

궤적 예측 및 이미지 유사도 측정에 기반한 다중 카메라 객체 추적 방법

김성종*¹, 이하은*², 송석일**

Multi-camera Object Tracking Method based on Trajectory Prediction and Image Similarity Measurement

Seongjong Kim*¹, Haeun Lee*², and Seokil Song**

2021년 한국교통대학교 지원을 받아 수행하였음

요 약

이 논문에서는 다중 카메라에서 동일 객체를 지속적으로 추적하기 위한 객체 재식별 방법을 제안한다. 다중 카메라 네트워크에서 각 카메라의 설치 상황에 따라 서로 다른 시점, 높이, 조도를 가지며 인접하는 카메라가 감지하는 영역간의 거리가 서로 다르다. 이 논문에서 제안하는 방법은 다중 카메라 네트워크에서 다양한 설치 환경에 따라 객체의 궤적을 이용한 위치 추정을 통해서 재식별하는 방법과 시각적 특징을 이용해 재식별하는 방법을 결합한다. 제안하는 방법은 다중 카메라 설치 상황에 따라 위치 기반 재식별 방법과 시각적 특징 기반 재식별 방법의 가중치를 조절하여 정확도를 높인다. 이 논문에서는, 제안하는 방법을 구현하고 실제 도로에 설치한 다중 카메라에서 수집한 데이터를 기반으로 성능평가를 수행하여 제안하는 방법의 우수성을 보인다.

Abstract

This paper proposes a method for re-identifying objects continuously tracked in multi-camera network. In a multi-camera network, each camera has different viewpoints, heights, and lighting conditions, and the distance between the regions detected by adjacent cameras is also different depending on their installation. The proposed method combines re-identification based on the object's location prediction and re-identification based on the visual features of objects. It enhances accuracy by adjusting the weights of location-based re-identification and visual feature-based re-identification methods according to the installation conditions. The proposed method is implemented and evaluated using data collected from multi-cameras installed on real roads to show its effectiveness.

Keywords

object Re-ID, multiple camera, object tracking, SNN

* 한국교통대학교 컴퓨터공학과 석사과정
- ORCID¹: <https://orcid.org/0000-0003-1589-728X>
- ORCID²: <https://orcid.org/0009-0009-0816-103X>
** 한국교통대학교 컴퓨터공학과 교수(교신저자)
- ORCID: <https://orcid.org/0000-0002-0110-7155>

· Received: Apr. 30, 2023, Revised: May 18, 2023, Accepted: May 21, 2023
· Corresponding Author: Seokil Song
School of Computer, Korea National University of Transportation,
Daehakro 50, Chungju, Chungbuk 27469, Korea
Tel.: +82-43-841-5349, Email: sisong@ut.ac.kr

1. 서론

영상 처리 기술이 발전하면서 지능형 교통 시스템(ITS)에 감시 카메라가 널리 사용되고 있다. 감시 카메라는 교통 모니터링 및 단순 데이터 수집뿐만 아니라 높은 정확도의 실시간 교통 감지 및 보안 관리 등에도 사용되고 있다[1]. 문헌 [2]에서는 V2X (Vehicle to everything) 통신기술을 기반으로 하는 C-ITS(Cooperative-Intelligent Transport Systems)[3]에서 도로 인프라 센서를 이용해 자율주행 차량의 인지 범위를 확장하는 방법을 제안하고 있다. 여기에서는 감시 카메라를 포함한 도로 인프라 센서를 통해 인지한 객체들을 V2X를 통해 자율주행 차량에 실시간으로 제공하여 자율차가 돌발상황에 신속하게 대응할 수 있도록 한다.

최근에는 다수 카메라 네트워크에서 다수 객체의 위치를 연속적으로 추적하는 MTMCT(Multi-Target Multi-Camera Tracking) 기술이 개발되고 있다[1]. 다수 카메라 네트워크에서 추적된 각 객체의 궤적은 이동시간 예측, 제한속도 위반 단속, 범죄 수사, 이상 행동 감지, 차량의 의도 파악 등을 가능하게 한다. 이러한 이유로 MTMC는 교통 감시 시스템에서 중요한 역할을 수행한다[4].

문헌 [4]에 의하면 일반적으로 MTMC는 두단계에 걸쳐서 수행된다. 첫 번째 단계에서는 각 단일 카메라에서 감지된 객체를 추적하여 로컬 트랙클릿(Tracklet)을 생성한다. 다음 단계에서는 카메라 네트

워크의 모든 카메라에 대해서 로컬 트랙클릿에 대한 교차 매칭(Matching)을 수행하여 각 객체에 대한 전역(Global) 궤적을 생성한다.

MTMC의 가장 핵심적인 기술은 객체 재식별(Re-Identification) 방법이다. 객체 재식별 방법은 카메라의 가려짐 및 시점 변경 때 객체를 효과적으로 재식별 할 수 있다. 이 특징은 분리된 영역을 감시하며 서로 시점이 다른 다중 카메라 네트워크에서 객체를 재식별하는데 효과적이다[5].

