Journal of Korea Robotics Society (2019) 14(2):114-121
https://doi.org/10.7746/jkros.2019.14.2.114 ISSN: 1975-6291 / eISSN: 2287-3961

딥러닝 기반 영상 주행기록계와 단안 깊이 추정 및 기술을 위한 벤치마크

Benchmark for Deep Learning based Visual Odometry and Monocular Depth Estimation

최 혁 두†
Hyukdoo Choi†

Abstract: This paper presents a new benchmark system for visual odometry (VO) and monocular depth estimation (MDE). As deep learning has become a key technology in computer vision, many researchers are trying to apply deep learning to VO and MDE. Just a couple of years ago, they were independently studied in a supervised way, but now they are coupled and trained together in an unsupervised way. However, before designing fancy models and losses, we have to customize datasets to use them for training and testing. After training, the model has to be compared with the existing models, which is also a huge burden. The benchmark provides input dataset ready-to-use for VO and MDE research in 'tfrecords' format and output dataset that includes model checkpoints and inference results of the existing models. It also provides various tools for data formatting, training, and evaluation. In the experiments, the exsiting models were evaluated to verify their performances presented in the corresponding papers and we found that the evaluation result is inferior to the presented performances.

Keywords: Visual Odomtery, Monocualr Depth Estimation, Deep Learning

1. 서 론

로봇 등의 지능형 이동체는 자율주행을 위해 자신의 위치나 주변 환경의 형태 등의 기하학적인 정보를 필요로 하게 된다. 우선 자신의 위치와 방향, 즉 자세, 혹은 그 변화량을 인식해야 하고 주변 환경의 2차원 혹은 3차원 형태를 알 수 있어야 주행의 방향과 속도 등을 결정할 수 있다. 본 논문은 이와 관련된 연구를 촉진하는 데이터와 도구 등을 제안한다.

딥러닝은 높은 유연성을 갖는 학습 알고리즘으로 다양한 분야에 적용되고 있고[1] 충분한 양의 학습 데이터만 주어진다면 좋은 성능을 내곤 한다. 딥러닝을 영상에 적용할 때 특히 효과적인데 영상 분류와 영상 속 객체 검출, 영역 분할 등에서 큰 발전을 이루었다[2]. 최근에는 이러한 영상의 의미 분석에만 그치지 않고 영상에서 기하학적인 정보를 추론하는 연구들이 발표되고 있다[3-7].

이동하는 카메라의 영상으로부터 카메라의 자세를 추적하는 기술은 10년이상 꾸준히 연구되어왔다[8,9]. 영상기반 자세 추정 방식은 크게 영상 주행기록계(Visual Odometry, VO)와 영상기반 위치인식 및 지도구축(Visual SLAM)이 있다. VO는 이동하는 카메라의 상대적인 자세 변화량을 계산하여 오차가 누적되는 반면 Visual SLAM은 VO에서 전역지도구축과 전역 경로최적화 기능을 추가하여 오차 누적을 최소화한 기술이다. 그러나 Visual SLAM의 성능도 사실상 VO의 성능에 의해 좌우되기 때문에 정확한 VO 기술의 확보가 지능형 이동체 개발에 중요한 요소라고 할 수 있다.

VO는 컴퓨터의 성능 향상과 함께 지속적으로 발전하고 있다. 전통적인 방식은 영상에서 특징점을 추출하여 이를 매칭하는 방법이지만 이후 특징점 없이 픽셀 값을 직접 비교하는 Direct 방법이 개발 되었다[10,11]. 하지만 이는 모두 전문가 시스

Received : Dec. 14. 2018; Revised : Jan. 19. 2019; Accepted : Jan. 25. 2019
※ This work was supported by the Soonchunhyang University Research Fund (No. 20180404) and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2018R1C1B5086360).
† Assistant Professor, Corresponding author: Department of Electronics and Information Engineering, Asan-si, Chungcheongnam-do, Korea (hyukdoo. choi@sch.ac.kr)

템(Expert system)이기 때문에 높은 정확도를 갖는 VO를 구현하기 위해서는 내부 구조가 여러 단계로 복잡하고 모든 파라미터들이 정밀하게 조절되어야 하기 때문에 개발에 전문인력과 많은 시간이 필요하다.

최근 VO에도 심층 신경망(Deep Neural Network, DNN)을 적용한 연구들이 발표되고 있다[5,6,12]. 심층 신경망을 이용한 VO의 장점은 학습 데이터만 준비되어 있다면 내부 구조를 신경망 하나로 단순화 시킬 수 있다는 것이다. 기존 VO에 비해서 이론적, 기술적으로 단순하기 때문에 기초 지식만을 갖춘 더 넓은 범위의 개발자들에게 활용될 수 있다.

