

인식된 문자의 강한 특징점을 활용하는 측위시스템

송도훈, 박종일†

한양대학교

dohoons@hanyang.ac.kr, jipark@hanyang.ac.kr

Odometry Using Strong Features of Recognized Text

Do-hoon Song, Jong-il Park

Hanyang University

요 약

본 논문에서는 시각-관성 측위시스템(Visual-Inertial Odometry, VIO)에서 광학 문자 인식(Optical Character Recognition, OCR)을 활용해 문자의 영역을 찾아내고, 그 위치를 기억해 측위시스템에서 다시 인식되었을 때 비교하기 위해 위치와 특징점을 저장하고자 한다. 먼저, 실시간으로 움직이는 카메라의 영상에서 문자를 찾아내고, 카메라의 상대적인 위치를 이용하여 문자가 인식된 위치와 특징점을 저장하는 방법을 제안한다. 또한 저장된 문자가 다시 탐색되었을 때, 문자가 재인식되었는지 판별하기 위한 방법을 제안한다. 인공적인 마커나 미리 학습된 객체를 사용하지 않고 상황에 따른 문자를 사용하는 이 방법은 문자가 존재하는 범용적인 공간에서 사용이 가능하다.

1. 서론

최근에 하드웨어의 소형화와 성능의 향상에 따라 로봇이나 드론에 장착되는 카메라와 연산장치의 성능도 급속도로 상승하고 있다. 시각-관성 측위시스템(Visual-Inertial Odometry, 이후 VIO)은 주어진 영상과 IMU 의 데이터를 동기화하여 미터법 기준으로 카메라의 위치를 추정하는 방법으로, 이동한 경로를 추적하는 분야로 알려져 있다. 이 때, IMU 의 가속도계를 사용하여 카메라의 이동거리를 미터법으로 변환하여 GPS 보다 더 정확한 위치를 획득할 수 있다. 이러한 VIO 는 로봇이나 드론에 카메라와 IMU 를 장착하여 사람이 진입하기 힘든 위험한 지역이나 공중에서 주변 환경의 정보를 획득하는 분야에서 사용되고 있다.

GPS 보다 정확하고 실시간으로 획득할 수 있음에도 불구하고 VIO 로 획득한 이동경로에도 오차가 발생한다. 매우 민감하게 동작하는 IMU 는 기본적으로 일정하게 고정된 신호를

획득하는 것이 아니라 계속 변화하고 있는 신호를 획득하기 때문에 오차가 지속적으로 발생한다. 또한, 카메라에서 획득한 영상은 급격한 움직임이 발생하면 모션 블러(Motion Blur)가 발생하게 되어 광학 흐름(Optical Flow)을 계산하는 과정에서 정보가 손실된다.

본 논문에서는 이상의 문제점을 해결하기 위해서 광학 문자 인식(Optical Character Recognition, 이후 OCR)을 사용하여 강력한 특징점을 활용하는 방법을 제안한다. 실시간으로 동작할 수 있는 OCR 에 대한 선행연구를 참고하여 카메라 영상에서 문자의 위치를 인식한 후 특징점을 추출한다. 이 특징점을 이용하여 PnP 문제(PnP Problem)를 이용하여 문자에 대한 카메라의 상대적인 위치를 계산해서 저장하는 것을 목표로 한다. 구체적으로 4 개 이상의 프레임에서 연속적으로 발견된 문자의 특징점을 이용하여 OCR 에서 인식하는 데 발생하는 오차를 줄인다. 이후 VIO 를 통해 이동경로를 추적하다가 새로운 문자를 인식하였을 때, 저장했던 문자와의 위치적인 정보와 특징점들의 비교를 통해 문자의 재인식 여부를 판단하여 이동경로의 기준점으로 활용하고자 한다. 이를 위해 문자가 재인식되었는지에 대한 여부를 판단하기 위한 방법을 제안한다.

