

논문 2024-19-09

다중 채널 동적 객체 정보 추정을 통한 특징점 기반 Visual SLAM (A New Feature-Based Visual SLAM Using Multi-Channel Dynamic Object Estimation)

박근형, 조형기*

(Geunhyeong Park, HyungGi Jo)

Abstract : An indirect visual SLAM takes raw image data and exploits geometric information such as key-points and line edges. Due to various environmental changes, SLAM performance may decrease. The main problem is caused by dynamic objects especially in highly crowded environments. In this paper, we propose a robust feature-based visual SLAM, building on ORB-SLAM, via multi-channel dynamic objects estimation. An optical flow and deep learning-based object detection algorithm each estimate different types of dynamic object information. Proposed method incorporates two dynamic object information and creates multi-channel dynamic masks. In this method, information on actually moving dynamic objects and potential dynamic objects can be obtained. Finally, dynamic objects included in the masks are removed in feature extraction part. As a results, proposed method can obtain more precise camera poses. The superiority of our ORB-SLAM was verified to compared with conventional ORB-SLAM by the experiment using KITTI odometry dataset.

Keywords : Visual SLAM, Object Segmentation, Optical Flow, Visual Feature

1. 서론

이동로봇의 자율 주행을 위해 필요한 위치인식과 지도구축을 수행하는 SLAM (Simultaneous Localization And Mapping) 기술은 임의의 새로운 공간에서 이동할 때 주변을 탐색하면서 이동한 공간의 지도를 작성하고 자신의 위치를 파악하는 기술이다. 카메라 센서를 통해 외부 환경을 감지하는 Visual SLAM은 작동하는 방식에 따라 단안 용 카메라 (Monocular), 양안 용 카메라 (Stereo), 깊이 카메라 (RGB-D)등을 활용한다.

ORB SLAM [1]은 Visual SLAM의 한 종류로서, ORB (Oriented FAST and Rotated BRIEF) [2] 특징점 검출 방식을 사용하여 주변 환경을 인식하는 특징점 기반의 SLAM이다. ORB SLAM은 Tracking, Local Mapping, Loop Closing의 3가지 독립적인 쓰레드로 구성되어 있다.

ORB SLAM의 Tracking 쓰레드에서는 입력 영상으로부터 특징점을 추출하여 Keyframe을 생성하고 기존의 지도 정보 (3D)와 함께 카메라의 위치를 추정하는 과정이 이루어진다. Loop Closing 쓰레드에서는 이전에 방문한 적 있는 공간

인지 확인하고 방문했던 공간이라고 판단되면 이전에 생성한 지도와 통합하는 과정이 이루어진다. Place Recognition을 통해 이전에 방문했던 공간인지를 확인하며, 이때 Bag Of Words (BoW) [3] 기법이 활용된다. Recognition Database를 후보군으로 현재의 keyframe 에서 나온 Visual Vocabulary 를 통해 표현된 데이터를 후보군 안의 모든 데이터와 비교하며, 같은 장소라고 판단할 경우 지도를 통합한다.

이처럼 영상의 특징점들은 카메라의 위치추정뿐 아니라 지도 통합 단계에서 중요한 역할을 하며 같은 장소일지라도 영상의 특징점이 달라진다면 시스템의 성능이 저하될 수 있다. 특히, 시간에 따라 변화하는 동적 객체들은 같은 장소에서의 특징점들을 변화시키기에 성능 저하의 주된 원인으로 여겨지며, 이를 제거하려는 다양한 연구가 진행되었다. 일부 전통적인 연구에서는 움직임 검출을 위하여 RANdom SAmple Consensus (RANSAC) [4] 알고리즘 혹은 확률론적인 방법 [5]이 사용되었다. 논문 [6]은 RANSAC을 통해 기하학적 제약에 따른 영상 프레임 사이의 배경 궤적과 전경 궤적을 구분하여 객체의 움직임을 판단하였다. 하지만 동적 객체가 영상의 대부분을 차지하거나 객체의 이동 속도가 빠르면 신뢰성 높은 정보들을 얻기 힘들다는 단점이 있다. 최근에는 딥러닝 네트워크 중 하나인 합성곱 신경망을 활용한 방법들이 제시되고 있다. 논문 [7-9] 은 Mask R-CNN [10], SSD [11], YOLACT [12] 과 같은 딥러닝 네트워크를 사용하여 동적 객체를 검출하였다.

동적 객체는 Visual SLAM의 동작을 방해하는 주된 요소이다. 동적 객체는 low, high dynamic으로 구분할 수 있다. high dynamic 객체는 센서에서 관측되는 동안 지속적으로

*Corresponding Author (hygijo@jbnu.ac.kr)

Received: Oct. 16, 2023, Revised: Nov. 20, 2023, Accepted: Dec. 21, 2023.
G. Park: Jeonbuk National University (B.S. Student)

H. Jo: Jeonbuk National University (Assistant Professor)

* This work was supported in part by the Materials/Parts Technology Development Program (20023305, Development of intelligent delivery robot with Cloud-Edge AI for last mile delivery between nearby multi-story buildings) funded By the Ministry of Trade, Industry & Energy (MOTIE, Korea), and in part by the "Regional Innovation Strategy (RIS)" through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (MOE) (2023RIS-008).

