JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 27 No. 10, pp. 19-28, October 2022
https://doi.org/10.9708/jksci.2022.27.10.019

A Self-Supervised Detector Scheduler for Efficient Tracking-by-Detection Mechanism

Dae-Hyeon Park*, Seong-Ho Lee*, Seung-Hwan Bae**

*M. S. candidate, Vision & Learning Laboratory, Inha University, Incheon, Korea
*Full-time Researcher, Vision & Learning Laboratory, Inha University, Incheon, Korea
**Associate Professor, Dept. of Computer Engineering, Inha University, Incheon, Korea

[Abstract]

In this paper, we propose the Detector Scheduler which determines the best tracking-by-detection (TBD) mechanism to perform real-time high-accurate multi-object tracking (MOT). The Detector Scheduler determines whether to run a detector by measuring the dissimilarity of features between different frames. Furthermore, we propose a self-supervision method to learn the Detector Scheduler with tracking results since it is difficult to generate ground truth (GT) for learning the Detector Scheduler. Our proposed self-supervision method generates pseudo labels on whether to run a detector when the dissimilarity of the object cardinality or appearance between frames increases. To this end, we propose the Detector Scheduling Loss to learn the Detector Scheduler. As a result, our proposed method achieves real-time high-accurate multi-object tracking by boosting the overall tracking speed while keeping the tracking accuracy at most.

▸ **Key words**: Multi-Object Tracking, Tracking-by-Detection Scheduling, Dissimilarity Learning, Self-Supervised Learning, Quality Measure

[요 약]

본 논문에서는 실시간 고성능 다중 객체 추적을 수행하기 위해 최적의 TBD (Tracking-by-detection) 메커니즘을 결정할 수 있는 Detector Scheduler를 제안한다. Detector Scheduler는 서로 다른 프레임 간의 특징량 차이를 측정하는 것으로 검출기 실행 여부를 결정하여 전체 추적 속도를 향상한다. 하지만, Detector Scheduler의 학습에 필요한 GT (Ground Truth) 생성이 어렵기 때문에 Detector Scheduler를 추적 결과만을 통해 학습 가능한 자가 학습 방법을 제안한다. 제안된 자가 학습 방법은 프레임 간의 객체 카디널리티와 객체 외형 특징량의 비유사도가 커질 때 검출기를 실행할 수 있도록 의사 레이블을 생성하고 제안된 손실함수를 통해 Detector Scheduler를 학습한다.

▸ **주제어**: 다중 객체 추적, 검출 기반 추적 스케줄링, 비유사도 학습, 자가학습, 품질 측정

• First Author: Dae-Hyeon Park, Seong-Ho Lee, Corresponding Author: Seung-Hwan Bae
*Dae-Hyeon Park (saintPalite2221@inha.edu), Vision & Learning Laboratory, Inha University
*Seong-Ho Lee (leesh_vl@inha.ac.kr), Vision & Learning Laboratory, Inha University
**Seung-Hwan Bae (shbae@inha.ac.kr), Dept. of Computer Engineering, Inha University
• Received: 2022. 09. 08, Revised: 2022. 09. 22, Accepted: 2022. 09. 28.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

다중 객체 추적(Multi-Object Tracking)의 주요 패러다임인 검출 기반 추적(Tracking-by-detection)[1, 2, 3, 4]은 시간적 흐름에 따라 검출 객체 간의 지역/전역적 연관(association)을 수행해 트랙을 생성한다. 따라서 고성능 검출기를 사용할 경우 객체 위치에 의한 불확실성을 감소하게 하여 다중 객체 추적 정확도(Multi-Object Tracking Accuracy, 이하 MOTA)를 높일 수 있다. 하지만 검출 기반 추적에서 검출기를 실행하기 위한 연산 비용이 연관을 위한 연산 비용보다 훨씬 높기 때문에 전체 추적 속도가 하락하는 문제가 발생한다.

본 연구에서는 이 문제를 해결하기 위해 높은 연산 비용이 필요한 검출기를 매 프레임마다 실행할 필요 없이 추적을 진행할 수 있는 방법을 제시한다. 본 방법은 인접 프레임 간의 유사한 컨텍스트(context)를 가진 특정 프레임에서 객체 카디널리티(cardinality)와 객체 외형(appearance)의 변화가 적다는 점을 이용한다. 이를 통해 검출 없이 모션 모델로 추적을 수행할 수 있는 프레임을 결정하여, 검출 기반 연관 없이 객체를 추적할 수 있는 모션을 학습할 수 있다.

결과적으로 검출기 실행 없이 모션 모델로 추적을 수행함으로써 전체 추적 속도를 향상할 수 있다. 이를 위해 우리는 일정 간격(즉, 프레임 도메인 내 균일 샘플링)마다 검출기 실행 스킵(skip)을 적용할 수 있다. 그러나 이 간격은 동일한 시퀀스 내에서도 다를 수 있기 때문에 최적 간격을 찾는 것은 어려운 일이다. 예를 들어 객체의 모션(motion) 변화가 증가하거나 추적 품질이 낮아질 때 MOTA를 유지하기 위해서는 검출기를 더 빈번히 실행함으로써 간격을 더 좁혀야 한다. 따라서 우리는 실시간 다중 객체 추적을 수행하기 위해 감소를 최소화하며 전체 추적 속도를 향상할 수 있는 최적의 검출 시점(키 프레임)을 찾는 것을 목표로 한다.