†: 교신저자

본 논문의 구성은 다음과 같다. 2 절에서는 VIO 와 OCR 의 선행연구에 대해 살펴본 후, 3 절에서는 본 논문에서 문자를 재인식하였는지 판별하기 위해 제안하는 방법을 간단히 설명하고, 4 절에서는 제안한 방법의 성능을 확인하기 위해 설계한 실험을 설명한다. 마지막으로 5 절에서는 본 논문의 이후 활용방안을 제시하며 결론을 맺는다.

2. 선행연구

2.1 시각관성 측위 시스템

영상 기반으로 이루어지는 측위시스템에 대한 학술적인 연구는 매우 방대하고 활발하게 이루어지고 있다. 대표적으로 VINS-Mono[1], PTAM[2], SVO[3] 같은 선행연구가 존재한다. 이러한 연구는 크게 영상정보만을 사용하는 시각 측위와 IMU 정보를 함께 사용하는 시각-관성 측위로 나뉘지는데, 여기서는 대표적으로 VINS-Mono[1]에 대해서 주로 다룬다.

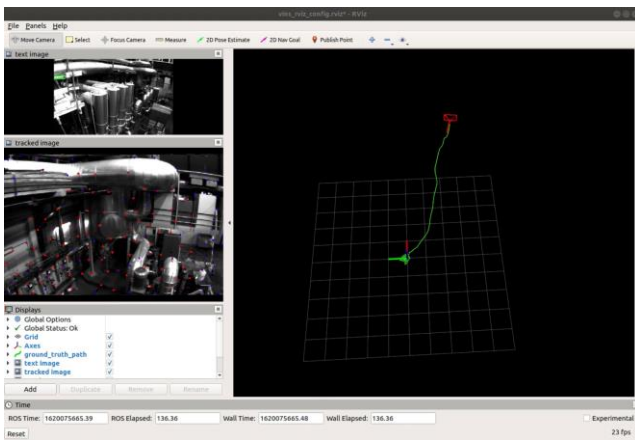


그림 1. VINS-Mono 가 동작하는 모습. 왼쪽 가운데에는 영상에서 추출된 특징점이 표시되고, 오른쪽의 좌표에서는 이동한 경로가 표시된다.

Tong Qin 외 연구진이 발표한 해당 논문에서는 ROS(Robot Operating System)을 활용하여 RGB 카메라와 IMU 의 데이터를 동기화하였고, 카메라의 이동경로를 미터법으로 변환하여 추적하였다. 그 과정에서 센서의 오차나 카메라의 급격한 움직임에 의해 발생한 경로의 오차를 보정하기 위해 Loop Closure 를 활용하였다. 이 논문의 코드는 공개되어 있어, 다양한 VIO 분야에서 기초적인 연구로 사용되고 있는데, 이벤트 카메라를 사용하는 EVIO[4]에서도 VINS-Mono 를 기반으로 연구를 진행하였다. 위의 그림 1 은 VINS-Mono 가 동작하는 것을 나타낸다.

본 논문에서는 VINS-Mono 의 구조에서 카메라의 이동 경로를 추정하는 정보와 ROS 구조를 활용하여 문자의 정보를 추출하고 가공하여 전달하는 데 사용한다. 그림 1 에서 왼쪽 위에서는

문자가 인식되었을 때 해당 영역을 표시하여 함께 출력하도록 하였다.

2.2 광학 문자 인식

기계학습의 네트워크 구조와 하드웨어의 연산속도가 발전됨에 따라 광학 문자 인식(Optical Character Recognition, 이후 OCR)분야에서도 기계학습을 사용한 연구가 진행되었다. 그 중 easyOCR[5]에서는 다양한 문자 종류에 대해 실시간으로 처리가 가능한 easyOCR 을 구현하였다. 이 연구에서는 이미지에 대해서 문자의 위치를 찾아내고, 해당 문자의 위치와 문자의 내용을 획득할 수 있었다. 또한, 이는 Python 에 API 로 등록되어 쉽게 설치하여 사용할 수 있고, ROS 에서도 동작이 가능하도록 코드가 공개되었다[6].