움직이는 반면, low dynamic 객체는 관측이 되지 않고 있을 때 다른 장소로 움직이는 것을 의미한다 [13-15]. 또한, 컴퓨터비전의 semantic segmentation을 활용하여 객체의 종류에 대한 정보를 활용하는 연구 또한 진행되었다 [16, 17]. 특히, dynamic 환경에 대응하기 위한 optical flow와 semantic Visual SLAM을 결합한 SLAM 시스템이 연구되었다 [17]. 본 논문에서는 high dynamic 객체 중 잠재적으로 움직일 수 있는 객체에 초점을 맞추어 이에 맞는 동적 객체 masking의 image-pyramid를 생성한다.

본 연구에서는 단안 카메라 (Monocular)에서 전통적인 움직임 검출 방식인 광학 흐름 (Optical-Flow) [18]와 딥러닝 기법인 Mask R-CNN을 동시에 적용하여 다중 동적 객체 정보를 더욱 효과적으로 추정하는 ORB-SLAM 알고리즘을 제시한다. Tracking 쓰레드의 전처리 부분에서 광학 흐름을 통해 2차원 영상의 픽셀 단위 움직임을 검출하고, Mask R-CNN 알고리즘을 적용하여 동적 객체를 마스킹해 준다. 최종적으로 이러한 마스킹 행렬은 ORB 특징점 추출 과정에 적용된다. 본 논문의 구성은 다음과 같다. 2절에서는 다중 동적 객체의 정보를 추정하여 동적 객체 마스킹 행렬을 생성한다. 3절에서는 동적 객체 마스킹 행렬을 이용하여 정적인 환경에서만 ORB 특징점을 추출하는 과정을 기술한다. 4절에서는 benchmark dataset을 활용하여 기존 ORB-SLAM 알고리즘과 본 연구의 성능을 비교하고 결론과 향후 연구 방향을 제시한다.

II. 다중 동적 객체 정보 추정

다중 채널 동적 객체 (Multi-channel Dynamic Objects) 정보를 추정하는 전반적인 과정은 그림 1과 같다. 첫째, 광학 흐름 알고리즘을 통해 추정된 픽셀 별 움직임 벡터와 연속된 프레임 사이의 카메라 움직임을 비교하여 동적 픽셀을 정의한다. 둘째, Mask R-CNN 알고리즘을 통해 예비 동적 객체의 마스킹 정보를 검출한다. 셋째, 동적 픽셀과 예비 동적 객체 마스킹 정보의 IOU를 계산하여 최종적인 동적 객체를 판별하고, ORB 특징점 추출에 활용할 동적 객체 마스킹 행렬을 생성한다.

1. 픽셀 움직임 판별

이 절에서는 밀집 광학 흐름 알고리즘을 적용하여 프레임의 픽셀 단위 움직임 벡터를 검출한다. 이후 연속된 프레임 사이의 카메라 움직임을 계산한다 (그림 2). 광학 흐름 움직임과 카메라 움직임 사이의 차이를 계산하여 해당 픽셀의 움직임 여부를 이진 행렬로 표현한다.

1.1 광학 흐름

광학 흐름 (Optical-Flow)이란 연속된 영상 프레임 속 물체의 움직임 패턴을 말한다. 광학 흐름은 다음 두 가지 가정을 가진다. 1) 연속된 프레임 사이에서 움직이는 물체의 픽셀 강도는 변함이 없다. 2) 이웃하는 픽셀은 비슷한 움직임을 가진다. 광학 흐름은 일부 픽셀만 계산하는 희소 (sparse)

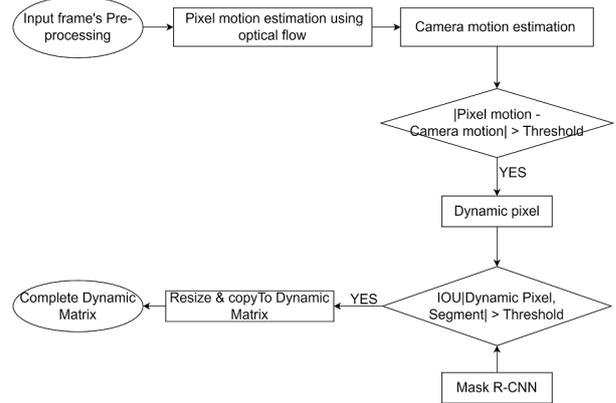


그림 1. 입력 영상의 전처리 흐름도.
Fig. 1. Input frame's Pre-processing flow chart

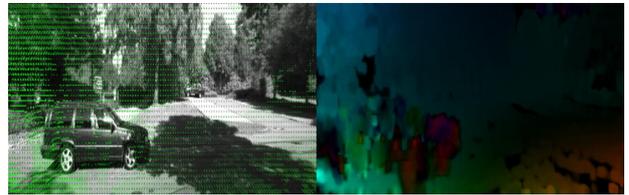


그림 2. 밀집 광학 흐름 예시: RGB 공간 (좌), HSV 공간 (우).
Fig. 2. Dense Optical Flow: RGB (left), HSV (right).