이를 위해 일부 최근 연구[4, 5]는 이전 프레임과 현재 프레임 간의 추적 결과의 바운딩 박스(bounding box) 쌍(pair)을 서로 비교하고, 바운딩 박스 간의 일치 정도가 낮을 경우 검출기를 실행한다. 그러나 트랙 초기화 및 종료는 검출 기반 연관 없이 수행될 수 없으므로 추적 품질 뿐만 아니라 객체 카디널리티 또한 고려되어야 한다. 따라서 추적 객체의 바운딩 박스 또는 외형 특징량만을 비교[5, 6, 7]하는 것만으로는 두 변동 사항을 모두 측정할 수 없다. 이를 해결하기 위해 우리는 현재 프레임과 키 프레임 간의 컨볼루션 특징량의 차이를 측정하는 방법을 제안한다. 또한, 온라인 다중 객체 추적이 실행되는 동안, MOT에 대한 최적의 검출 기반 추적 메커니즘을 결정할 수 있는 Detector Scheduler를 제안한다. 그러나 Detector Scheduler에 대한 GT가 없기 때문에 지도 학습(Supervised learning)을 사용할 수 없다. 또한, 이에 대한 GT를 만드는 것은 추적기와 검출기의 성능에 의존하기 때문에 무의미할 수 있다. 따라서 본 연구에서는 자가 학습을 통해 Detector Scheduler를 학습한다. 또한, 본 연구에서는 검출기 실행 여부를 나타내는 의사 레이블(pseudo label)을 제안한다. 검출기 실행 유무에 따른 다중 객체 추적 결과 기반 객체 카디널리티(cardinality) 및 객체 위치정보(localization) 차이를 통해 새로운 Detector Scheduling 손실 함수를 제안한다. 본 손실 함수를 최소화함으로써 MOTA를 최대한 유지하면서 MOT 전체 속도를 높이는 최적의 검출 기반 추적 메커니즘을 결정할 수 있다. 이 메커니즘은 검출 기반 추적 방법의 정확성에 따라 적응적으로 결정될 수 있다. 결과적으로 본 연구의 주요 기여는 다음과 같다. 첫째로 실시간 및 고성능 추적을 위한 최적의 검출 기반 추적 메커니즘을 결정할 수 있는 Detector Scheduler 제안, 둘째로 자가 학습 방법을 통해 Detector Scheduler 학습을 위한 의사 레이블 및 Detector Scheduling 손실 함수를 제안한다. Baseline과 비교해 MOTA를 5.8% 줄이는 것만으로 속도를 47.0% 향상시켰다. 또한, 우리는 광범위한 절제 실험과 MOT 벤치마크 데이터 셋에서 최신 MOT 모델 간의 성능 비교를 제공한다. 우리의 Detector Scheduler는 단일 NVIDIA Titan Xp를 사용하여 MOT16 테스트 셋에서 MOTA 74.5%, 20.6Hz 속도를 달성한다.

본 연구은 다음과 같이 구성된다. 2장에서는 본 연구와 연관된 기존 연구에 대해 기술하고, 3장에서는 본 연구의 핵심인 Detector Scheduler에 기술한다. 4장에서는 본 연구에서 제안하는 Detector Scheduler를 결합한 모델을 MOT 벤치마크 챌린지 데이터 셋에서 절제 실험을 수행하고, 최신 다중 객체 추적기와 비교 평가를 수행한다. 최종적으로 5장에서는 본 연구의 결론을 기술한다.

II. Related Works

이 장에서는 본 연구와 연관된 기존 연구에 대해 기술한다.

1. Tracking-by-detection

검출 기반 추적(Tracking-by-detection)은 먼저 검출기를 사용하여 주어진 이미지에서 객체 위치를 결정한 다음, 연속 프레임 간에 검출 결과를 연결하여 식별 가능한 ID를 갖는 트랙을 생성하는 추적 방법이다. 검출 기반 추적 방법은 SDE (Seperated Detection and Embedding) [2, 8]와 JDE (Joint Detection and Embedding) [1, 3, 4]로 구분된다. 두 방법의 주요 차이점은 검출 작업과 연관 작업이 동일 네트워크에서 처리하는지에 대한 여부로 설명할 수 있다. SDE는 검출 및 연관 작업이 독립적으로 학습된 서로 다른 네트워크를 사용한다. 한편, JDE는 검출 및 연관 헤더를 백본 네트워크에 연결하여 동시에 학습한다. 따라서 JDE는 저수준(low-level)의 특징량을 공유하여 추적 속도를 향상한다. 그러나 두 방법 모두 전체 프레임에 대해 검출기를 적용하고 있어 검출 단계는 여전히 전반적인 MOT 연산 비용에 크게 영향을 미친다.

2. Key-Frame Scheduling

MOT 전체 속도를 향상시키기 위해 검출기 실행을 스킵하는 방법이 고려될 수 있다. 이때 고정 프레임 간격을 갖는 검출기 실행 스킵 방법[9, 10]은 그 단순성 때문에 일반적으로 사용된다. 하지만 객체 카디널리티 오류 및 객체 위치정보 오류로 인한 추적 객체 손실을 고려하지 않고 검출을 결정하는 키 프레임을 선택하기 때문에 MOTA를 저하시킬 가능성이 있다. 이를 해결하기 위해 일부 적응형 샘플링 방법[5, 7, 11]은 object region과 region feature를 비교하여 키 프레임을 선택한다. 예를 들어, [7]은 객체 크기 및 연속 프레임 간의 모션 차이를 고려하여 키 프레임을 결정한다. [6, 11]은 이 문제를 강화 학습 문제로 규정하고, 객체 위치정보 비유사도을 바탕으로 키 프레임을 결정한다. 하지만 이런 방법들은 프레임 간의 객체 카디널리티 변동을 고려하지 않는다. 본 연구에서는 검출 기반 추적 스케줄링을 위해 프레임 간의 비유사성을 측정하여 객체 위치정보 및 카디널리티를 본 연구에서 제안하는 손실 함수에 반영할 수 있는 방법을 제시한다.

3. Self-Supervised Learning

자가 학습[12, 13]은 레이블이 없는 데이터로 모델을 학습하기 위한 의사 레이블을 생성한다. 자가 학습을 사용하여 MOT 모델의 일반화 능력을 향상하기 위한 몇 가지 연구가 있다. [4]는 사람 검출 데이터 셋을 사용하여 재식별(Re-ID)에 대한 일반화 방법을 개선한다. [14]는 spatial-correlation learning의 constraint [15]를 적용하기 위해 자가 학습 손실 함수를 활용한다. [16]은 자가 학습을 사용하지 않고 Re-ID 모델을 학습시키기 위해 SORT [2]를 사용하여 의사 레이블(pseudo label)을 생성한다. 본 연구에서는 마찬가지로 MOT에 대한 자가 학습을 활용한다.

III. Methodology

Fig. 1에서는 본 연구에서 제시하는 MOT에 대한 Detector Scheduler의 Detector Scheduling Network에 대해 기술한다. 이 장에서는 Detector Scheduler의 각 구성 요소들(Detector Scheduling Network, 의사 레이블 및 Detector 손실함수, 자가학습)에 대해 자세히 설명한다.

