계열의 객체 검출 모델인 YOLOv4<sup>[29]</sup>, YOLOv4-tiny<sup>[30]</sup>, SSDlite<sup>[31]</sup>의 객체 검출 정확도를 PxRx 데이터로 측정하였고, YOLOv4, YOLOv4-tiny, SSDlite 순으로 YOLOv4가 가장 높은 정확도를 보여주었다. 본 실험에서는 정확도와 처리 속도 측면을 모두 고려하여 YOLOv4-tiny 모델을 객체 검출 모델로 선정하여 사용하였다.

## 4.4 객체 추적기의 생존 기간 동적 설정 실험

객체 추적기의 생존 기간은 추적기의 예측값을 얼마나 신뢰하는가로 결정된다. SORT의 경우 frame-to-frame만을 고려한 설계이고, 카메라의 움직임이 존재 할 경우 예측의 오차는 더욱 커질 수밖에 없기 때문에  $T_{Lost}$  을 1로 설정했을 때 가장 높은 성능을 보여주었다.

실제로 이동 로봇에서 수집한 테스트 데이터를 통한 실험결과인 [Table 2]을 확인해보면  $T_{Lost}$  를 1로 설정하였을 경우 50.77의 정확도(MOTA)를 보였고,  $T_{Lost}$  를 2, 3, 4로 설정하였을 경우 순서대로 49.63, 49.63, 49.83로 정확도 하락을 보여주었다.

이에 반해 카메라의 움직임 보정을 통해 예측값의 신뢰도를 높인 후 객체 추적기의 생존 기간을 동적으로 설정한 경우 추적 정확도가 상승하였다.

[Table 2] Performance evaluation table according to the survival period setting of the object tracker

Method	$T_{Lost}$	$MOTA \uparrow$	$FN \downarrow$	$IDSW \downarrow$
SORT	1(Default)	50.77	4332	274
SORT	2	49.63	4446	266
SORT	3	49.63	4447	265
SORT	4	49.83	4437	<b>258</b>
SORT	Dynamic	<b>51.93</b>	<b>4226</b>	268

[Table 3] Tracking performance test results on MOT17

Method	$T_{Lost}$	MOTA ↑	MOTP ↑	FN ↓	IDSW ↓
SORT	1	49.01	<b>85.83</b>	51343	<b>1043</b>
SORT	Dynamic	<b>49.97</b>	85.65	<b>49997</b>	1051

카메라가 정면을 바라보고 있는 상태에서 수집된 MOT17-02, 05, 09, 10, 11 데이터<sup>[32]</sup>를 통한 실험 결과인 [Table 3]를 보면 객체 추적기의 생명 기간을 동적으로 설정해준 간단한 작업으로 추적 정확도(MOTA) 성능이 0.96% 증가한 것을 확인할 수 있다.

#### 4.5 실험 결과

다중 객체 추적 ground truth 정보와 오도메트리 정보를 포함하는 시퀀스를 통해 객체 추적 성능 측정 실험을 진행하였다. [Table 4]에선 SORT, SORT+Dynamic(생존 기간 동적 설정만 적용한 경우), SORT+Odom(움직임 보정만 적용한 경우) 그리고 Proposed(움직임 보정과 생존 기간 동적 설정 둘다 적용한 경우) 을 수집한 테스트 데이터를 통해 추적 성능 비교 평가를 진행하였다.

SORT+Odom의 경우 모든 지표에서 SORT의 성능을 높은 차이로 넘어섰다. 특히 추적하던 객체의 식별값이 변경되는 Idsw지표의 경우 277에서 180로 줄어 눈에 띄는 성능 향상을 보였는데 이는 제안된 보정 기법을 통해 움직이는 상황에서의 추적 안정성이 향상되었음을 의미한다.