광학 흐름과 전체 픽셀을 모두 계산하는 밀집 (dense) 광학 흐름이 있으며 희소 광학 흐름에는 Lucas-Kanade [19] 알고리즘이, 밀집 광학 흐름에는 Gunner Farneback [20] 알고리즘이 사용된다. 본 연구에서는 픽셀 단위의 정확한 움직임 판별이 요구되므로 모든 픽셀값에 대해 광학 흐름을 계산하는 Gunner Farneback 알고리즘을 적용하였다.

해당 알고리즘을 적용하면 현재 프레임의 픽셀을 기준으로 이전 프레임 픽셀로의 x, y 방향 모션벡터 u, v 가 계산된다. 수식 (1)과 같이 현재 픽셀에 u, v 를 더해줄 경우, 전 프레임에서 해당 픽셀의 위치를 예측할 수 있다.

$$p_{t-1}(x, y) = p_t(x + u, y + v). \quad (1)$$

광학 흐름을 통해 예측된 이전 프레임의 2D 픽셀 좌표 p_{t-1} 를 3D 공간좌표 P_{t-1}^F 로 표현해주기 위해 다음 과정을 진행하였다. 1) 픽셀 좌표를 카메라 초점과의 거리가 1인 정규좌표로 변환한다. 2) 실제 물체와의 거리 Z 를 반영하여 3차원 공간좌표로 표현한다. 픽셀 좌표 $p(x, y)$ 와 정규좌표 $p'(x', y')$ 는 수식 (2-1)의 행렬 관계에 있으며, 따라서 정규좌표 p' 는 수식 (2-2)와 같이 계산된다.

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix}, \quad (2-1)$$

$$p'(x', y') = p((x - c_x)/f_x, (y - c_y)/f_y). \quad (2-2)$$

f_x, f_y 는 초점 거리 (focal length), c_x, c_y 는 주점 (principal points)이다. 이 후 실제 물체와의 거리 Z 를 정규 좌표에 곱해준다면 현재 프레임에서의 픽셀 좌표 $p_t(x, y)$ 를 기준으로 광학 흐름을 통해 예측된 이전 프레임에서의 3D 좌표 $P_{t-1}^F(X, Y, Z)$ 로의 표현이 가능하며 최종적인 계산은 수식 (3)과 같이 이루어진다.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} (x+u-c_x)/f_x \times Z \\ (y+v-c_y)/f_y \times Z \\ Z \end{bmatrix}. \quad (3)$$

1.2 카메라 자세 (Pose) 추정

ORB-SLAM에서는 Tracking 과정을 통해 월드 좌표계의 카메라 자세 (Pose)가 계산된다. 각 프레임의 카메라 자세 (Pose) 행렬은 3x3의 회전 행렬과 3x1의 이동 행렬로 이루어져 있으며, 동차 좌표계에 따라 최종 4x4 행렬로 표현된다. 해당 프레임에서의 카메라 자세 (Pose) 행렬을 M_t , 이전 프레임에서의 카메라 자세 (Pose) 행렬을 M_{t-1} 이라고 설정할 경우, 이전 프레임을 기준으로 현재 프레임과의 카메라 자세 (Pose) 관계 M_r 은 수식 (4)를 통해 구해지며,

$$M_r = \begin{bmatrix} R_r & T_r \\ 0 & 1 \end{bmatrix} \text{으로 표현할 수 있다.}$$

$$M_r = \begin{bmatrix} R_t & T_t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} (R_{t-1})^T & -(R_{t-1})^T T_{t-1} \\ 0 & 1 \end{bmatrix}. \quad (4)$$

현재 프레임에서의 픽셀 좌표 $p_t(x, y)$ 를 수식 (5)를 통해 3D 공간 $P_t(X, Y, Z)$ 로 변환한다. 이후 프레임 사이의 카메라 자세 (Pose) 관계 M_r 을 적용한다면 최종적으로 이전 프레임의 3D 카메라 좌표 $P_{t-1}^M(X, Y, Z)$ 를 계산할 수 있다. 해당 계산은 수식 (6)에 따라 이루어진다.

$$P_t = ((x-c_x)Z/f_x, (y-c_y)Z/f_y, Z), \quad (5)$$

$$P_{t-1}^M = P_t(R_r)^T - (R_r)^T T_r. \quad (6)$$

1.3 움직임 여부를 표현하는 이진 행렬

정적인 환경에서 밀집 광학 흐름을 통해 예측된 이전 프레임에서의 3D 좌표 P_{t-1}^F 와 카메라 움직임을 통해 예측된 이전 프레임에서의 3D 좌표 P_{t-1}^M 은 같아야 한다. 하지만 밀집 광학 흐름 계산에서의 노이즈로 야기된 부정확성 때문에, 적절한 기준값 Γ 을 설정하였다. P_{t-1}^F 과 P_{t-1}^M 의 차이가 Γ 을 초과할 경우 해당 픽셀을 움직임이 있는 픽셀로 간주하여 1의 값을, 기준값 이하의 픽셀들은 0의 값을 부과하여 최종적으로 움직임 여부를 표현하는 이진 행렬 $\hat{p}_t(x, y)$ 을 수식 (7)에 따라 생성한다. 최종적인 이진 행렬 $\hat{p}_t(x, y)$ 은 그림 3과 같다.