1. Detector Scheduling Network

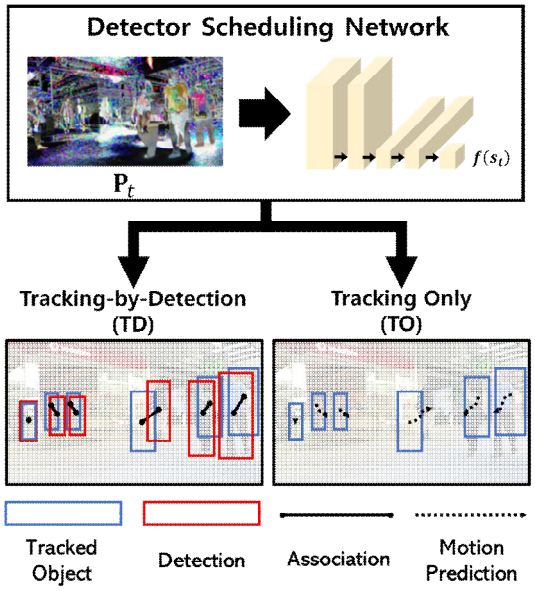

Fig. 1. Detector Scheduling Network

1장에서 논의한 바와 같이 Detector Scheduler의 학습 목표는 다중 객체 추적에서 검출기 실행 여부를 결정하는 것이다. 이를 위해 현재 t번째 프레임의 특징량 $\mathbf{F}$과 키 프레임의 특징량 $\mathbf{F}$을 뺀 특징량 $\mathbf{P}_t \in \mathbb{R}^{C \times H \times W}$을 통해, 두 특징량 간의 차이를 계산한다. 이 때 $\mathbf{P}_t$는 다음과 같다.

$$\mathbf{P}_t = \mathbf{F}_t - \mathbf{F}_{k_e} \quad (1)$$

Detector Scheduling Network는 $\mathbf{P}_t$를 입력으로 하여 detector scheduling probability $f(s_t)$를 출력한다. 이 때, $f(s_t)$가 낮은 경우, 프레임에서 검출기를 실행하도록 유도한다. 구체적으로 $\mathbf{P}_t$를 3×3 ConvBlocks에 입력한 다음, 1×1 ConvBlocks에 차례대로 입력한다. 여기서 각 Blocks은 컨볼루션, 배치 정규화 및 Leaky ReLU 활성화 함수($a=0.01$) 레이어들을 포함한다. 3×3 및 컨볼루션 레이어의 출력 채널 크기는 각각 C 및 1이다. 연속 컨볼루션 연산 후 특징량을 $\mathrm{H}\times W$ 차원을 갖는 특징 벡터로 평탄화(Flatten)한 다음 64, 64, 1개의 뉴런으로 3개의 Fully Connected Layers를 적용하여 s_t를 출력한다. 마지막으로, 시그모이드 함수 $f(\cdot)$로 $f(s_t)$를 계산할 수 있다.

2. Detector Scheduler Training via Self-Supervision

Detector Scheduler는 다음과 같은 근거에 기반하여 자가 학습 방법으로 학습된다. 첫째로, MOT에 대한 검출 기반 추적 메커니즘 학습을 위한 사용 가능한 공개된 GT 또는 정의된 가이드라인이 없다. 둘째로, 추적기와 검출기 간의 성능 의존성이 강하기 때문에 GT를 생성하는 것이 어렵다. 정확도와 속도는 trade-off 관계이기 때문에 특정 성능에 맞는 GT를 생성하는 것은 까다로운 일이다. 따라서 본 연구에서는 키 프레임과 현재 프레임의 추적 결과 간의 비유사성을 측정하여 검출기 실행 여부를 나타낼 수 있는 의사 레이블 생성 방법을 제안한다. 구체적으로 본 연구에서는 두 프레임 간의 객체 카디널리티 및 객체 위치 정보 오류를 측정한다. 결과적으로 본 연구에서 제안하는 자가 학습 방법은 추적 결과에 따라 적응적으로 의사 레이블을 생성할 수 있다.

3. Quality Measure of Tracking without Detection

N 프레임의 시퀀스가 주어질 때, 프레임 I_t에서 검출기 $\mathbf{D}$를 실행함으로써 검출 결과 $\mathbf{D}(I_t)=D_t=\left\{\mathbf{d}_t^i,\mathrm{y}_t^i\right\}_{i=1}^{|D_t|}$를 생성한다. 여기서 $\mathbf{d}$는 객체 i에 대한 바운딩 박스이고, $|D_t|$는 t번째 프레임에서 탐지된 객체 수다. 본 연구에서는 현재 프레임인 t번째 프레임에서 $T_t=\left\{\hat{\mathbf{d}}_t^j,\hat{y}_t^j\right\}_{j=1}^{|T_t|}$를 평가하기 위해 트랙 $T_t^{(TD)}=\mathbf{T}\{T_{t-1},D_t\}$를 위한 온라인 검출 기반 추적 문제를 공식화한다. 이 때 $\mathbf{T}$는 추적기, $\hat{\mathbf{d}}_t^j$와 $\hat{y}_t^j$는 각각 트랙 j에 대한 추적 결과와 추적 식별 레이블(track identity label)이다. $|T_t|$는 t번째 프레임에서의 트랙들의 개수이다.