그림 1은 이 논문에서 수행하는 실험 대상 영역에 두 대의 카메라가 서로 중첩되지 않는 영역을 감지 하는 예를 보여준다. 한 대의 차량이 그림 1의 오른쪽 Infra_Cam1의 검지 영역에서 추적되고 있다. Infra_Cam1이 검지하는 영역을 벗어난 차량은 한동안 추적이 안 되다가 다시 그림 1의 왼쪽 Infra_Cam2가 검지하는 영역에 나타나고 다시 추적이 재개된다. 검지가 중단되는 시간은 교내 규정속도인 25km/h 기준으로 약 6초 정도이다. 해당 차량이 Infra_Cam2에 나타났을 때 Infra_Cam1에서 추적되던 차량인지를 판단할 수 있다면 두 카메라에 걸쳐 연속적인 추적이 가능하다.

기존에 다수의 객체 재식별 방법 기반의 MTMCT 기술이 제안된 바 있다. 대부분의 방법은 CNN(Convolutional Neural Network)을 기반으로 하여 각 프레임의 이미지 특징과 옵티컬 플로우(Optical flow), 차량 모양, 차량 번호판 등을 특징으로 학습한 모델을 이용하여 유사도를 측정한다[6]-[9].

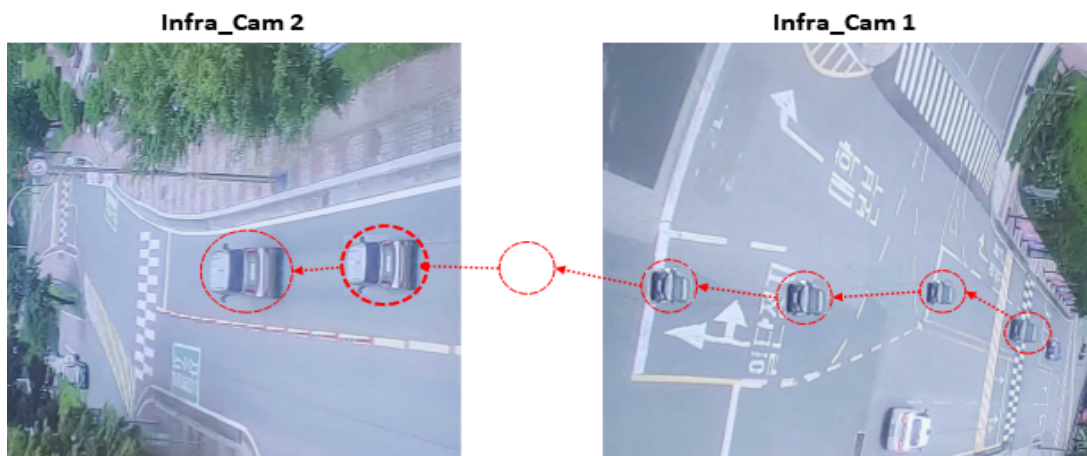


그림 1. 분리된 영역을 감지하는 두 대의 카메라에서 객체 추적 예
 Fig. 1. Example of object tracking using two cameras that detect separated areas

또한, 몇몇 방법은 각 카메라에서 추적한 궤적을 기반으로 차량을 재식별하기도 한다[10][11]. 일부 방법은 실시간으로 객체를 재식별하는 것이 아니라 각 카메라에서 추적한 궤적을 사후에 비교하는 방식을 사용한다[1].

다중 카메라 네트워크에서는 각 카메라의 설치 상황에 따라 시공간 특징과 시각적 특징을 이용한 각각의 객체 재식별 방법이 서로 다른 성능을 낼 수 있다. 인접한 두 카메라의 시점과 조도, 높이 등 시각적 특징의 차이가 큰 경우에는 시각적 특징을 이용하는 객체 재식별 방법의 정확도가 낮아질 수 있다. 반면 두 카메라가 겹치는 영역 간의 거리가 가까워서 객체가 추적되지 않는 시간이 짧은 경우에는 객체의 궤적특징(시공간 특징)을 이용한 재식별 방법의 정확도가 더 우수할 수 있다. 하지만, 기존의 방법들은 주로 이 두 접근 방법 중 하나를 취하고 있다. 이렇게 되면 실제 도로에 설치된 카메라의 상황에 따라서 성능이 결정되게 된다.

이 논문에서는 이러한 기존 방법의 문제점을 보완하기 위해 다중 카메라 네트워크에서 각 카메라의 설치 상황에 따라 시공간 특징과 시각적 특징을 서로 다른 비중으로 사용하는 객체 재식별 방법을 제안한다. 궤적을 이용한 재식별 방법은 먼저 하나의 카메라에서 추적한 객체의 궤적을 기반으로 다음 카메라의 어떤 위치에 해당 객체가 언제 나타날지를 예측한다. 다음으로, 다음 카메라에서 처음으로 인식된 객체의 MBB(Minimum Bounding Box)와 예측한 위치에서의 MBB간의 IoU(Intersection over Union)를 측정하여 유사도를 측정한다.