$$\hat{p}_t(x, y) = |P_{t-1}^F - P_{t-1}^M| > \Gamma ? 1 : 0. \quad (7)$$



그림 3. 광학 흐름을 통한 움직임 이진 행렬
Fig. 3. Flow matrix via optical flow

2. 동적 객체 마스킹 행렬 생성

Mask R-CNN 알고리즘을 적용하여 해당 프레임에서의 예비 동적 객체 Segmentation을 진행한다. 예비 동적 객체의 Bounding Box 좌표를 활용하여, 2.1절에서 생성한 움직임 이진 행렬의 ROI를 추출하고, 예비 동적 객체 Segment와의 IOU를 측정하여 최종적인 동적 객체를 검출한다. ORB 특징점 검출에서 사용하기 위해 다중 동적 객체가 마스킹 된 이진 행렬을 생성한다.

2.1 Mask R-CNN을 통한 예비 동적 객체 선별

Instance Segmentation은 이미지 내에 존재하는 모든 객체를 탐지하는 동시에 각각의 경우를 정확하게 픽셀 단위로 분류하는 과정이다. Mask R-CNN은 Instance Segmentation의 대표적인 알고리즘으로, Faster R-CNN [21]의 브랜치와 평행하게 Segmentation 마스크를 예측하는 마스크 브랜치를 추가한 구조이다. 해당 알고리즘의 적용을 위하여 Tensorflow에서 MS COCO dataset [22]으로 선행 학습된 모델 (Frozen graph)과 환경파일을 활용하여 OpenCV inference 모델을 생성하였다. MS COCO dataset은 80개의 객체 범주들로 구성되어 있으며, 해당 범주 중 자동차, 자전거 등의 동적 객체 범주를 따로 정의해 주어, 예비 동적 객체에 대한 Segmentation을 진행하였다.

2.2 ROI의 IOU 계산

Inference 모델은 검출된 객체의 Bounding Box 정보와 마스킹 정보를 함께 반환해준다. 각 객체의 최초 마스크는 14x14 크기의 특징 맵 형태이며, 후처리 과정을 통해 Bounding Box의 크기에 맞추어 원본 크기로 rescale 해준다. 이후 각 픽셀값이 기준값 이상일 경우 객체가 있음을 의미하는 1을 할당하고, 나머지는 0을 할당하여 마스크 이진 행렬을 생성한다 (그림 4). 해당 이진 행렬과 2.1에서 생성한 움직임 이진 행렬의 비교를 위하여 검출된 객체의 Bounding Box 좌표에 맞추어 움직임 이진 행렬의 ROI를 추출한다 (그림 4). 이후 수식 (8)에 맞추어 두 이진 행렬의 IOU를 계산한다. 교집합은 AND 연산, 합집합은 OR 연산을 통해 도출된다. IOU가 기준값 이상일 경우, 해당 예비 동적 객체가 최종 동적 객체를 결정한다.

$$IOU = \frac{(segment \cap \hat{p}_t)}{(segment \cup \hat{p}_t)}. \quad (8)$$

2.3 최종 동적 객체 마스킹 행렬 생성

최종적으로 결정된 동적 객체는 ORB 특징점 검출 시 제외되어야 하는 부분이다. 그러므로 최종 마스킹 행렬에서

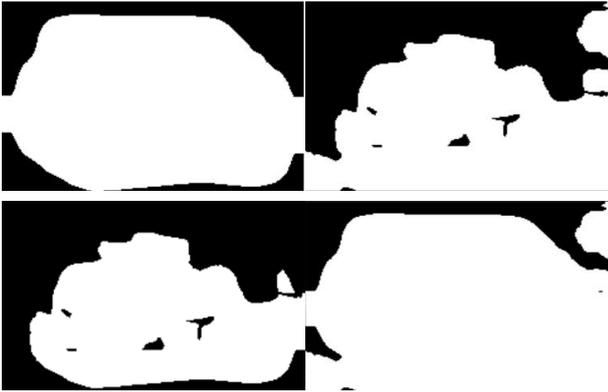


그림 4. 세그멘테이션 결과, 움직임 이진 행렬 ROI, AND 연산 결과, OR 연산 결과
 Fig. 4. Segmentation result (up left) and ROI of flow matrix (up right), AND operator result (down left), OR operator result