그림 2. 데이터셋에서 문자를 인식했을때의 모습. 인식된 문자는 초록색으로 영역을 표시하였다.

본 논문에서는 easyOCR 에서 문자의 위치를 인식하여 해당 문자 주변의 이미지를 획득한다. 획득한 이미지를 통해 문자의 특징점을 추출하고, 이 특징점을 이용해 PnP 문제(PnP Problem)을 해결하여 카메라의 상대적인 위치를 파악한다. 이 정보들을 저장하여 이후 문자가 다시 인식되었을 때 재인식되었는지 판별하기 위한 정보로 활용한다.

시각-관성 측위에서 기본적으로 사용하는 정보는 영상에서 추출한 특징점과 IMU 에서 획득한 센서의 값이다. 이 중에서 문자가 인식되는 OCR 은 영상에서의 특징점을 기반으로 추출된다. 그래서 인식되었던 문자가 재인식되었는지 여부를 확인하기 위해서 문자 영역에서의 특징점을 비교한다.

3. 제안하는 방법

1. 문자 인식: 먼저, 영상에서 문자의 위치를 인식하기

위해서 EasyOCR 을 사용한다. 이 때, 문자의 의미와 종류를 획득하지 않고, 문자의 위치만을 획득하여 사용한다. 문자의 위치를 획득할 때, 오류로 인해 문자가 아닌 경우에 문자로 인식되는 경우나, 문자의 위치가 급격하게 변하는 경우를 배제하기 위해서 연속된 영상에서 4 번 이상 비슷한 위치에서 인식되었을 때에만 문자를 제대로 인식했다고 판단하여 위치를 다음 단계로 전달한다.

2. 특징점 추출: 다음에는 획득한 문자의 영역에서 특징점을 추출한다. 이 때, 문자의 추적과 추출을 용이하게 하기 위해 Shi 와 연구진[7]이 제시한 특징점 추출 방법을 사용하고, 문자 영역 주변의 특징점도 일부 획득한다.

3. 광학 흐름 및 PnP 문제: 이후 문자를 인식했을 때의 카메라의 위치를 획득하기 위해서 광학흐름(Optical Flow)과 PnP 문제(PnP Problem)를 계산한다. 이 때, 카메라의 위치는 VINS-Mono 에서 획득한 카메라의 위치를 기준으로 하여 앞에서 구한 카메라의 위치와 문자의 상대적인 위치를 계산한다.

4. 재인식 판별 및 저장: 마지막으로, 앞에서 저장했던 문자와 비슷한 위치에서 비슷한 특징점이 추출되었을 때, 해당 문자가 재인식되었는지 판별한다. 그렇지 않은 경우, 문자가 새롭게 추출되었다고 판별하여 새로 저장한다.

4. 실험 설계 및 결과

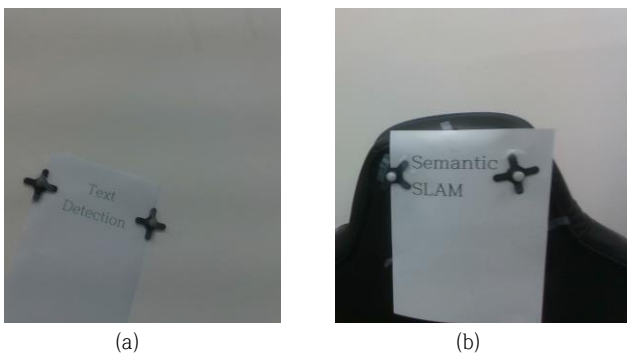


그림 3. Intel Realsense 로 촬영한 영상. 영상정보와 함께 IMU 정보를 함께 획득하였다. 또한, Optitrack 의 적외선 마커를 사용하여 글씨의 좌표를 획득하였다.