시각적 특징을 이용한 재식별 방법은 하나의 카메라에서 객체가 마지막으로 인식되었을 때의 MBB 이미지를 추출하고, 다음 카메라에서 인식된 객체의 MBB 이미지를 추출한다. 추출된 두 MBB 이미지를 CNN의 일종인 SNN(Siamese Neural Network)[6]을 이용해서 유사도를 측정한다.

예측된 위치기반 IoU 값과 MBB 이미지간 유사도를 이용하여 최종 유사도를 계산하여 객체 재식별을 수행한다. 최종 유사도 계산시에는 각 유사도에 대한 가중치를 반영한다. 이 논문에서는 그림 1에서와 같이 실제 도로에 설치한 두 대의 카메라에서 제안하는 객체 재식별 방법을 실험하여 가능성을 입증한다.

II. 관련 연구

다중 카메라 네트워크에서 연속적인 객체 추적을 위한 가장 핵심적인 기술은 객체 재식별 기술이다. 객체 식별문제는 다중 카메라 네트워크에서 객체가 이동할 때 동일 객체의 서로다른 인스턴스에 대해 동일 라벨(Label)을 할당하는 절차로 정의할 수 있다 [4]. 약 10여년 이상 다중 카메라 네트워크에서 객체를 재식별하는 연구가 활발히 진행되고 있다. 최근의 연구 경향은 SNN[6][7] 기반의 방법, 다중 스트림 CNN 기반의 방법[8][9], 연속적인 시간에서의 패턴을 이용하는 방법 등으로 나누어 볼 수 있다[10][11].

SNN 기반의 방법들은 다중 카메라에서 추출한 객체 이미지 쌍들에 대해서 동일 객체는 거리를 최소화하고 다른 객체들에 대해서는 거리를 최대화할 수 있도록 모델을 학습하고 이를 이용해서 동일 객체를 식별하는 방법을 사용한다. 다중 스트림 CNN기반의 방법들은 두가지 이상의 서로다른 특징을 이용하여 모델을 학습한다. 각 프레임의 이미지 특징과 옵티컬 플로우(Optical flow), 차량 모양, 차량 번호판 등을 삼 신경망 등을 이용하여 학습한다. [10][11]는 연속적인 시간에서의 패턴을 이용하는 방법으로 다중의 카메라에서 나타나는 객체의 시간별 위치 정보를 같이 이용한다.

이 논문에서는 궤적 예측방법을 적용하여 객체 재식별을 수행한다. 궤적을 예측하는 방법은 물리적 방법(Physical-based methods)과 기계학습(Machine learning)을 이용하는 예측 방법들로 구분할 수 있다 [12]. 기계학습 기반의 예측방법들은 대부분 순환신경망(RNN, Recurrent Neural Network) 계열인 LSTM(Long Short Term Memory)을 이용하는 방법들이다.

기존 제안된 궤적 예측 방법들은 예측해야 하는 궤적의 길이 (시간 범위)가 길어질수록 정확도가 하락하는 경향을 보인다. 문헌 [13]에 따르면 방법에 따라 정확도 하락의 정도는 있지만 LSTM을 이용하는 방법들은 예측해야 하는 궤적의 크기가 1에서 5로 증가함에 따라 RMSE(Root Mean Squared Error)가 약 0.1~0.6 정도로 증가한다. 따라서, 인접한 카메라들의 겹치 영역간의 거리가 멀수록 객체의 궤적 예측 정확도가 하락하여 객체 재식별 정확도가 하락 할 수 있다.

III. 제안하는 다중 카메라에서 객체 재식별 방법

그림 2는 이 논문에서 제안하는 다중 카메라 간 객체 재식별 방법의 전체적인 구조이다. 그림 2는 두 개의 카메라간에 객체를 연속적으로 추적하기 위해 이전 인프라 카메라(Previous infra camera)에서 현재 인프라 카메라(Current infra camera) 검지 영역으로 이동하는 객체에 대해 재식별을 수행하는 방법을 보여준다.

그림에서 궤적 예측 관리자 (Trajectory prediction manager)는 이전 인프라 카메라에서 객체의 궤적을 분석하여 현재 인프라 카메라에서 해당 시간에 객체의 위치를 예측한다. IoU 관리자 (IoU Manager)는 객체의 예측된 위치에서 MBB(Minimum Bounding Box)와 현재 인프라 카메라에서 검지된 객체의 MBB를 비교하여 동일 여부를 판단한다. 삼 신경망 유사도 관리자(Siamese neural network similarity manager)는 이전 인프라 카메라에서 마지막 객체 이미지와 현재 인프라 카메라의 첫 번째 객체 이미지 간의 시각적 특징 유사도를 비교한다.