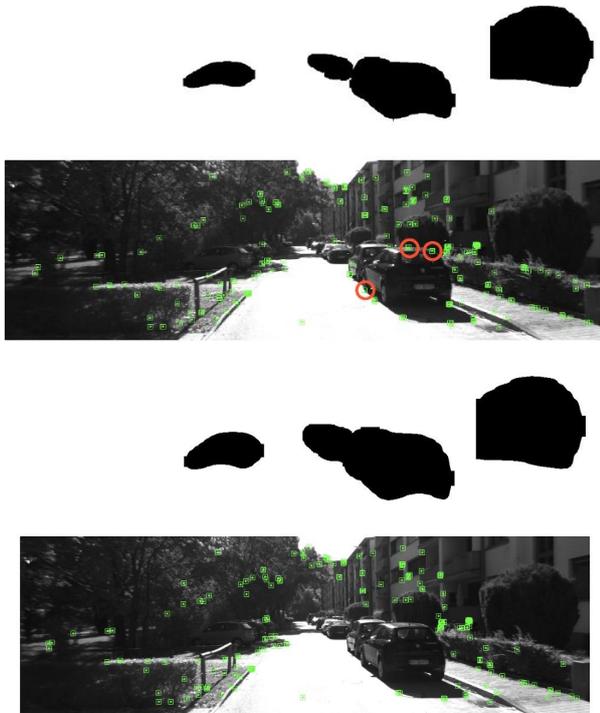


그림 5. 마스크 크기 조정: 기본 크기 (위), 크기 조정 (아래)
 Fig. 5. Mask size scaling: Up is ordinal size mask, Down is scale up mask.

각 동적 객체들은 픽셀값이 0으로 표현되어야 한다. 입력 프레임과 크기가 같고, 각 픽셀이 1로 초기화된 Mat 객체를 생성한 뒤, 해당 객체의 Bounding Box 위치에 0으로 반전된 마스크 행렬을 복사해 준다. 또한, 그림 5와 같이 마스크 크기를 미세하게 조정하여, 객체의 가장자리에서 특징점이 검출되는 문제를 해결해 주었다.



그림 6. 마스크 행렬의 이미지 피라미드 예시
 Fig. 6. Image-Pyramid of Masking matrix: technique for analyzing images across scales

III. ORB 특징점 추출 방법

1. 동적 객체 마스크 행렬의 Image-Pyramid 과정

영상에서 특징점을 검출하는 데 사용하는 ORB (Oriented FAST and rotated BRIEF) 알고리즘은 FAST (Features from Accelerated Segment Test) [23] 알고리즘과 BRIEF (Binary Robust Independent Elementary Features) [24] 알고리즘이 결합한 방식으로, 연산속도가 빠르다는 장점이 있다.

하지만 FAST 알고리즘은 Multi-Scale 기능을 지원하지 않기 때문에, ORB 알고리즘은 Image-Pyramid 기법을 활용하여 이러한 문제를 해결한다. Image-Pyramid를 생성한 뒤, FAST 알고리즘을 적용하여 각각의 scale 이미지별 특징점을 검출하기 때문에 ORB는 부분 scale 변화에 불변한 특징을 가진다. 2절에서 최종적으로 생성한 동적 객체 마스크 행렬의 Image-Pyramid를 생성해주어, ORB 알고리즘에 적용해 주었다. 그림 6는 동적 객체 마스크 행렬의 Image-Pyramid 과정 예시이다.

2. 동적 객체를 제외한 특징점 추출

동적 객체 마스크 행렬을 적용하여 ORB 특징점을 추출한 결과는 다음과 같다. 기존의 알고리즘 그림 7과 달리, Mask R-CNN만을 적용해 준다면, 그림 8과 같이 모든 차량을 마스크한다. 하지만 추가로 광학 흐름을 적용하였을 때 그림 9과 같이 움직이는 차량만을 마스크 행렬에 추가하여 정적인 공간에서만 특징점 추출이 이루어짐을 확인할 수 있다.

IV. 실험 결과

1. 데이터셋 및 성능지표

연구에 대한 평가를 위하여 KITTI odometry dataset [25]을 사용하였다. 해당 dataset은 총 22개의 stereo sequence로 이루어져 있으며, 성능평가는 ground truth 정보가 제공된 11개의 sequence (00-10)에 대해 진행하였다. 해당 sequence에는 그림 9와 같이 성능평가 지표로써 ATE (Absolute Trajectory Error) [26]을 사용했다. ATE (Absolute Trajectory Error)란 Ground truth로 주어진 trajectory와 측정된 trajectory 차이의 절댓값을 말하며, 수식 (9)에 따라 전체 ATE에 대한 RMSE (Root Mean Squared Error)를 계산하여 최종적인 평가지표로 사용하였다.



그림 7. 기존 ORB-SLAM의 특징점 추출 결과
Fig. 7. ORB features from Conventional ORB-SLAM



그림 8. 제안된 연구의 특징점 추출 결과
Fig. 8. ORB features from proposed method : only Mask R-CNN



그림 9. 제안된 연구의 특징점 추출 결과
Fig. 9. ORB features from proposed method : Optical Flow + Mask R-CNN

$$ATE_{R\text{MSE}}(\hat{X}, X) = \sqrt{\frac{1}{n} \sum_{i=1}^n |trans(\hat{X}_i) - trans(X_i)|} \quad (9)$$

$\hat{X} = \hat{X}_1 \dots \hat{X}_n$: trajectory estimate
 X : groundtruth

11개의 sequence 중 06, 08에 대한 Ground truth 경로와 기존 ORB SLAM의 결과를 그림 10과 11에서 확인할 수 있다.