SORT+Dynamic의 경우 MOTA 측면에서 SORT 대비 2.49의 낮은 성능 향상만 보였지만, 카메라 움직임 보정과 같이 적용시킨 Proposed는 SORT+Odom 대비 4.16의 성능 향상을 보여주어 성능의 향상폭이 증가했다. 이를 통해 생존 기간 동적 설정 방법은 추적기의 예측 성능이 상승할수록 더 높은 성능 향상폭을 보여주는 것을 확인할 수 있다.

[Table 4] Tracking performance test results on PxRx sequences

Method	MOTA ↑	MOTP ↑	FP ↓	FN ↓	Hz ↑
SORT	49.44	81.50	13	4453	<b>446.3</b>
SORT+Dynamic	51.93	81.34	15	4226	429.6
SORT+Odom	53.42	<b>82.23</b>	<b>11</b>	4179	440.6
Proposed	<b>57.58</b>	81.90	14	<b>3813</b>	420.7

Method	IDF <sub>1</sub> ↑	IDP ↑	IDR ↑	ID sw ↓	Hz ↑
SORT	29.86	43.27	22.79	277	<b>446.3</b>
SORT+Dynamic	37.62	52.94	29.18	268	429.6
SORT+Odom	39.19	54.86	30.49	180	440.6
Proposed	<b>46.02</b>	<b>61.68</b>	<b>36.70</b>	<b>152</b>	420.7

전체적인 추적 정확도를 의미하는 MOTA 외에도 대부분의 지표들이 두 기법을 같이 사용할 때 가장 높은 성능을 보여주고 있으며 정확도와 안정성 측면에서의 의미 있는 성능 향상을 보이는 동안 처리속도는 420.7 Hz 즉 2.38 ms로 SORT의 2.24 ms와 비교하여 매우 적은 속도 저하만이 발생하였다. 실험된 다중 객체 추적 기법들은 1.5 GHz Cortex-A72 싱글코어와 4 GB의 메모리로 구성된 시스템 위에서 실행되었다.

## 5. 결론

본 논문에서는 이동 로봇의 움직임으로 생기는 객체 추적 오차를 휠 엔코더 오도메트리 정보를 사용해 실시간으로 보정하고, 객체 추적기의 생존 주기를 동적으로 관리함으로써 소형 이동 로봇의 사람 추적 성능을 향상시키는 다중 객체 추적 방법을 제안했다. 방법론의 성능 평가를 위해 이동 로봇에서 오도메트리 정보를 포함한 6 개의 시퀀스 데이터를 수집하였고, 객체 검출기로 yolov4-tiny를 사용할 경우 2.38 ms의 처리속도로 실시간성을 유지한채 MOTA 측면에선 SORT보다 16.46%의 성능 향상을 보임을 실험적으로 증명하였다. 또한 추적하고 있는 객체의 예측 모델 신뢰도에 따라 생존 주기를 동적으로 관리해줌으로써 객체의 식별자가 전환되지 않고 긴 시간 동안 유지해 추적 안정성이 향상됨을 확인할 수 있었다. 이동 로봇과 사람 간의 상호 작용에 대한 시도가 활발해지고 있는 현 시점에서 본 논문에서 제안하는 기법을 통해 보다 정확하고 안정적으로 사람을 추적함으로써 사람 추종을 통한 로봇과 인간 간의 신뢰도 높은 상호 작용이 가능해질 것이라 예상된다.

## References

- [1] R. Bogue, "Growth in e-commerce boosts innovation in the warehouse robot market," *Industrial Robot*, 2016, DOI: 10.1108/IR-07-2016-0194.
- [2] N. Hitti. *Ballie the rolling robot is samsung's near-future vision of personal care*, 2020, [Online], <https://www.dezeen.com/2020/01/08/samsung-ballie-robot-ces-2020/>.
- [3] L. Leal-Taixe, A. Milan, I. Reid, S. Roth, and K. Schindler, "Motchallenge 2015: Towards a benchmark for multi-target tracking," *arXiv preprint arXiv:1504.01942*, 2015, DOI: 10.48550/arXiv.1504.01942.
- [4] P. Dendorfer, H. Rezatofighi, A. Milan, J. Shi, D. Cremers, I. Reid, S. Roth, K. Schindler, and L. Leal-Taix e, "Mot20: A benchmark for multi object tracking in crowded scenes," *arXiv: 2003.09003*, 2020, [Online], DOI: 10.48550/arXiv.2003.09003.
- [5] Y. Xiu, Y. Ban, G. Delorme, C. Gan, D. Rus, and X. Alameda-Pineda, "Transcenter: Trans-formers with dense queries for multiple-

