

멀티 UxV기반 객체 식별을 위한 다시점 객체 매칭 기술에 대한 연구

박진수, 신수용

IT융복합공학과

국립금오공과대학교

jp@kumoh.ac.kr, wdragon@kumoh.ac.kr

Research on multi-view object matching technique for multi-UxV-based object Identification

Jin Su Park, Soo Young Shin

Department of IT Convergence Engineering

Kumoh National Institute of Technology

요약

본 논문은 멀티 UxV를 활용한 객체 식별을 위해 다시점 영상에서 동일한 객체를 식별하는 시스템을 제안한다. 제안하는 방식은 2대 이상의 UxV를 통해 영상을 획득하고 영상에서 공간에 대한 특징점과 식별하고자 하는 객체의 특징점을 추출한다. 각 UxV는 DeepSORT를 사용해 인식된 객체에 ID를 부여해 이를 구별하여 관리하고, 다른 UxV에서 획득한 영상에 시야가 겹치는 부분이 있는지 판단한다. 시야가 겹치는 부분이 있다고 판단되면 그 특징점을 매칭해 영상간의 변환 관계를 추정한다. 추정된 변환 관계를 바탕으로 UxV에서 인식한 객체를 다른 UxV에서 인식한 객체와 매칭시켜 전체 시스템에서의 ID를 부여한다. 이를 통해 멀티 UxV를 사용하여 특정 객체를 추적할 때 발생하는 식별 문제를 해결할 수 있을 것으로 기대한다.

I. 서론

영상을 기반으로 하는 추적 기술은 사람의 힘으로 모니터링 하기 힘든 스마트 팩토리, 스마트 시티 등 대규모 시스템을 운용하는데 필수적이다. 최근에는 무인항공체(UxV - Unmanned Any Vehicle)를 기반으로 하는 모니터링 연구가 활발히 진행되고 있다. 기존의 고정형 모니터링 시스템과 달리 UxV를 사용하는 모니터링 시스템은 다양한 환경과 조건에 따라 유연한 운용이 가능하다. 고정형 시스템의 설치가 어려운 해양 환경, 장애물이 많아 음영지역이 많이 발생하는 산간지역과 도심 그리고 재난 현장의 탐색 등 그 활용성이 점차 확대되고 있다.

일반적으로 여러 카메라를 사용한 모니터링에는 객체의 식별을 위해 각 카메라 시점에 대한 상관관계를 활용한다. 고정형 시스템의 경우 각 카메라가 설치된 위치를 활용하여 상관관계를 추정하거나, 카메라의 시야가 겹치는 부분을 바탕으로 상관관계를 추정하여 객체를 식별할 수 있다. 하지만 고정형 시스템과 달리 UxV를 사용한 시스템은 그 자유도로 인해 다시점으로 영상을 획득했을 때 상관관계 추정을 통한 동일 객체 식별에 어려움이 있다. 이는 복수의 UxV를 사용해 지역을 모니터링하거나 특정 객체를 추적하는데 장애물로 작용한다.

이러한 문제를 해결하기 위해 다중 카메라 시스템에서의 다중 객체 추적(Multi-Camera Multi-Object Tracking, MCT)에 대한 연구가 활발히 진행되고 있다[1][2]. 이는 각 카메라의 획득 정보를 통합하고, 객체의 식별을 하는 것이 핵심이다. 이를 위한 대표적인 해결 방법으로는 개별 카메라에서 객체를 추적한 뒤 객체별로 재식별(Re-ID)을 하여 카메라간의 객체를 매칭하는 단계적 접근법과, 모든 관측 데이터를 통합하여 한 번에 최적의 매칭을 수행하는 글로벌 접근법이 있다.

본 논문에서는 단계적 접근법을 기반으로, 멀티 UxV를 사용해 획득한 다시점 영상에서 딥러닝 기반 객체 인식을 사용해 공간 정보와 객체 정보를 구분

한 뒤 공간과 객체에 대한 특징점을 추출한 뒤 중첩되는 공간의 특징점을 기반으로 1차 매칭을 하고, 객체의 특징점을 사용해 추가적인 매칭을 하여 객체를 식별하는 기술을 제안한다.