이 방법은 현재 인프라 카메라가 검지 영역에 처음으로 진입한 객체를 감지할 때 시작된다. 제안하는 방법은 먼저 현재 인프라 카메라에서 검출한 객체의 위치에 따라 해당 방향으로 인접한 영역을 검지하는 이전 인프라 카메라를 선택한다. 다음으로,

선택된 이전 인프라 카메라에서 현재 인프라 카메라 검지 영역 방향으로 이동한 객체들 중 아직 현재 인프라 카메라에서 동일 객체로 인식되지 않은 객체들을 선택한다.

다음으로 선택된 각 객체에 대해서 객체의 과거 N개 위치목록을 궤적 예측 관리자에 입력하여 현재 인프라 카메라에서 객체를 검지한 시간의 위치를 예측한다. IoU 관리자는 예측이 완료된 후 예측한 위치에서 객체의 MBB와 현재 인프라 카메라 검지기가 검출한 객체의 MBB의 IoU를 계산한다. IoU는 예측한 객체 위치에서의 MBB와 현재 인프라 카메라로 검출한 MBB의 교집합 영역을 합집합 영역으로 나눈 값이다.

이 논문에서는 LSTM 기반의 기본적인 궤적 예측 방법을 이용한다. 하지만, 서론에서 언급한 것처럼 다중 카메라 네트워크의 설치 상황에 따라 시공간 특징을 이용하는 재식별 방법과 시각적 특징을 이용하는 재식별 방법의 정확도가 차이가 발생할 수 있다.

이에 따라 이 논문에서는 CNN의 일종인 SNN을 이용해 동일 객체 여부를 판단하는 방법을 같이 사용한다. 제안하는 방법의 삼 신경망 유사도 관리자는 이전 인프라 카메라에서 마지막으로 검출한 객체의 MBB 이미지와 동일한 객체를 현재 인프라 카메라에서 처음으로 검출했을 때 MBB 이미지를 동일 객체로 하여 SNN을 학습한다.

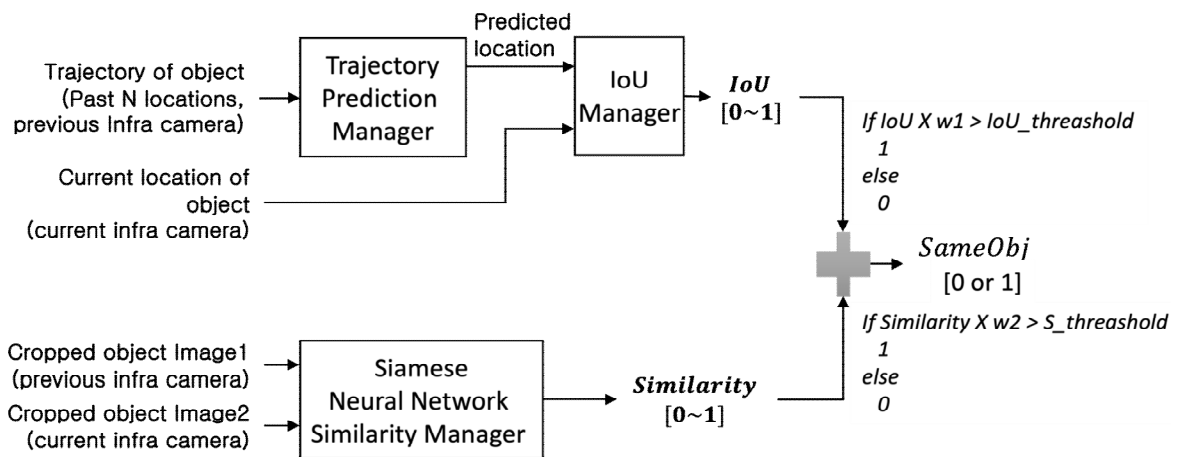


그림 2. 제안하는 위치 예측 및 이미지 유사도 측정 기반 객체 재식별 방법의 전체 구조

Fig. 2. Overall architecture of the proposed object re-identification method based on location prediction and image similarity measureme

이후 학습한 모델을 이용하여 동일 객체여부를 판단해야할 이미지들의 특징을 SNN을 이용하여 추출하고 이를 임베딩(Embedding)한 값을 비교하여 유사도(Similarity)를 계산하게 된다. 이 과정은 그림 2의 하단에 표현되어 있다.

최종적으로 두 객체의 동일 여부는 IoU와 유사도를 이용하여 판단하게 된다. 이때 두 인프라 카메라의 검지 영역간의 간격이 얼마 되지 않아서 궤적 예측 정확도가 충분히 높다면 IoU의 비중을 높이고, 검지영역간의 거리가 멀어서 예측 정확도 하락이 예상 될 경우 유사도의 비중을 높여서 최종 동일 여부를 확정하게 된다. 그림 2에서 w_1 과 w_2 는 각 방법에 대한 가중치로 0~1 사이의 값을 갖는다.

객체 재식별 결과는 IoU와 유사도에 대해 가중치를 반영하여 계산한 결과가 일정 값 이상일 경우 판정하게 된다. IoU에 가중치를 반영한 결과가 $IoU_threshold$ 이상이면 1, Similarity에 가중치를 반영한 결과가 $S_threshold$ 이상이면 1로 하며 최종 재식별 여부는 각 결과에 대한 OR 연산으로 판단한다. 즉, 하나라도 1이 나오면 재식별 결과 True로 판정한다.