본 연구에서는 객체 모션을 예측하기 위해 칼만 필터[17]를 사용한다. t번째 프레임에서 검출기 실행을 스킵하면 추적 문제를 $T_t^{(TO)}=\mathbf{T}(T_{t-1})$로 공식화할 수 있다. 일반적으로, 이 문제는 프레임 $t-1$까지의 추적 결과 T_{t-1}를 기반으로 한 예측값을 통해 해결될 수 있다. 대부분의 경우 검출을 수행하여 추적하는 $T_t^{(TD)}$의 추적 품질이 $T_t^{(TO)}$보다 높다. 하지만 $T_t^{(TD)}$에 대한 추적 연산 비용은 $T_t^{(TO)}$에 비해 높기 때문에 MOT 정확도를 최소한으로 줄이면서 $T_t^{(TO)}$의 수를 늘리는 것을 목표로 한다. 전체 프레임에서 최적의 $T_t^{(TO)}$를 찾기 위해 $T_t^{(TO)}\approx T_t^{(TD)}$일 경우, t번째 프레임에서 검출 없이 추적이 가능하다는 가정을 세운다. 따라서 객체 카디널리티와 객체 위치정보 측면에서 $T_t^{(TO)}$의 추적 결과가 $T_t^{(TD)}$의 추적 결과와 유사하다면 본 가정은 성립될 수 있다. 따라서 최적의 $T_t^{(TO)}$를 찾는 문제는 전체 프레임에서 $T_t^{(TO)}\approx T_t^{(TD)}$를 찾는 문제로 변환할 수 있다. 이를 위해 $T_t^{(TO)}\approx T_t^{(TD)}$를 추정하기 위해 두 방법 간의 객체 카디널리티와 객체 위치정보 유사성을 정의한다.

3.1 Cardinality Measure

객체 카디널리티 유사성 측정 방법은 1장에서 논의한 바와 같이, $T_t^{(TD)}$와 $T_t^{(TO)}$ 간의 객체 카디널리티가 추적 품질의 척도로 고려되어야 한다. 따라서, 우리는 $T_t^{(TD)}$와 $T_t^{(TO)}$ 사이의 객체 카디널리티를 비교하고, 객체 카디널리티 유사성 척도 S_{card}를 다음과 같이 정의할 수 있다

$$S_{card}(T_t^{(TO)},T_t^{(TD)}) = \min\left(-\frac{1}{e}\cdot\ln\left(1-R(T_t^{(TO)},T_t^{(TD)})\right),1\right) \quad (2)$$

$R(T_t^{(TO)}T_t^{(TD)})$는 $T_t^{(TD)}$와 $T_t^{(TO)}$사이의 객체 카디널리티의 비율이다. 이를 [0, 1] 사이의 값으로 정규화하기 위해, 우리는 다음과 같이 계산할 수 있다.

$$R(T_t^{(TO)}T_t^{(TD)}) = \min\left(\frac{|T_t^{(TD)}|}{|T_t^{(TO)}|},\frac{|T_t^{(TO)}|}{|T_t^{(TD)}|}\right) \quad (3)$$

위 식 2에서 $S_{card}=1$인 경우, 두 추적 방법이 같은 개수의 트랙을 생성했음을 나타낸다. 즉, t번째 프레임에서 트랙 초기화 및 종료가 일어나지 않음을 의미한다. 따라서 S_{card}가 높을수록 $T_t^{(TO)}$의 추적 품질이 높으며, 아니라면 추적 품질이 낮다는 것을 의미한다.

3.2 Localization Measure

객체 위치정보 유사성 측정 방법은 $T_t^{(TD)}$와 $T_t^{(TO)}$사이의 바운딩 박스 $\hat{\mathbf{d}}_t$ 사이의 비유사성을 평가하기 위해, 본 연구에서는 객체 위치정보 유사성 척도 S_{loc}를 정의한다. 두 방법 간의 바운딩 박스 쌍이 주어지면, IoU(Interest of Union)를 사용하여 두 바운딩 박스 쌍 간의 유사성을 평가할 수 있다. 따라서, 우리는 S_{loc}을 다음과 같이 정의할 수 있다.

$$S_{loc}(T_t^{(TO)}, T_t^{(TD)}) = \frac{1}{|T_t^*|}\sum_{(i_k j_k)_{k=1}}^{|T_t^*|} IOU(\hat{\mathbf{d}}_t^{(TD),i_k}, \hat{\mathbf{d}}_t^{(TO),j_k}) \quad (4)$$

여기서 $|T_t^*|$ 는 $T_t^{(TD)}$와 $T_t^{(TO)}$간에 매칭된 트랙 쌍에 대한 카디널리티다. 그리고 $(i_k j_k)$ 간의 관련성을 결정하기 위해 우리는 $T_t^{(TD)}$와 $T_t^{(TO)}$의 바운딩 박스로 구성된 bigraph를 정의한다. 그리고 $T_t^{(TD)}$에 대한 각 노드는 $T_t^{(TO)}$에 대한 모든 노드와 연결된다. 이때 엣지 가중치(edge weight)는 연결된 노드 간의 IoU 점수를 사용하여 평가한다. 이 그래프의 최대 가중치 일치 쌍은 헝가리안 알고리즘[18]에 의해 최적으로 결정될 수 있다. 그런 다음 일치된 쌍의 평균 IoU 점수를 계산하여 S_{loc}을 평가한다.

4. Pseudo Labeling and Detector Scheduling Loss

Detector Scheduler를 학습하기 위해, 우리는 의사 GT 레이블 $G=\{G_t\}^{N_t=1}$을 생성한다. 학습 데이터 N개 프레임의 시퀀스가 주어지면 온라인 추적기 T와 검출기 D를 실행하여 각 프레임에서 의사 레이블(pseudo label)을 생성한다. 이를 통해 매 프레임마다 $T_t^{(TD)}$와 $T_t^{(TO)}$를 생성할 수 있다. t번째 프레임의 각 레이블 G_t는 이진 레이블이며, 검출 없는 추적 방법(TO=1) 또는 검출 기반 추적 방법(TD=0)을 나타낸다. 그리고 $T_t^{(TD)}$와 $T_t^{(TO)}$를 비교하여 S_{card}와 S_{loc}의 유사성을 평가한다. 따라서 G_t를 다음과 같이 정의할 수 있다.

$$G_t = \begin{cases} 1, & s.t.\ S_{card} \cdot S_{loc} \geqq \theta_{ps} \\ 0, & s.t.\ S_{card} \cdot S_{loc} < \theta_{ps} \end{cases} \quad (5)$$

이때 θ_{ps}는 의사 레이블 임계값(threshold)이다. 이때, $S_{card} \cdot S_{loc} \geqq \theta_{ps}$ 일 경우 추적 방법 $T_t^{(TD)}$와 $T_t^{(TO)}$의 추적 품질은 비슷하다. 따라서 해당 프레임에서는 검출 없는 추적 방법 $T_t^{(TO)}$를 유도한다. 반대의 경우, 객체 위치정보 혹은 객체 카디널리티 오류로 인해 추적 품질이 저하된 경우이므로, 검출 기반 추적 방법 $T_t^{(TD)}$을 유도한다. 따라서 $G_t=1$일 때는 t번째 프레임에서 추적 속도를 향상할 수 있으며, $G_t=0$일 때는 검출 기반 연관을 통해 MOTA를 향상시킬 수 있다. 본 연구에서는 G_t를 통해 Detector Scheduler를 학습하기 위한 손실함수 $L_{decision}$을 제안한다.