- object tracking,” *arXiv:2103.15145*, 2021, DOI: 10.48550/arXiv.2103.15145.
- [6] D. Stadler and J. Beyerer, “Improving multiple pedestrian tracking by track management and occlusion handling,” *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, TN, USA, 2021, DOI: 10.1109/cvpr.46437.2021.01081.
- [7] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Uproft, “Simple online and realtime tracking,” *2016 IEEE International Conference on Image Processing (ICIP)*, Phoenix, AZ, USA, 2016, DOI: 10.1109/icip.2016.7533003.
- [8] S. Kapania, D. Saini, S. Goyal, N. Thakur, R. Jain, and P. Nagrath, “Multi object tracking with uavs using deep sort and yolov3 retinanet detection framework,” *1st ACM Workshop on Autonomous and Intelligent Mobile Systems*, 2020, DOI: 10.1145/3377283.3377284.
- [9] E. Bochinski, V. Eiselein, and T. Sikora, “High-Speed tracking-by-detection without using image information,” *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, Lecce, Italy, 2017, DOI: 10.1109/avss.2017.8078516.
- [10] R. Pereira, G. Carvalho, L. Garrote, and U. J. Nunes, “Sort and Deep-SORT Based Multi-Object Tracking for Mobile Robotics: Evaluation with New Data Association Metrics,” *Applied Sciences*, vol. 12, no. 3, 2022, DOI: 10.3390/app12031319.
- [11] T. Horiuchi, S. Thompson, S. Kagami, and Y. Ehara, “Pedestrian tracking from a mobile robot using a laser range finder,” *2007 IEEE International Conference on Systems, Man and Cybernetics*, Montreal, QC, Canada, 2007, DOI: 10.1109/icsmc.2007.4413964.
- [12] D. Reid, “An algorithm for tracking multiple targets,” *IEEE Transactions on Automatic Control*, vol. 24, no. 6, Dec., 1979, DOI: 10.1109/tac.1979.1102177.
- [13] N. A. Tsokas and K. J. Kyriakopoulos, “Multi-robot multiple hypothesis tracking for pedestrian tracking,” *Autonomous Robots*, vol. 32, no.1, 2012, DOI: 10.1007/s10514-011-9259-7.
- [14] B. Nam, S.-I. Kang, and H. Hong, “Pedestrian detection system based on stereo vision for mobile robot,” *2011 17th Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV)*, Ulsan, South Korea, 2011, DOI: 10.1109/fcv.2011.5739758.
- [15] F. Basso, M. Munaro, S. Michieletto, E. Pagello, and E. Menegatti, “Fast and Robust Multi-people Tracking from RGB-D Data for a Mobile Robot,” *Intelligent Autonomous Systems 12*, 2013, DOI: 10.1007/978-3-642-33926-4\_25.
- [16] H. Zhang, C. Reardon, and L. E. Parker, “Real-Time Multiple Human Perception With Color-Depth Cameras on a Mobile Robot,” *IEEE Transactions on Cybernetics*, vol. 43, no. 5, Oct., 2013, DOI: 10.1109/tcyb.2013.2275291.
- [17] Z. Sun, J. Chen, L. Chao, W. Ruan, and M. Mukherjee, “A Survey of Multiple Pedestrian Tracking Based on Tracking-by-Detection Framework,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 5, May, 2020, DOI: 10.1109/tcsvt.2020.3009717.
- [18] S. Hossain and D.-J. Lee, “Deep learning-based real-time multiple-object detection and tracking from aerial imagery via a flying robot with gpu-based embedded devices,” *Sensors*, vol. 19, no. 15, 2019, DOI: 10.3390/s19153371.