2. 결과 Trajectory

odometry 정보를 시각화하여 기존 알고리즘과의 차이를 비교하였다. 그림 12와 그림 13에서 ground truth 경로와 기존 알고리즘, Mask R-CNN만 적용했을 때, 광학 흐름과 Mask R-CNN을 함께 적용했을 때의 차이를 확인할 수 있다. 정량적 비교를 위해 표 1에서 KITTI의 모든 sequence

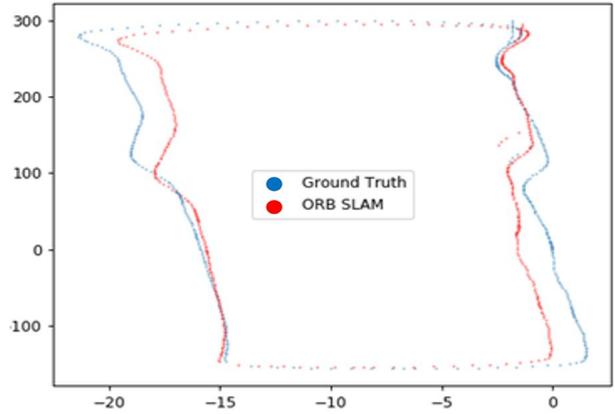


그림 10. Ground truth와 기존 ORB SLAM 결과 경로
Fig. 10. Trajectory of Ground Truth and ORB SLAM results for KITTI 06

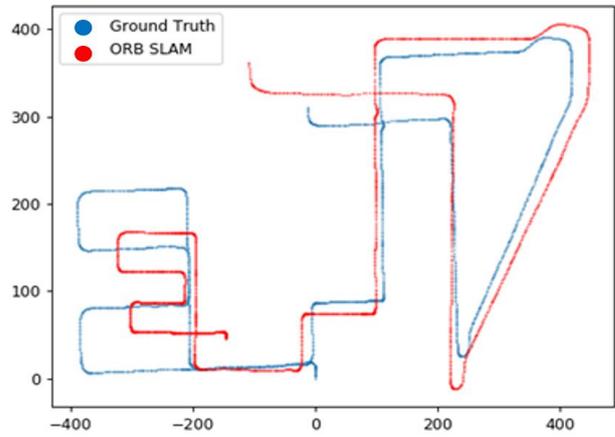


그림 11. Ground truth와 기존 ORB SLAM 결과 경로
Fig. 11. Trajectory of Ground Truth and ORB SLAM results for KITTI 08

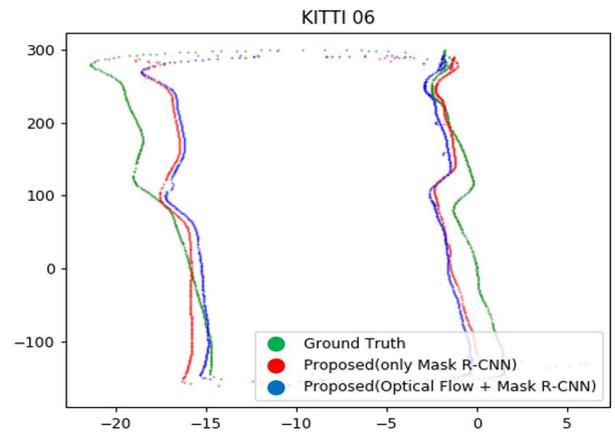


그림 12. 제안된 방법과 Ground Truth 경로 결과 KITTI Odometry 데이터셋 (KITTI-06)

Fig. 12. Comparisons of resultant trajectory for proposed method : Optical Flow + Mask R-CNN

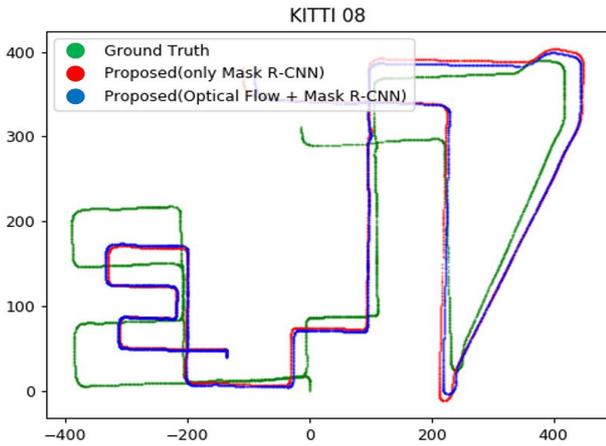


그림 13. 제안된 방법과 Ground Truth 경로 결과 KITTI Odometry 데이터셋 (KITTI-06)

Fig. 13. Comparisons of resultant trajectory for proposed method : Optical Flow + Mask R-CNN

표 1. ORB-SLAM의 절대 궤적 오차 정량적 비교

Table 1. Absolute Trajectory RMSE (m) for ORB-SLAM

	ORB-SLAM	+ Mask R-CNN	+ Optical Flow, Mask R-CNN
KITTI 00	6.2614	5.9938	7.9187
KITTI 01	417.4684	308.6948	372.2654
KITTI 02	25.5549	23.7065	25.7413
KITTI 03	1.2842	2.5587	0.9333
KITTI 04	0.4641	0.8635	1.0653
KITTI 05	4.9076	5.9362	3.2522
KITTI 06	15.8781	19.9489	15.5701
KITTI 07	2.8188	2.5120	4.4582
KITTI 08	48.9107	43.8289	43.8019
KITTI 09	44.3646	28.0709	32.3063
KITTI 10	7.2425	7.1274	8.4215

에 대해 기존 방법과 본 논문에서 제안한 방법의 ATE를 비교하였다.