$$L_{decision} = -G_t \log(f(s_t)) - (1-G_t)\log(1-f(s_t)) \quad (6)$$

5. Tracking by Detector Scheduler

Detector Scheduler를 구현하기 위해 MOT 데이터 셋[19]에서 최신(state-of-the-art) 모델의 정확도와 속도를 나타내는 FairMOT [4]를 baseline으로 사용한다. 따라서 본 연구에서는 그림 1에 제안된 네트워크와 baseline을 결합한다. 각 프레임의 기존 트랙과 연관되지 않은 검출 결과가 주어질 때, IoU 연관 방법을 통해 $t-1$번째 프레임의 트랙과 t번째 프레임의 검출 결과를 연관하여 트랙 초기화를 수행한다.

그리고 $f(s_t)$을 추론하여 각 프레임의 추적 동작을 결정한다. 만약 $f(s_t) \leqq \theta_{\text{det}}$라면 검출 기반 추적(TD) 방법을 수행한다. 따라서 검출기를 사용해 검출 결과를 도출한 뒤 기존 트랙과 연관시킨다. 반대로 $f(s_t) > \theta_{\text{det}}$일 때, 검출 없는 추적(TO) 방법을 수행하며, 칼만 필터로 모션을 예측해 트랙을 도출한다. 또한, 트랙 종료를 위해 트랙 신뢰도(track confidence)를 정의하고 연관된 검출 결과의 분류 점수(classification score)로 초기화한다. 그리고 검출 없는 추적이 발생할 때, 스케일링 계수(본 실험에서는 0.9로 설정)로 트랙 신뢰도를 감소시킨다. False Positive를 줄이기 위해 추적 신뢰도가 0.5 미만일 때 트랙을 종료한다. 전체 추적 속도를 더욱 높이기 위해, 다음과 같이 $f(s_t) \geqq \theta_{\text{det}}$일 때, Detector Scheduler를 사용하지 않고 $f(s_t)$를 $t+1$번째 프레임으로 전파하여 $f(s_{t+1})$를 계산한다.

Table 1. Detailed comparison with the baseline on MOT15 dataset.

	DR	MOTA ↑	FP ↓	FN ↓	Hz ↑
Baseline	100%	72.9%	1,409	3,204	19.3
	80.0%	57.4%	1,068	6,210	23.5
	75.0%	53.5%	930	7,005	24.9
	66.6%	46.9%	801	8,253	27.9
Detector Scheduler (Ours)	100%	72.9%	1,409	3,204	18.9
	80.3%	69.2% [3.7% ↓]	1,881	3,307	23.1 [22.2% ↑]
	74.1%	68.1% [4.8% ↓]	1,924	3,523	25.1 [32.8% ↑]
	66.3%	67.1% [5.8% ↓]	1,852	3,753	27.8 [47.0% ↑]

$$f(s_{t+1}) = \begin{cases} \xi \cdot f(s_t), \ s.t. \ f(s_t) \geqq \theta_{\text{det}} \\ DM(F_{k_e}, F_t), \ s.t. \ f(s_t) < \theta_{\text{det}} \end{cases} \quad (7)$$

여기서 ξ는 0.83으로 설정한다. $f(s_t) < \theta_{\text{det}}$일 때 Detector Scheduler를 실행하고, $f(s_t) \geqq \theta_{\text{det}}$일 때 객체의 모션 모델만으로 객체를 추적한다. 이 예측 기반 추적은 $T_t^{(TO)} \approx T_t^{(TD)}$ 일 때 실행할 수 있으며, $f(s_t) \geqq \theta_{\text{det}}$일 때 $T_t^{(TO)} \approx T_t^{(TD)}$라고 고려한다.

IV. Experiments

이 장에서는 Detector Scheduler의 효과를 증명하기 위해 SOTA 모델 간의 비교와 광범위한 절제 실험을 수행한다.

1. Datasets

실험에 사용할 데이터 셋은 baseline이 MOTChallenge 데이터 셋으로 학습하였기 때문에 본 모델 또한 이 데이터 셋으로 학습한다. Detector Scheduler를 학습하기 위해 MOT17 학습 데이터 셋을 사용한다. 이 학습 데이터 셋은 14-30Hz 프레임의 동적/정적 카메라에서 캡처된 7개의 시퀀스로 구성된다. SOTA 모델 간의 비교 평가를 위해 MOT16, MOT17 및 MOT20 테스트 데이터 셋에서 Detector Scheduler를 실행하고 챌린지 서버에서 성능을 평가한다. 절제 실험(ablation study)의 경우 MOT15 학습 데이터 셋을 검증 데이터 셋으로 활용한다. 단, 학습에 사용되었던 중복된 시퀀스는 사용하지 않는다. 이때 중복되어 절제 실험에 사용되지 않는 시퀀스는 Venice-2, ADL-Rundle-8, ADLRundle-6, ETH-Pedcross2이다.

2. Evaluation Metric

평가 메트릭(Evaluation Metric)은 MOTChallenge [19]에서 일반적으로 사용하는 메트릭[20]을 사용한다. Multi Object Tracking Accuracy (MOTA↑), ID F1 score (IDF1↑), False Positive (FP↓), False Negative (FN↓), ID Switches (IDs↓), 다중 객체 추적 속도(Hz↑). 여기서 ↑와 ↓는 각각 더 높거나 낮을수록 더 우수한 MOT 결과임을 나타낸다. 또한, 검출기 실행 수를 총 프레임 수로 나누어 검출기 실행 비율(Detection Run, 이하 DR)을 평가한다. 1장에서 설명한 바와 같이, 검출기 실행 비율은 검출 기반 추적 방법의 정확도과 속도에 가장 큰 영향을 미친다. 일반적으로 DR이 증가할수록 정확도는 높아지지만 속도는 느려진다.