- [19] L. A. Nguyen, P. T. Dung, T. D. Khoa, N. H. Son, N. T. Hiep, P. V. Nguyen, V. D. Truong, D. H. Toan, N. M. Hung, T.-D. Ngo, and X.-T. Truong, “Deep learning-based multiple objects detection and tracking system for socially aware mobile robot navigation framework,” *2018 5th NAFOSTED Conference on Information and Computer Science (NICS)*, Ho Chi Minh City, Vietnam, 2018, DOI: 10.1109/nics.2018.8606878.
- [20] G. de S. Carvalho, “Kalman Filter-Based Object Tracking Techniques for Indoor Robotic Applications,” PhD thesis, University of Coimbra, Coimbra, Portugal, 2021, [Online], <https://studogeral.sib.uc.pt/handle/10316/98163>.
- [21] G. Koch, “Siamese neural networks for one-shot image recognition,” M.S thesis, University of Toronto, Toronto, Canada, 2015, [Online], <http://www.cs.toronto.edu/~gkoch/files/msc-thesis.pdf>.
- [22] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, “Visual object tracking using adaptive correlation filters,” *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, USA, 2010, DOI: 10.1109/cvpr.2010.5539960.
- [23] N. Wojke, A. Bewley, and D. Paulus, “Simple online and realtime tracking with a deep association metric,” *2017 IEEE international conference on image processing (ICIP)*, Beijing, China, 2017, DOI: 10.1109/icip.2017.8296962.
- [24] D. Scaramuzza and F. Fraundorfer, “Visual odometry [tutorial],” *IEEE Robotics & Automation Magazine*, vol. 18, no. 4, Dec., 2011, DOI: 10.1109/mra.2011.943233.
- [25] Z. Z. Nejad and A. H. Ahmadabadian, “ARM-VO: an efficient monocular visual odometry for ground vehicles on ARM CPUs,” *Machine Vision and Applications*, vol. 30, no. 6, 2019, DOI: 10.1007/s00138-019-01037-5.
- [26] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, “Performance Measures and a Data Set for Multi-target, Multi-camera Tracking,” *European Conference on Computer Vision*, 2016, DOI: 10.1007/978-3-319-48881-3\_2.
- [27] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, DOI: 10.1109/cvpr.2016.91.
- [28] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C Berg, “SSD: Single Shot MultiBox Detector,” *European Conference on Computer Vision*, 2016, DOI: 10.1007/978-3-319-46448-0\_2.
- [29] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” *arXiv:2004.10934*, 2020, DOI: 10.48550/arXiv.2004.10934.
- [30] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, “Scaled-YOLOv4: Scaling Cross Stage Partial Network,” *2021 IEEE/*



*CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, 2021, DOI: 10.1109/cvpr46437.2021.01283.

- [31] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018*, DOI: 10.1109/cvpr.2018.00474.
- [32] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "MOT16: A Benchmark for Multi-Object Tracking," *arXiv: 1603.00831*, 2016, DOI: 10.48550/arXiv.1603.00831.



### 박재훈

2020 한양대학교 ERICA 컴퓨터공학과 (학사)  
2022 한양대학교 융합로봇시스템학과 (석사)

관심분야: Robot Perception, Deep Learning, Computer Vision



### 안민성

2014 UCLA 기계공학과(학사)  
2018 UCLA 기계공학과(석사)  
2022 UCLA 기계공학과(박사 과정)

관심분야: Legged Robots, Optimization and Learning-based Control



### 한재권

2000 고려대학교 기계공학과(학사)  
2002 고려대학교 기계공학과(석사)  
2012 Mechanical Engineering at Virginia Tech(박사)  
2012~2015 (주)로보티즈 수석연구원  
2018~현재 한양대학교 공학대학 로봇공학과 부교수

관심분야: 휴머노이드 로봇, 기계설계