이렇게 하는 이유는 $IoU_threshold$ 와 $S_threshold$ 를 실험을 통해 충분히 어느 하나라도 True이면 충분히 재식별 결과를 True로 할 수 있도록 설정하기

때문이다. 또한, 다중 카메라 설치 환경에 따라 IoU 값과 Similarity 값 각각에 적정한 가중치를 곱하기 때문에 그 결과가 $threshold$ 이상이 된다면 충분히 객체 재식별 결과를 True로 할 수 있다. 하지만 이것은 각 카메라 설치 환경에 따라 OR 연산이 아닌 AND 연산을 통해서 최종 재식별 판단을 하도록 할 수도 있다. 이 논문에서는 OR 연산을 기본으로 하여 실험을 진행한다.

그림 3은 SNN을 이용한 객체 MBB 이미지간의 유사도 측정방법을 보여주고 있다. 인프라 카메라는 시점에 따라 그림 3에서처럼 동일 객체의 앞면이나 후면을 촬영할 수 있다. 이들 시점이 다른 동일 객체의 두 객체 MBB 이미지의 유사도를 측정하기 위해서 이전 인프라 카메라의 MBB 이미지를 앵커(Anchor), 현재 인프라 카메라의 MBB 이미지를 양성(Positive), 임의의 MBB 이미지를 음성(Negative)로 어노테이션 하여 학습데이터를 구축한다. 구축한 학습데이터를 이용하여 SNN을 학습하고 모델을 생성한다.

w_1 과 w_2 는 실험을 통해서 정해지며 이 논문에서는 w_1 은 0.8, w_2 는 1로 하여 실험을 진행하였다. 또한, $IoU_threshold$ 와 $S_threshold$ 는 각각 0.5, 0.9999로 하였다. $S_threshold$ 는 실험을 통해 결정된 값이며 $IoU_threshold$ 는 일반적으로 객체 인식에서 사용하는 IoU 값을 적용하였다.

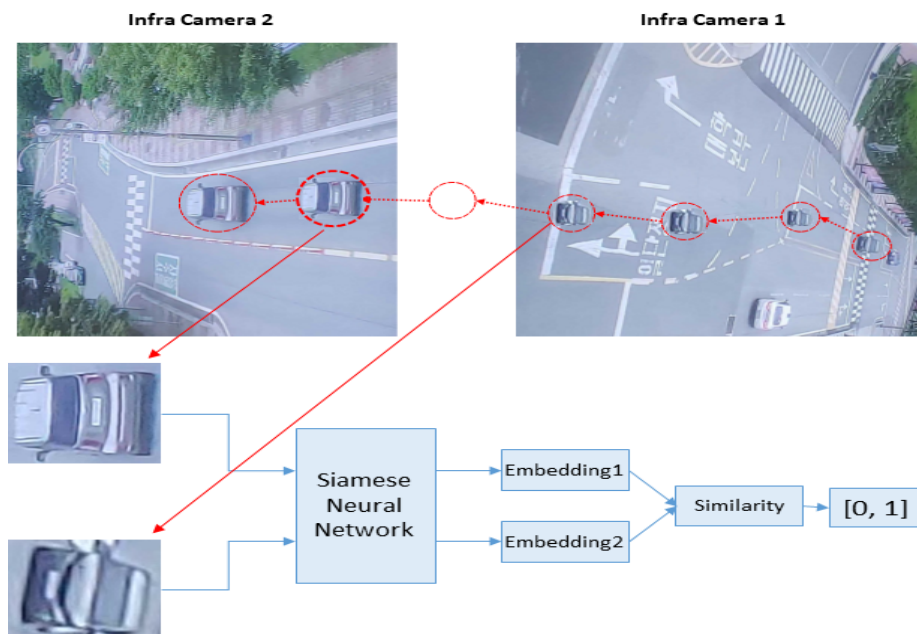


그림 3. SNN을 이용한 두 MBB 이미지간 유사도 측정
Fig. 3. Measuring similarity between two MBB images using SNN

IV. 성능 평가

이 논문에서는 제안하는 방법의 실효성을 평가하기 위해 실험을 진행하였다. 제안하는 방법은 LSTM 기반의 궤적 예측과 이를 통한 MBB의 IoU 측정방법과 SNN 기반의 이미지 유사도 측정 방법을 결합한다. 설치된 인프라 카메라들의 검지 영역간의 간격에 따라 IoU 및 유사도 비중을 조절하게 된다.

이 논문에서 수행한 실험의 환경은 표 1과 같고 사용한 SNN과 LSTM의 파라미터는 표 2와 같이 설정하였다. SNN학습을 위해 사용한 데이터 집합은 그림 1과 같은 환경에서 수집한 총 1,284개의 Infra_Cam 1과 Infra_Cam 2의 동일 객체 MBB 이미지 쌍이다. 실제 학습을 수행할때는 임의의 MBB 이미지를 추가하여 (앵커, 양성, 음성) 쌍을 만들어 입력한다. 위치 예측을 위한 LSTM학습을 위해서는 총 1,284의 궤적을 수집하였다. 각 궤적은 Infra_Cam 1에서 30 포인트와 Infra_Cam 2에서의 1 포인트로 구성된다.