II. 본론

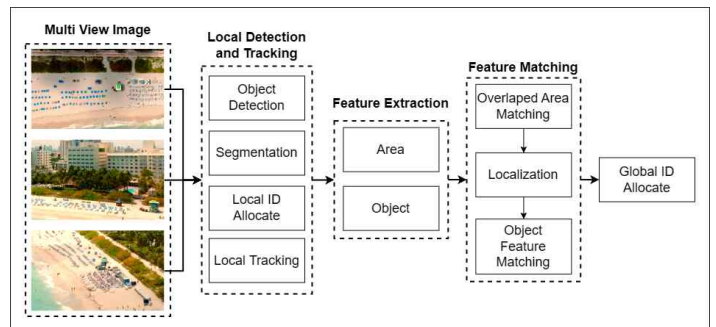


그림 1 제안하는 다시점 객체 매칭 기술의 순서도

그림 1은 제안하는 시스템의 순서도를 나타낸다. UxV를 사용해 지역을 모니터링 하며 객체를 인식한다. 객체가 인식되면 세그멘테이션을 통해 객체가 위치한 공간과 객체를 구분한다. 구분된 객체는 추적 알고리즘을 사용해 지역 ID를 할당하고 이를 추적한다. 이후 각 시점의 통합을 위해 객체가 위치한 공간과 객체의 특징을 별도로 추출한다. 먼저 추출한 공간의 특징을 사용해 중첩된 시야가 있는지 판단하고 중첩된 영역이 있다면 이를 바탕으로 각 시점의 상관관계를 추정한다. 추정된 공간의 상관관계를 통해 시점을 연결하고 객체 간 매칭을 시도한다. 이때 객체 간 매칭에 대한 점수를 책정하고 그 매칭이 일정 점수 이상일 경우 글로벌 ID를 할당하여 시스템 전체에서 객체를 식별하여 모니터링할 수 있도록 한다.

본 논문에서는 다양한 UxV 플랫폼 활용을 위해 다양한 크기와 복잡성의 모델을 지원하며, 특징 추출에 활용하기 위한 세그멘테이션 기능을 제공하는 Yolov8을 사용한다. 시스템의 객체 인식 및 추적 단계에서는 Yolov8를 사용하여 객체의 인식 및 세그멘테이션을 수행한다. 이후 추적 알고리즘을 통해 인식된 객체에 지역 ID를 부여하고 이동을 추적한다.

객체에 지역 ID를 부여하고 이를 추적하는 데는 다양한 알고리즘이 있다. 대표적인 추적 방식으로는 객체 추적과 인식을 동시에 수행하는 DeepSORT와 추적 대상의 특징을 추출하고 이를 학습하여 객체를 추적하는 Online Learning 기반 추적 방식이 있다. DeepSORT는 객체 인식 알고리즘을 이용해 객체의 위치와 그 특징을 추출하고 Kalman filter와 Hungarian Algorithm을 사용해 다음 프레임의 객체 위치를 예측하여 객체를 추적하는 알고리즘이다[3]. Online Learning 기반 추적은 지속적으로 입력되는 데이터를 학습하여 모델을 실시간으로 업데이트 하여 대규모 데이터가 없는 경우에 유용하다[4]. 본 논문에서는 멀티 UxV를 사용해 시점의 변화가 크고, UxV간의 시점 역시 차이가 크기에 실시간 학습 기반의 추적 알고리즘인 Online Learning이 아닌 프레임 간 객체 위치 예측을 수행하는 DeepSORT를 채택하였다.

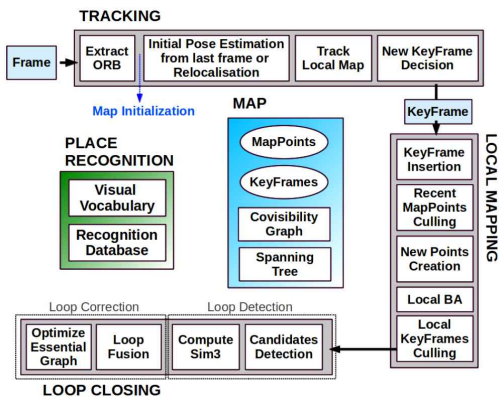


그림 2 ORB SLAM 시스템 구조

제안하는 시스템은 각 UxV에서 획득한 프레임에서 ORB-SLAM을 사용해 공간에 대한 특징점을 추출해 맵을 만들고 UxV의 위치를 파악한다[5]. 그림 2는 본 논문에서 사용한 ORB-SLAM의 시스템 구조를 나타낸다. ORB-SLAM은 단일 카메라로 프레임을 획득하여 ORB(Oriented FAST and Rotated BRIEF)를 사용해 특징점을 추출해 SLAM을 적용하는 알고리즘이다. 그림 2는 ORB SLAM의 시스템 구조를 나타낸다. ORB-SLAM은 특징점을 기반으로 지도에서 자신의 위치를 찾는 추적 스테드와 2D 이미지를 기반으로 프레임을 3D에 맵핑하는 지역 맵핑 스테드, 오차를 보정하기 위한 Loop Closing 스테드를 가지고 이를 병렬로 사용하여 빠르고 정확한 SLAM 기능을 제공한다.