광학 흐름과 Mask R-CNN을 적용한 본 연구와 기존 알고리즘의 성능평가 결과, 총 11개의 dataset 중 KITTI 04를 제외한 데이터셋에 대해서 RMSE가 감소하는 성능 향상이 있었다. 04번 시퀀스에 대해서 RMSE가 증가하는 성능 하락이 발생하였다. 대부분의 시퀀스에서 본 연구에서 제안한 방식으로 Mask R-CNN을 적용한 경우에 성능 향상을 얻을 수 있었다. 또한, Optical Flow를 사용하는 경우에서도 성능 하락이 발생하는 경우가 존재하는데, 총 6개의 시퀀스에서는 Mask R-CNN만을 사용하는 경우 더욱 좋은 성능을 보여주고 있다. 그 이유는 외부 환경의 불규칙한 명암 변화로 발생하는 광학 흐름 오차 및 영상의 깊이 정보 오차 때문에 움직임 이진 행렬을 정확하게 생성하지 못함으로써 실질적인 동적 객체를 판별하지 못했기 때문이라고 판단된다. 또한, 실제 움직이는 차량과 주차되어 정지해 있는 차량의 비율에 따라 성능이 달라진다고 볼 수 있다. 주차된 차량의

비율이 높은 경우, 해당 부분에서 얻을 수 있는 특징점의 수가 현저히 감소함에 따라 오히려 SLAM이 불안정하게 자세를 추정하게 된다.

그러나 KITTI 06, 08에서 보여주는 결과와 같이 실제 움직임이 있는 객체를 효과적으로 제거해준다면 기존 알고리즘뿐만 아니라 모든 동적 객체 범주를 제거해 줬을 때 보다 뛰어난 결과를 도출할 수 있음을 확인하였다.

V. 결론

본 논문에서는 다중 객체 정보 추정을 통해 외부 환경에서 강건하게 작동하는 특징점 기반의 Visual SLAM을 기술하였다. 정적인 환경에서만 특징점 추출이 이루어지기 위해선, 동적 객체를 마스킹한 이진 행렬이 필요하였고, 이를 위해 밀집 광학 흐름인 Gunner Farneback 알고리즘과 Instance Segmentation을 위한 Mask R-CNN 알고리즘을 활용했다. 연속된 프레임 사이에 밀집 광학 흐름과 카메라의 움직임을 비교하여 움직임 이진 행렬을 생성하였고, Mask R-CNN을 통해 검출된 예비 동적 객체 segment와의 IOU 계산을 통해 최종적인 동적 객체 마스킹 행렬을 구성하였다. 동적 객체 마스킹 행렬은 Image-Pyramid를 생성하여 ORB 특징점 검출 과정에 적용되었고, 최종적으로 영상의 정적인 환경에서 특징점을 검출할 수 있었다.

연구에 대한 평가를 위해 KITTI odometry dataset 중 ground truth가 제공된 11개의 sequence에 대해서 기존 ORB-SLAM (Mono)과 본 연구 결과가 적용된 ORB-SLAM을 비교하였다. Visual SLAM 성능측정의 대표적인 평가지표인 ATE (Absolute Trajectory Error)를 측정하였고, 대부분의 sequence에 대해 ATE의 RMSE가 감소하는 결과를 얻었다. 하지만 광학 흐름 오차, 영상의 깊이 정보 오차 등으로 정확한 움직임 이진 행렬을 생성하지 못할 때는 기존 알고리즘보다 성능이 하락하는 결과를 초래했다. 그러므로 불규칙한 조명 변화 등에 강건하게 작동하는 광학 흐름을 적용하여 정확하게 동적 픽셀을 구분하는 추가 연구가 우선시 되어야 한다.

또한 마스킹을 제외한 부분에서 ORB 특징점을 추출한다는 것은 즉 추출되는 특징점의 절대량이 감소한다는 것을 의미하고, 이는 오히려 위치추정의 성능 하락을 일으킬 수 있다. 그러므로 마스킹 때 특징점 추출이 안 되는 공간을 처리해야하는 추후 연구가 필요하다.

References

- [1] R. Mur-Artal, J. M. M. Montiel, J. D. Tardos, "ORB-SLAM: a Versatile and Accurate Monocular SLAM System," *IEEE Trans. Robot.*, Vol. 31, No. 5, pp. 1147-1163, 2015.
- [2] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, "ORB: An Efficient Alternative to SIFT or SURF," In *Proc. International Conference on Computer Vision (ICCV)*, pp. 2564-2571, 2011.