표 1. 실험환경

Table 1. Experimental environments

Division	Contents
HW	GPU : TITAN XP CPU : i7-7700 3.6GHz (8 cores) RAM : 64GB
SW	Ubuntu 20.04.5 LTS Tensorflow 2
SNN data set	1,284 pairs (Infra_Cam 2, Infra_Cam 1) Train : Valid : Test = 0.8 (1,156 pairs) : 0.1 (128 pairs) : 0.1 (128 pairs)
Location prediction data set	1,284 trajectories (50 points of Infra_Cam) Train : Valid : Test = 0.8 (1,156) : 0.1 (128) : 0.1 (128)

표 2. SNN 과 LSTM의 파라미터

Table 2. Parameters of SNN and LSTM

Division	Contents
SNN	epoch : 100 loss : TripletLoss
LSTM	epoch : 2000 window : 25 horizon : 1 input_dim : 2 hidden_dim : 128 num_layers : 4 output_dim : 2

그림 4는 실제 실험에 사용된 MBB 이미지 학습 데이터의 일부를 보여준다. 그림에서처럼 Infra_Cam 1과 Infra_Cam 2에서 인식한 동일 객체에 대해서 앵커 및 양성 이미지로 어노테이션하고 임의의 이미지를 하나 추가하여 음성 이미지로 어노테이션한 결과를 SNN에 입력한다.

실험에서 사용한 파라미터는 앞서 기술한 바와 같이 w_1 은 0, w_2 은 1.0으로 하였으며 $IoU_{threshold}$ 와 $S_{threshold}$ 는 각각 0.5, 0.9999로 하였다. w_1 을 0으로 한 것은 Infra_Cam 1과 2 사이에 검지가 안되는 시간이 약 6초로 다소 예측해야 하는 시간 간격이 크다고 판단하였기 때문이다. 또한, $S_{threshold}$ 를 0.9999로 설정한 것은 SNN 학습 모델을 테스트 했을 때 양성과 음성 MBB 이미지를 구분하는 Similarity가 0.9999일 때 정확도가 가장 높은 1.0에 달하였기 때문이다.



그림 4. SNN 학습데이터의 일부

Fig. 4. Set of data of SNN trained data set

그림 5는 학습된 SNN 모델로 앵커 이미지와 양성 및 음성 이미지에 대한 유사도를 측정된 것의 일부이다. 그림에서 양성 이미지는 대부분 유사도가 0.9999 이상이며, 음성 이미지는 대부분 유사도가 0.9999 이하임을 볼 수 있다.

이렇게 설정한 후 최종 객체 재식별을 수행하는 실험을 수행하였을 때 총 128건의 테스트 데이터에 대해서 1.0의 정확도를 보임을 확인하였다.