먼저, 영상에서 획득하는 문자와 영상을 촬영하는 카메라의 정확한 위치를 획득하기 위해 적외선 센서를 사용하여 위치를 획득하였다. Optitrack 사의 적외선 마커 트래커를 이용하여 정확한 텍스트의 위치를 획득하였다. 그리고 Intel 의 Realsense 를 이용하여 프레임 영상과 IMU 센서를 rosbag 형태로 저장하여 실험을 위한 영상을 촬영하였다.



그림 4. 문자가 인식되었을 때의 카메라 위치. 문자가 인식되었을 때, VINS-Mono[1]에서 구한 카메라 위치를 함께 저장한다.

이렇게 카메라의 이동경로와 문자의 정확한 위치를 획득하였고, 이들의 위치를 카메라에서 획득하는 실험을 진행하였다. 문자가 인식되었을 때에 해당 문자가 인식된 여러 프레임을 통해서 Optical Flow 와 PnP Problem 을 이용하여 카메라의 상대적인 위치를 파악하였다. 그리고 VINS-Mono 에서 구해진 카메라 위치를 함께 구하여 위치를 저장하였다.

5. 결론

본 논문에서는 시각-관성 측위 시스템에서 문자를 실시간으로 인식하고, 그 문자의 특징점과 위치를 저장하여 이후 문자가 다시 발견되었을 때 재인식되었는지 판별하기 위한 방법을 제안하였다. 그리고 이 방법을 검증하기 위해서 측위시스템에서 추정된 값과 적외선 센서를 부착한 환경에서의 값을 비교하는 실험을 진행하였다.

이 방법에서 문자의 특징점을 추출하기 위해 OCR 을 사용하였는데, OCR 에서 문자의 영역이 안정적으로 획득되지 않으면 오류가 발생할 수 있다. 이는 OCR 의 성능 개선으로 해결이 가능할 것이다. 또한, 시각-관성 측위 시스템에서 추정된 카메라의 위치를 사용하기 때문에 측위시스템에서 카메라의 위치가 잘못 추정되면 문자의 위치도 잘못 추정될 수 있는 문제점이 있다.

이렇게 획득된 문자의 위치는 재인식되었을 때 동일한 위치로서 사용하여 측위 과정 중 발생한 오차를 보정하기 위한 정보로 사용할 수 있고, 해당 문자의 위치를 기억함으로써 위치에 대한 정보를 추가로 획득하거나 저장하는 데 사용하여 현장에서 각 위치에 필요한 정보를 보완하는 데 사용할 수 있다.

감사의 말

본 연구는 한국전자통신연구원 연구운영비지원사업의 일환으로 수행되었음.

(21ZS1200, 인간 중심의 자율지능시스템 원천기술 연구)

참고문헌

- [1] T. Qin, P. Li and S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," in IEEE Transactions on Robotics, vol. 34, no. 4, pp. 1004-1020, Aug. 2018, doi: 10.1109/TRO.2018.2853729.
- [2] G. Klein and D. Murray, "Parallel Tracking and Mapping for Small AR Workspaces," 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, 2007, pp. 225-234, doi: 10.1109/ISMAR.2007.4538852.
- [3] C. Forster, M. Pizzoli and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," 2014 IEEE International Conference on Robotics and Automation (ICRA), 2014, pp. 15-22, doi: 10.1109/ICRA.2014.6906584.
- [4] A. Z. Zhu, N. Atanasov and K. Daniilidis, "Event-Based Visual Inertial Odometry," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5816-5824, doi: 10.1109/CVPR.2017.616.
- [5] <https://www.jaided.ai/easyocr/>
- [6] https://github.com/knorth55/easyocr_ros
- [7] Jianbo Shi and Tomasi, "Good features to track," 1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1994, pp. 593-600, doi: 10.1109/CVPR.1994.323794.