3. Implementation details

앞서 언급한 바와 같이, 우리는 FairMOT [4]를 baseline으로 사용한다. 앞서 제안한 Detector Scheduling Network를 baseline에 결합하여 Detector Scheduler를 구현한다. 절제 실험의 경우, 먼저 동일한 학습 설정으로 MOT17 학습 데이터 셋에서 baseline을 학습한다. 그 후, 학습된 baseline을 프리징(freezing)하고, 식 6의 $L_{decision}$를 최소화하는 것으로 Detector Scheduler를 학습한다. 또한, MOTChallenge 테스트 데이터 셋에서 평가하기 위해, MIX 데이터 셋[4]에서 학습된 baseline를 프리징하고, 스크래치로 우리의 모델을 학습한다. 3장 4절에서 의사 GT를 생성하기 위해, 사전 학습된 baseline을 사용하여 $T_t^{(TD)}$와 $T_t^{(TO)}$를 생성한다. 우리는 추적 품질 측정을 사용하여 두 방법 간의 유사성 $S_{card} \cdot S_{loc}$ 을 계산하고, 식 5를 사용하여 G_t를 생성한다. 또한 $\beta_1 = 0.9$, $\beta_2 = 0.999$인 Adam 옵티마이저를 사용한다. Detector Scheduler는 서로 다른 시퀀스의 6개의 이미지를 포함한 미니 배치로 30 epoch 동안 추적기를 학습한다. 우리는 학습률(Learning Rate)을 5e-5로 설정하고, 20개의 epoch에서 0.1의 비율로 decay한다. Detector Scheduler는 i7-8700K CPU와 단일 Titan Xp로 구성된 PC에서 Detector Scheduler를 실행한다.

Table 2. Comparison with the SOTA trackers on MOTChallenge 16/17 test sets

	Tracker	MOTA ↑	IDF1 ↑	FP ↓	FN ↓	IDs ↓	Hz ↑
MOT16	QDTrack [21]	69.8%	67.1%	9,861	44,050	1,097	20.3
	CTracker [22]	67.6%	57.2%	8,934	48,305	1,897	6.8
	LM_CNN [23]	67.4%	61.2%	10,109	48,435	981	1.7
	HTA [24]	62.4%	64.2%	19,071	47,839	1,619	19.7
	TraDeS [25]	70.1%	64.7%	8,091	45,210	1,144	22.3
	Tube_TK [26]	64.0%	59.4%	10,962	53,626	1,117	1.0
	GSDT [27]	74.5%	68.1%	8,913	36,428	1,229	1.6
	CenterTrack [28]	69.6%	60.7%	10,458	42,805	2,124	17.5
	SOTMOT [29]	72.1%	72.3%	14,344	34,784	1,681	16.0
	Ours	74.5%	72.3%	8,614	36,856	1,016	20.6
MOT17	QDTrack [21]	68.7%	66.3%	26.589	146,643	3,378	20.3
	CTracker [22]	66.6%	49.0%	22,284	160,491	5,529	6.8
	TraDeS [25]	69.1%	63.9%	20,892	150,060	3,555	17.5
	Tube_TK [26]	63.0%	58.6%	27,060	177,483	4,137	3.0
	GSDT [27]	73.2%	66.5%	26,397	120,666	3,891	4.9
	CenterTrack [28]	67.8%	64.7%	18,498	160,332	3,039	17.5
	PermaTrack [30]	73.8%	68.9%	28,998	115,104	3,699	11.9
	Ours	73.0%	71.3%	23,622	125,664	3,114	20.6

4. Ablation study

Detector Scheduler의 영향: MOTA와 추적 속도 간의 trade-off를 조사하기 위해, 일정 간격으로 검출기 수행 스킵을 적용한다. 따라서 각 추적기는 일정 간격으로 검출기 실행을 스킵한다. #4TD-#1TO (DR=80%), #3TD-#1TO (DR=75%), #2TD-#1TO (DR=66.6%), 서로 다른 3가지의 간격을 적용한다. 여기서 #TD와 #TO는 각각 검출 기반 추적 방법과 검출 없는 추적 방법을 적용하는 프레임의 수다. Detector Scheduler의 경우, θ_{det}를 변경하는 것으로 3가지 검출기 실행 스킵 체계와 유사하도록 Detector Scheduler의 DR을 조정한다. Table 1은 Detector Scheduler와 baseline 간의 자세한 비교 결과를 나타낸다. 그리고 DR=100%일 때(즉, 모든 프레임에서 검출기를 실행) 두 추적기가 동일한 MOTA를 달성한다는 것을 확인할 수 있다. 하지만 Detector Scheduler와 baseline 간의 정확도 차이는 DR이 감소할수록 더 커진다. 구체적으로 DR=100%의 baseline과 비교하여 DR=66%의 Detector Scheduler는 MOTA를 5.8%를 줄이는 것으로 속도를 47.0% 향상할 수 있다. 그러나 DR=66%의 baseline은 DR=100%를 사용할 때와 비교하여 MOTA를 26%까지 저하시킨다. 유사한 DR을 가진 다른 추적기와 비교하여, 우리의 Detector Scheduler는 더 나은 MOTA와 속도를 보여준다. 이러한 결과로부터, 우리의 방법이 전체 추적 속도를 높이는 동시에 MOTA 감소를 최소화하는 데 매우 효과적이라는 것을 보여준다.

5. Comparison with state-of-the-arts methods

최신 추적기 모델과 비교 평가를 위해 MOT16 및 MOT17 테스트 셋에서 Detector Scheduler를 평가한다. 비교를 위해, 우리는 Private Detector를 사용하는 온라인 추적 방법과 비교한다. 우리는 이때 θ_{det}을 0.5로 설정했다. Table 2에 나타난 바와 같이, 우리의 Detector Scheduler는 다른 모델보다 더 높은 MOTA를 보여주면서 놀라운 속도를 달성한다. 이 비교를 통해, 우리의 방법이 추적 정확도를 유지하면서 전체 추적 속도 향상에 실제로 효과적이라는 것을 입증한다.