1 image Positive similarity: 0.9999281 Negative similarity: 0.9998001	21 image Positive similarity: 0.99991786 Negative similarity: 0.9997822	41 image Positive similarity: 0.9999207 Negative similarity: 0.9997816
2 image Positive similarity: 0.99991643 Negative similarity: 0.99979186	22 image Positive similarity: 0.99991834 Negative similarity: 0.99978346	42 image Positive similarity: 0.9999388 Negative similarity: 0.9997879
3 image Positive similarity: 0.99993426 Negative similarity: 0.99978304	23 image Positive similarity: 0.99993443 Negative similarity: 0.99981105	43 image Positive similarity: 0.99993235 Negative similarity: 0.9998025
4 image Positive similarity: 0.999919 Negative similarity: 0.99974763	24 image Positive similarity: 0.9999325 Negative similarity: 0.99980384	44 image Positive similarity: 0.9999114 Negative similarity: 0.99977326
5 image Positive similarity: 0.99992883 Negative similarity: 0.99981195	25 image Positive similarity: 0.9999289 Negative similarity: 0.99981666	45 image Positive similarity: 0.99993193 Negative similarity: 0.9997877
6 image Positive similarity: 0.9999209 Negative similarity: 0.99979335	26 image Positive similarity: 0.9999177 Negative similarity: 0.99975836	46 image Positive similarity: 0.9999149 Negative similarity: 0.99978805
7 image Positive similarity: 0.9999158 Negative similarity: 0.99980223	27 image Positive similarity: 0.9999209 Negative similarity: 0.9998083	47 image Positive similarity: 0.99991107 Negative similarity: 0.9998011
8 image Positive similarity: 0.99992 Negative similarity: 0.99976885	28 image Positive similarity: 0.9999247 Negative similarity: 0.99976814	48 image Positive similarity: 0.99991924 Negative similarity: 0.99979126
9 image Positive similarity: 0.9999162 Negative similarity: 0.9998031	29 image Positive similarity: 0.9999051 Negative similarity: 0.99975884	49 image Positive similarity: 0.9999339 Negative similarity: 0.9997505
10 image Positive similarity: 0.99992883 Negative similarity: 0.99978125	30 image Positive similarity: 0.9999361 Negative similarity: 0.9998175	50 image Positive similarity: 0.9999298 Negative similarity: 0.99978
11 image Positive similarity: 0.9999331 Negative similarity: 0.9998028	31 image Positive similarity: 0.9999316 Negative similarity: 0.99979633	51 image Positive similarity: 0.99993324 Negative similarity: 0.9997746
12 image Positive similarity: 0.99991417 Negative similarity: 0.99977696	32 image Positive similarity: 0.9999101 Negative similarity: 0.999758	52 image Positive similarity: 0.999932 Negative similarity: 0.99975896
13 image Positive similarity: 0.99992126 Negative similarity: 0.999773	33 image Positive similarity: 0.9999338 Negative similarity: 0.9998046	53 image Positive similarity: 0.9999229 Negative similarity: 0.9997787
14 image Positive similarity: 0.99991727 Negative similarity: 0.9997871	34 image Positive similarity: 0.9999335 Negative similarity: 0.999799	54 image Positive similarity: 0.9999068 Negative similarity: 0.99977463
15 image Positive similarity: 0.99993175 Negative similarity: 0.9998019	35 image Positive similarity: 0.99992687 Negative similarity: 0.9998032	55 image Positive similarity: 0.9999102 Negative similarity: 0.9997797
16 image Positive similarity: 0.9999145 Negative similarity: 0.9997781	36 image Positive similarity: 0.99992037 Negative similarity: 0.9997832	56 image Positive similarity: 0.9999194 Negative similarity: 0.9997552
17 image Positive similarity: 0.999908 Negative similarity: 0.99976337	37 image Positive similarity: 0.99992275 Negative similarity: 0.9997633	57 image Positive similarity: 0.999928 Negative similarity: 0.9997731
18 image Positive similarity: 0.9999187 Negative similarity: 0.9997709	38 image Positive similarity: 0.9999216 Negative similarity: 0.9997647	58 image Positive similarity: 0.99991554 Negative similarity: 0.9997691
19 image Positive similarity: 0.9999315 Negative similarity: 0.99978966	39 image Positive similarity: 0.9999232 Negative similarity: 0.99980843	59 image Positive similarity: 0.99991524 Negative similarity: 0.9997617
20 image Positive similarity: 0.9999337 Negative similarity: 0.9998077	40 image Positive similarity: 0.99991035 Negative similarity: 0.99973077	60 image Positive similarity: 0.9999082 Negative similarity: 0.9997757

그림 5. SNN 테스트 결과 일부 : 앵커(Anchor)에 대한 양성(Positive)와 음성(Negative) 이미지의 유사도 로그
 Fig. 5. Partial results of SNN test: similarity log of positive(Anchor) and negative images

실제 환경에서 데이터를 수집하고 어노테이션을 수행하는데 있어 시간 제약 때문에 약 1000건의 데이터만을 대상으로 실험을 수행한 것이어서 향후 추가적인 실험이 필요하지만, 제안하는 방법의 기본적인 성능은 확인 가능하였다.

다음으로는 수행시간을 측정하였다. LSTM과 SNN 각각을 순차적으로 실행하고 그 결과를 종합하는데 소요되는 시간을 측정하였다. 대부분의 시간은 LSTM과 SNN을 수행하는데 소요되었으며 그 시간은 다음 표 3과 같다. 총 시간은 약 0.01초로 실시간 객체 재식별에 문제가 없다고 판단된다.

표 3. LSTM과 SNN의 수행시간
 Table 3. Execution time of SNN and LSTM

Division	Execution time (seconds)
SNN	0.0056
LSTM	0.0062
Total	0.0118

V. 결 론

이 논문에서는 협력 자율주행에서 다중 인프라 카메라에 의해 감지되는 객체들을 연속적으로 추적하기 위한 방법을 제안하였다. 제안하는 방법은 LSTM 기반의 궤적 예측 방법과 객체 이미지 간의

유사도를 측정하는 방법을 결합하여 정확도를 높인다. 실험 결과 객체 재식별 정확도가 1.0으로 매우 높은 것을 확인하였다. 향후에는 추가적인 카메라 투입 및 데이터 수집을 통해 보다 확장성 있는 실험을 수행할 계획이다. 또한, 타 방법과의 비교 평가도 수행할 계획이다.

References

- [1] H. Yang, J. Cai, M. Zhu, C. Liu, and Y. Wang, "Traffic-Informed Multi-Camera Sensing (TIMS) System Based on Vehicle Re-Identification", in *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 10, pp. 17189-17200, Oct. 2022. <https://doi.org/10.1109/TITS.2022.3154368>.
- [2] C. Mun, "Development of advanced technology for road situation awareness based on infrastructure sensors", *Korean Society of Transportation 86th Conference*, Vol. 86, pp. 547-551, Apr. 2022.
- [3] J. Ko, J. Jang, and C. Oh, "Assessing the safety benefits of in-vehicle warning information by vehicle interaction analysis in C-ITS environments", *Journal of Korean Society of Transportation*, Vol. 39, No. 1, pp. 1-13, Feb. 2021. <https://doi.org/10.7470/jkst.2021.39.1.001>.