- [3] D. Galvez-Lopez, J. D. Tardos, "Bags of Binary Words for Fast Place Recognition in Image Sequences," *IEEE Trans. Robot.*, Vol. 28, No.5, pp. 1188-1197, 2012.
- [4] M. A. Fischler, R. C. Bolles, "Random Sample Consensus: a Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Communications of the ACM*, Vol. 24, No. 6, pp. 381-395, 1981.
- [5] Y. Sheikh, M. Shah, "Bayesian Modeling of Dynamic Scenes for Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 27, No. 11, pp. 1778-1792, 2005.
- [6] Y. Sheikh, O. Javed, T. Kanade, "Background Subtraction for Freely Moving Cameras," In *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 1219-1225, 2009.
- [7] B. Bescos, J. M. Fácil, J. Civera, J. Neira, "DynaSLAM: Tracking, Mapping, and inpainting in Dynamic Scenes," *IEEE Robot. Autom. Lett.*, Vol. 3, No. 4, pp. 4076-4083, 2018.
- [8] D. Lai, C. Li, B. He, "YO-SLAM: A Robust Visual SLAM Towards Dynamic Environments," In *Proc. IEEE International Conference on Communications, Information System and Computer Engineering (CISCE)*, pp. 720-725, 2021.
- [9] F. Zhong, S. Wang, Z. Zhang, Y. Wang, "Detect-SLAM: Making Object Detection and SLAM Mutually Beneficial," In *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, pp. 1001-1010, 2018.
- [10] K. He, G. Gkioxari, P. Dollár, R. Girshick, "Mask R-CNN," In *Proc. International Conference on Computer Vision (ICCV)*, pp. 2961-2969, 2017.
- [11] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, A. C. Berg, "SSD: Single Shot Multibox Detector," In *Proc. Eur. Conf. Comput. Vis.*, pp. 21-37, 2016.
- [12] D. Bolya, C. Zhou, F. Xiao, Y. J. Lee, "Yolact: Real-time Instance Segmentation," In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 9157-9166, 2019.
- [13] A. Walcott-Bryant, M. Kaess, H. Johannsson, J. J. Leonard, "Dynamic Pose Graph Slam: Long-term Mapping in Low Dynamic Environments," In *Proc. IEEE Int. Conf. Intell. Robots Syst.*, pp. 1871-1878, 2012.
- [14] D. Lee, and M. Hyun, "Solution to the SLAM Problem in Low Dynamic Environments Using a Pose Graph and an RGB-D Sensor," *Sensors*, Vol. 14, No. 7, pp. 12467-12496, 2014.
- [15] S. Song, H. Lim, S. Jung, H. Myung, "G2P-SLAM: Generalized RGB-D SLAM Framework for Mobile Robots in Low-dynamic Environments," *IEEE Access*, Vol. 10, pp. 21370-21383, 2022.
- [16] C. Yu, Z. Liu, X. J. Liu, F. Xie, Y. Yang, Q. Wei, Q. Fei, "DS-SLAM: A Semantic Visual SLAM Towards Dynamic Environments," In *Proc. IEEE Int. Conf. Intell. Robots Syst. (IROS)*, pp. 1168-1174, 2018.
- [17] L. Cui, C. Ma, "SOF-SLAM: A Semantic Visual SLAM for Dynamic Environments," *IEEE Access*, Vol. 7, pp. 166528-166539, 2019.
- [18] B. K. P. Horn, B. G. Schunck, "Determining Optical Flow," *Artificial Intelligence*, Vol. 17, pp. 185-203, 1981.
- [19] B. D. Lucas, T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," In: *IJCAI'81: 7th International Joint Conference on Artificial Intelligence*, pp. 674-679, 1981.
- [20] G. Farneback, "Two-frame Motion Estimation Based on Polynomial Expansion," In *Proc. Scandinavian Conference on Image Analysis*. Springer, Berlin, Heidelberg, pp. 363-370, 2003.
- [21] S. Ren, K. He, R. Girshick, J. Sun., "Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks," In *Advances in Neural Information Processing Systems*, 2015.
- [22] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, "Microsoft coco: Common Objects in Context," In *Proc. Eur. Conf. Comput. Vis.*, pp. 740-755, 2014.
- [23] E. Rosten, T. Drummond, "Machine Learning for High-speed Corner Detection," In *Proc. Eur. Conf. Comput. Vis.*, pp. 430-443, 2006.
- [24] M. Calonder, V. Lepetit, C. Strecha, P. Fua, "Brief: Binary Robust Independent Elementary Features," In *Proc. Eur. Conf. Comput. Vis.*, pp. 778-792, 2010.
- [25] A. Geiger, P. Lenz, R. Urtasun, "Are we Ready for Autonomous Driving? the Kitti Vision Benchmark Suite," In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 3354-3361, 2012.
- [26] J. Sturm, N. Engelhard, F. Endres, W. Burgard, D. Cremers, "A Benchmark for the Evaluation of RGB-D SLAM Systems," In *Proc. IEEE Int. Conf. Intell. Robots Syst.*, pp. 573-580, 2012.

Geunhyeong Park (박근형)



2023 Electronic Engineering from Jeonbuk National University (B.S.)

2023~LG Electronics Vehicle component Solutions Company Cluster Unit

Field of Interests: Computer Vision, Visual SLAM, Robotics

Email: park11127@naver.com

HyungGi Jo (조형기)



2012 Electrical and Electronic Engineering from Yonsei University (B.S.)

2020 Electrical and Electronic Engineering from Yonsei University (Ph.D.)

2021~Div. of Electronic Engineering from Jeonbuk National University (Assistant Professor)

Field of Interests: SLAM, Robot Perception, Autonomous Navigation, Spatial AI

Email: hygijo@jbnu.ac.kr