V. Conclusions

본 논문에서는 실시간으로 작동 가능한 고성능 다중 객체 추적을 위해 최적의 검출 기반 추적 메커니즘을 결정할 수 있는 Detector Scheduler를 제안한다. 제안된 방법에서는 의사 레이블과 Detector Scheduling 손실 함수 기반으로 Detector Scheduler를 학습한다. 결과적으로, 검출 유무에 따른 추적 차이를 줄임으로써 자가 학습 방법을 통해 Detector Scheduler를 학습시킬 수 있다. 또한, 광범위한 절제 실험과 SOTA 모델 간의 비교를 통해, 본 방법이 MOTA의 손실을 최소화하면서 전체 추적 속도를 높이는 데 유리하다는 것을 검증한다. 본 스케줄링 방법은 추후 실시간 작업이 필요한 산업 로봇 및 자율 주행 차량 등에 관한 분야로 확장할 수 있으리라 기대한다. 우리는 본 방법이 향후 실시간 MOT의 중요한 가이드라인이 될 수 있으리라 믿는다.

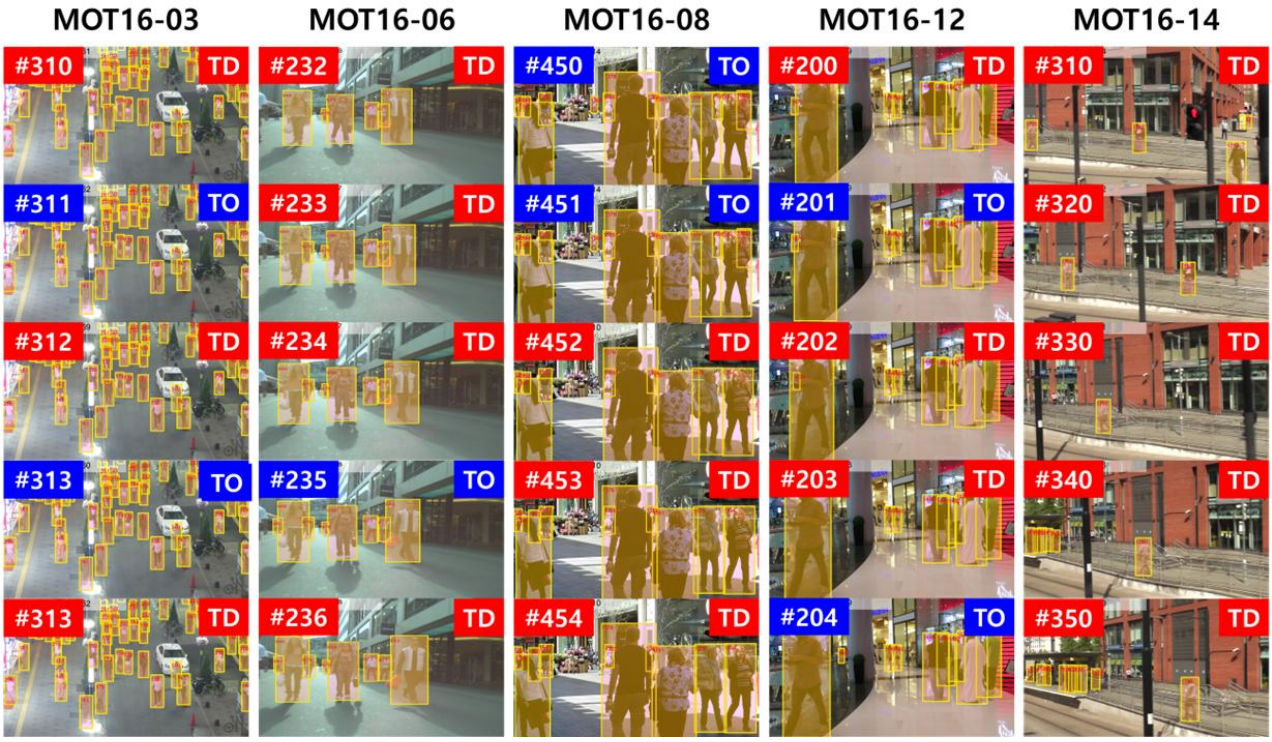

Fig. 2. Qualitative Results on MOT16 datasets

ACKNOWLEDGEMENT

This work was supported by the INHA UNIVERSITY Research Grant and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2022R1C1C1009208) and funded by the Ministry of Education (No.2022R1A6A1A03051705), and funded by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2022-0-00448: Deep Total Recall, 10%, No.RS-2022-00155915, No.2020-0-01389: Artificial Intelligence Convergence Research Center (Inha University)).

REFERENCES

[1] Wang, Zhongdao, et al., "Towards real-time multi-object tracking.", European Conference on Computer Vision, pp. 767-770, Aug. 2020, DOI: https://doi.org/10.1007/978-3-030-58621-8_7

[2] Wojke, Nicolai, Alex Bewley, and Dietrich Paulus, "Simple online and realtime tracking with a deep association metric.", 2017 IEEE international conference on image processing (ICIP), pp. 107-122, Sep. 2020, DOI: https://doi.org/10.1109/icip.2017.8296962

[3] Lu, Zhichao, et al., "Retinatrack: Online single stage joint detection and tracking.", Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 14668-14678, Jun. 2020, DOI: https://doi.org/10.1109/CVPR42600.2020.01468

[4] Zhang, Yifu, et al., "Fairmot: On the fairness of detection and re-identification in multiple object tracking.", International Journal of Computer Vision 129.11, pp. 3069-3087, Nov. 2021, DOI: https://doi.org/10.1007/s11263-021-01513-4

[5] Zhu, Xizhou, et al., "Towards high performance video object detection.", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7210-7218, Jun. 2018, DOI: https://doi.org/10.1109/cvpr.2018.00753

[6] Luo, Hao, et al., "Detect or track: Towards cost-effective video object detection/tracking.", Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 33. No. 01, pp. 8803-8810, Jan. 2019, DOI: https://doi.org/10.1609/aaai.v33i01.33018803

[7] Chen, Kai, et al., "Optimizing video object detection via a scale-time lattice.", Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7814-7823, Jun. 2018, DOI: https://doi.org/10.1109/CVPR.2018.00815

[8] Yu, Fengwei, et al., "Poi: Multiple object tracking with high performance detection and appearance feature.", European Conference on Computer Vision, pp. 36-42, Nov. 2016, DOI: https://doi.org/10.1007/978-3-319-48881-3_3