- [4] I. Oliveira, R. Laroca, D. Menotti, K. V. Ono Fonseca, and R. Minetto, "Vehicle-Rear: A new dataset to explore feature fusion for vehicle identification using convolutional neural networks", IEEE Access, Vol. 9, pp. 101065-101077, Jul. 2021. <https://doi.org/10.1109/ACCESS.2021.3097964>.
- [5] D. N. N. Tran, L. H. Pham, H. J. Jeon, H. H. Nguyen, H. M. Jeon, T. H. P. Tran, and J. W. Jeon, "A Robust Traffic-Aware City-Scale Multi-Camera Vehicle Tracking Of Vehicles", I2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, New Orleans, LA, USA, Aug. 2022. <https://doi.org/10.1109/CVPRW56347.2022.00355>.
- [6] Y. Bai, Y. Lou, F. Gao, S. Wang, Y. Wu, and L. Y. Duan, "Group-sensitive triplet embedding for vehicle reidentification", IEEE Transactions on Multimedia, Vol. 20, No. 9, pp. 2385-2399, Jan. 2018. <https://doi.org/10.1109/TMM.2018.2796240>.
- [7] X. Liu, W. Liu, T. Mei, and H. Ma, "Provid: Progressive and multimodal vehicle reidentification for large-scale urban surveillance", IEEE Transactions on Multimedia, Vol. 20, No. 3, pp. 645-658, Mar. 2018. <https://doi.org/10.1109/TMM.2017.2751966>.
- [8] D. Chung, K. Tahboub, and E. J. Delp, "A two stream siamese convolutional neural network for person re-identification", 2017 IEEE International Conference on Computer Vision, Venice, Italy, Dec. 2017. <https://doi.org/10.1109/ICCV.2017.218>.
- [9] I. Oliveira, K. V. Fonseca, and R. Minetto, "A two-stream siamese neural network for vehicle re-identification by using non-overlapping cameras", In 2019 IEEE International Conference on Image Processing (ICIP), pp. 669-673, Feb. 2019. <https://doi.org/10.48550/arXiv.1902.01496>.
- [10] Y. Shen, T. Xiao, H. Li, S. Yi, and X. Wang, "Learning deep neural networks for vehicle re-id with visual-spatio-temporal path proposals", 2017 IEEE International Conference on Computer Vision, Venice, Italy, pp. 1900-1909, Oct. 2017. <https://doi.org/10.1109/ICCV.2017.210>.
- [11] Y. Lou, Y. Bai, J. Liu, S. Wang, and L. Y. Duan, "Embedding adversarial learning for vehicle re-identification", IEEE Transactions on Image Processing, Vol. 28, No. 8, pp. 3794-3807, Feb. 2019. <https://doi.org/10.1109/TIP.2019.2902112>.
- [12] L. Lin, W. Li, H. Bi, and L. Qin, "Vehicle Trajectory Prediction Using LSTMs With Spatial-Temporal Attention Mechanisms", IEEE Intelligent Transportation Systems Magazine, Vol. 14, No. 2, pp. 197-208, Feb. 2021. <https://doi.org/10.1109/MITS.2021.3049404>.
- [13] W. S. Choi and C. S. Ahn, "Deep Learning based Lane Change and Vehicle Trajectory Prediction Method", The Korean Society of Mechanical Engineers Conference, Vol. 2020, No. 12, pp. 28-29, Dec. 2020.
- [14] S. J. Kim and S. I. Song, "Virtualization Method for Multiple Road Infra Camera", Korean Society of Transportation 87th Conference, Vol. 87, pp. 300-301, Sep. 2022.

저자소개

김성종 (Seongjong Kim)



2022년 2월 : 한국교통대학교
컴퓨터공학과(공학사)
2022년 3월 ~ 현재 :
한국교통대학교 컴퓨터공학과
석사과정
관심분야 : 인공지능, 이동객체,
데이터베이스, 컴퓨터 비전,

이하은 (Haeun Lee)



2022년 8월 : 한국교통대학교
컴퓨터공학과(공학사)
2022년 9월 ~ 현재 :
한국교통대학교 컴퓨터공학과
석사과정
관심분야 : 내용기반 영상검색,
ML, DL, GNN

송 석 일 (Seokil Song)



1998년 2월 : 충북대학교

정보통신공학과(공학사)

2000년 2월 : 충북대학교

정보통신공학과(공학석사)

2003년 2월 : 충북대학교

정보통신공학과(공학박사)

2003년 7월 ~ 현재 :

한국교통대학교 컴퓨터공학과 교수

관심분야 : 데이터베이스, 센서 네트워크, 스토리지
시스템