제안하는 시스템은 ORB-SLAM을 사용해 맵을 생성하고 이를 바탕으로 각 UxV간의 상관관계를 파악한다. 만약 ORB-SLAM을 통해 UxV가 맵의 동일한 영역의 특징점을 획득하고 있으면 UxV가 같은 중첩된 시야를 가지고 있다고 판단한다. ORB-SLAM은 매칭된 2D 특징점과 3D 맵 포인트를 사용해 카메라의 위치와 자세를 추정한다. UxV가 중첩된 시야를 가지고 있다고 판단되면 추정된 카메라의 위치와 자세를 활용해 두 이미지에서 객체를 매칭한다. 객체의 매칭은 호모그래피 행렬을 사용한다[6]. 호모그래피는 2D 평면 상의 점들을 다른 2D평면으로 변환하는 방법이다. ORB를 사용해 추출한 공간의 특징점을 매칭하고 특징점 쌍을 사용해 호모그래피 행렬을 계산한다. 이 때 RANSAC 알고리즘을 사용해 특징점의 노이즈를 제거하여 안정적으로 호모그래피를 계산한다. 이후 객체의 특징점을 사용해 동일한 과정을 반복하고, 계산된 두 호모그래피를 대조해 유사도를 판별한다. 유

사도가 일정 수치 이상일 경우 매칭된 객체의 지역 ID를 사용하여 객체에 글로벌 ID를 할당한다.

III. 결론

본 논문은 공간과 객체의 특징점을 기반으로 하는 2단계의 매칭 단계를 통한 멀티 UxV기반 시스템을 위한 객체 매칭 시스템을 제안한다. 제안하는 시스템은 Yolov8과 DeepSORT를 기반으로 하는 지역 객체 인식 및 추적기능과 ORB-SLAM을 사용한 공간의 특징점 매칭, 호모그래피 행렬 기반 객체 매칭을 통해 UxV를 사용해 다시점에서 획득한 영상에서 동일한 객체를 식별하여 이를 추적할 수 있다. 본 논문은 시스템을 구성하는 핵심 기술들과 구성방식에 대해서 서술하고 있다. 추후 실제로 제안하는 시스템을 구현하여 시뮬레이션과 실제 환경에서 검증을 통해 멀티 UxV기반 시스템의 객체 식별 기술에 대한 발전에 기여하고자 한다.

ACKNOWLEDGMENT

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ICAN(ICT Challenge and Advanced Network of HRD) program(IITP-2024-RS-2022-00156394) supervised by the IITP(Institute of Information & Communications Technology Planning & Evaluation)

This work was supported by Innovative Human Resource Development for Local Intellectualization program through the Institute of Information & Communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (IITP-2024-2020-0-01612)

참고 문헌

- [1] Amosa, T. I., Sebastian, P., Izhar, L. I., Ibrahim, O., Ayinla, L. S., Bahashwan, A. A., Bala, A., & Samaila, Y. A. (2023). "Multi-camera multi-object tracking: A review of current trends and future advances." *Neurocomputing*, 552, 126558. <https://doi.org/10.1016/j.neucom.2023.126558>.
- [2] Z. Liu et al., "Robust Multi-Drone Multi-Target Tracking to Resolve Target Occlusion: A Benchmark," in *IEEE Transactions on Multimedia*, vol. 25, pp. 1462-1476, 2023, doi: 10.1109/TMM.2023.3234822.
- [3] 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, pp. 3645-3649, doi: 10.1109/ICIP.2017.8296962.
- [4] Abbass, M.Y., Kwon, K.C., Kim, N. et al. "A survey on online learning for visual tracking." *Vis Comput* 37, 993-1014 (2021). <https://doi.org/10.1007/s00371-020-01848-y>
- [5] R. Mur-Artal, J. M. M. Montiel and J. D. Tardós, "ORB-SLAM: A Versatile and Accurate Monocular SLAM System," in *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147-1163, Oct. 2015, doi: 10.1109/TRO.2015.2463671.
- [6] Basic concepts of the homography explained with code. OpenCV. (2024). Retrieved, May, 2024 https://docs.opencv.org/4.x/d9/dab/tutorial_homography.html