[9] Feichtenhofer, Christoph, Axel Pinz, and Andrew Zisserman,

"Detect to track and track to detect.", Proceedings of the IEEE international conference on computer vision, pp. 3038-3046, Oct. 2017, DOI: https://doi.org/10.1109/ICCV.2017.330

[10] Zhu, Xizhou, et al., "Deep feature flow for video recognition.", Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2349-2358, Jul. 2017, DOI: https://doi.org/10.1109/CVPR.2017.441

[11] Yao, Chun-Han, et al., "Video object detection via object-level temporal aggregation.", European conference on computer vision, pp. 160-177, Nov. 2020, DOI: https://doi.org/10.1007/978-3-030-58568-6_10

[12] Chen, Ting, et al., "Self-supervised gans via auxiliary rotation loss.", Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 12154-12163, Jun. 2019, DOI: 10.1109/CVPR.2019.01243

[13] Noroozi, Mehdi, et al., "Boosting self-supervised learning via knowledge transfer.", Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 9359-9367, Jun. 2018, DOI: https://doi.org/10.1109/CVPR.2018.00975

[14] Wang, Qiang, et al., "Multiple object tracking with correlation learning.", Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3876-3886, Jun. 2021, DOI: https://doi.org/10.1109/CVPR46437.2021.00387

[15] Dosovitskiy, Alexey, et al., "Flownet: Learning optical flow with convolutional networks.", Proceedings of the IEEE international conference on computer vision, pp. 2758-2766, Dec. 2015, DOI: https://doi.org/10.1109/ICCV.2015.316

[16] Karthik, Shyamgopal, Ameya Prabhu, and Vineet Gandhi, "Simple unsupervised multi-object tracking.", arXiv preprint arXiv:2006.02609, 2020, DOI: https://doi.org/10.48550/arXiv.2006.02609

[17] Kalman, Rudolph Emil, "A new approach to linear filtering and prediction problems.", Journal of Fluids Engineering Vol 82, pp. 35-45, Mar. 1960, DOI: https://doi.org/10.1115/1.3662552

[18] Kuhn, Harold W, "The Hungarian method for the assignment problem.", Naval research logistics quarterly 2.1‐2, pp. 83-97, Mar. 1955, DOI: https://doi.org/10.1002/nav.3800020109

[19] Milan, Anton, et al., "MOT16: A benchmark for multi-object tracking.", arXiv preprint arXiv:1603.00831, 2016, DOI: https://doi.org/10.48550/arXiv.1603.00831

[20] Bernardin, Keni, and Rainer Stiefelhagen, "Evaluating multiple object tracking performance: the clear mot metrics.", EURASIP Journal on Image and Video Processing May 2008, pp. 1-10, 2008, DOI: https://doi.org/10.1155/2008/246309

[21] Pang, Jiangmiao, et al., "Quasi-dense similarity learning for multiple object tracking.", Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 164-173, Jun. 2021, DOI: https://doi.org/10.1109/CVPR46437.2021.00023

[22] Peng, Jinlong, et al., "Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking.", European conference on computer vision, pp. 145-161, Oct. 2020, DOI: https://doi.org/10.1007/978-3-030-58548-8_9

[23] Babaee, Maryam, Zimu Li, and Gerhard Rigoll, "A dual cnn–rnn for multiple people tracking.", Neurocomputing 368, pp. 69-83, Nov. 2019, DOI: https://doi.org/10.1016/j.neucom.2019.08.008

[24] Lin, Xufeng, et al., "On the detection-to-track association for online multi-object tracking.", Pattern Recognition Letters 146, pp. 200-207, Jun. 2021, DOI: https://doi.org/10.1016/j.patrec.2021.03.022

[25] Wu, Jialian, et al., "Track to detect and segment: An online multi-object tracker.", Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp, 12352-12361, Jun. 2021, DOI: https://doi.org/10.1109/CVPR46437.2021.01217

[26] Pang, Bo, et al., "Tubetk: Adopting tubes to track multi-object in a one-step training model.", Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 6308-6318, Jun. 2020, DOI: https://doi.org/10.1109/CVPR42600.2020.00634

[27] Wang, Yongxin, Kris Kitani, and Xinshuo Weng, "Joint object detection and multi-object tracking with graph neural networks." 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 13708-13715, 2021, DOI: https://doi.org/10.1109/ICRA48506.2021.9561110

[28] Zhou, Xingyi, Vladlen Koltun, and Philipp Krähenbühl, "Tracking objects as points.", European Conference on Computer Vision, pp. 474-490, Oct. 2020, DOI: https://doi.org/10.1007/978-3-030-58548-8_28

[29] Zheng, Linyu, et al., "Improving multiple object tracking with single object tracking.", Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2453-2462, Jun. 2021, DOI: https://doi.org/10.1109/CVPR46437.2021.00248

[30] Tokmakov, Pavel, et al., "Learning to track with object permanence.", Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10860-10869, Oct. 2021, DOI: https://doi.org/10.1109/iccv48922.2021.01068

Authors

Dae-Hyeon Park received the BS degree in Computer Engineering from Inha University in 2020, and is currently pursuing the MS degree with the Department of Electronic Computer Engineering at Inha University, Korea. His current research interest include single/multi-object tracking, multi-modal learning, real-time system and self-attention mechanism.

Seong-Ho Lee received the BS degree in Computer Science and Engineering from Incheon National University in 2019 and received the MS degree with the Department of Electronic Computer Engineering at Inha University, Korea. He is currently a full-time researcher at Inha University. His current research interest include multi-object tracking, object detection, generative adversarial networks, multi-scale representation, and self-supervised learning.

Seung-Hwan Bae received the BS degree in information and communication engineering from Chungbuk National University, in 2009 and the MS and PhD degrees in information and communications from the Gwangju Institute of Science and Technology (GIST), in 2010 and 2015, respectively. He was a senior researcher at Electronics and Telecommunications Research Institute (ETRI) in Korea from 2015 to 2017. He was an assistant professor in the Department of Computer Science and Engineering at Incheon National University, Korea from 2017 to 2020. He is currently an Associate Professor with the Department of Computer Engineering at Inha University, His research interests include multi-object tracking, object detection, deep learning, feature learning, medical image analysis, generative adversarial networks, face forensic, etc.