딥러닝을 활용한 또 다른 분야로 단안 영상 깊이 추정(Monocular Depth Estimation, MDE) 기술이 있다. 영상의 깊이 정보는 자율주행에 중요한 정보지만 단안 영상으로부터는 원리적으로 얻을 수가 없다. 하지만 사람이 영상에서 깊이를 느끼듯 영상의 텍스쳐로부터 깊이를 추론하는 학습 알고리즘들이 제안되었다[13]. 초기에는 지도학습을 활용했지만 이후 비지도학습을 이용해 더 다양한 데이터를 학습에 활용할 수 있게 되었다[4].

비지도학습 딥러닝 기반 MDE의 특징은 VO와 함께 학습한다는 것이다. 영상의 깊이와 자세 변화를 함께 추론하면 두 가지 추론이 서로 상응하는 결과를 내는 방향으로 비지도학습을 시킬 수 있다.

KITTI 데이터셋은 VO, 깊이 추정, 객체 인식, 옵티컬 플로우(Optical flow) 등 다양한 연구에 필요한 학습 및 평가 데이터와 벤치마크를 제공한다[14]. 많은 연구자들이 KITTI 데이터셋을 기준으로 성능을 비교하고 경쟁하여 이 분야의 연구 발전에 크게 기여하였다. 그러나 KITTI 데이터셋은 다양한 데이터들과 복잡한 디렉토리 구조로 인해 데이터를 활용하기가 쉽지 않다. 딥러닝을 이용한 영상 깊이 추정 및 VO 연구들은 KITTI 데이터셋에서 학습 데이터와 평가 데이터를 동일하게 분리하여 표준화된 비교를 해왔다[5,15]. 대용량의 데이터셋을 다운로드 받아서 데이터의 구조와 의미를 파악하고 이를 학습과 평가에 활용하는데 적합하도록 가공하는데 많은 시간과 노력이 필요하다.

따라서 본 논문에서는 이와 관련된 연구를 촉진하고자 심층 신경망 모델의 학습과 평가를 편리하게 할 수 있는 데이터와 도구를 제공한다. 이를 통해 기존 모델의 성능을 누구나 객관적으로 비교할 수 있고 새로운 모델도 쉽게 학습시켜 기존 연구들과 비교할 수 있게 된다. 본 연구가 기여하는 바는 다음과 같다.

첫째, 딥러닝 기반으로 영상 깊이 추정과 VO를 위한 새로운 모델을 쉽게 학습하고 평가할 수 있는 도구를 제공한다. KITTI 데이터셋을 분석하고 가공하는데 들이는 시간과 노력을 줄이고 새로운 모델을 개발하는데 집중할 수 있다.

둘째, 개발 환경은 GeoNet[7]의 소스코드를 기반으로 구현되었는데 GeoNet을 비롯한 대부분의 최신 연구들도 구버전의 텐서플로우(Tensorflow)를 사용한다. 그런데 최신 텐서플로우의 사용방법이 변화하여 앞으로 기존 코드는 호환이 되지 않을 것이다. 본 연구에서는 GeoNet에 텐서플로우의 최신 기술을 적용하여 새로운 연구도 최신 기술의 지원을 받으며 할 수 있게 된다.

셋째, 학습을 위해 전처리 등의 과정 없이 바로 사용 가능한 학습 데이터를 제공한다. tfrecords 형식으로 제공하기 때문에 수 만장의 영상을 직접 읽는 것보다 효율적인 학습이 가능하다.

넷째, 평가와 비교를 위한 결과 데이터셋을 제공한다. VO와 영상 깊이의 기준 값(ground truth)과 대표적인 최신 연구들의 위치 및 깊이 추정 결과를 통일된 형식으로 제공한다. 새로운 모델 개발 시에도 같은 형식으로 결과를 저장하게 된다. 이로부터 기존 연구들이 측정했던 성능 지표들을 계산할 수도 있고 새로운 지표를 개발하여 비교할 수도 있다.

다섯째, 기존의 딥러닝 기반 VO 논문들이 발표한 성능을 확인하고 다른 지표를 통해 실제적인 성능을 검증한다.

위 기능들을 구현한 소스코드는 다음 저장소에서 확인할 수 있다: https://github.com/goodgodgd/vode-bench.git.

본 논문은 다음과 같이 구성된다. 2장에서는 기존의 VO 관련 연구를 소개한다. 3장에서는 제안한 벤치마크의 구성요소들을 설명한다. 4장에서는 벤치마크를 통해 측정한 기존 연구들의 성능을 비교한다.

2. 기존 연구

VO기술은 크게 전문가 시스템과 학습 시스템으로 나눌 수 있으며 현재까지는 일반적으로 전문가 시스템의 성능이 좋다. 하지만 학습 기반 VO는 개발이 쉽고 개발에 필요한 데이터셋이 점점 대용량화 되는 추세이기 때문에 앞으로 발전 가능성이 높다. 본 장에서는 두 가지 시스템의 기존 연구들을 소개한다.

2.1 전문가 시스템 기반 VO

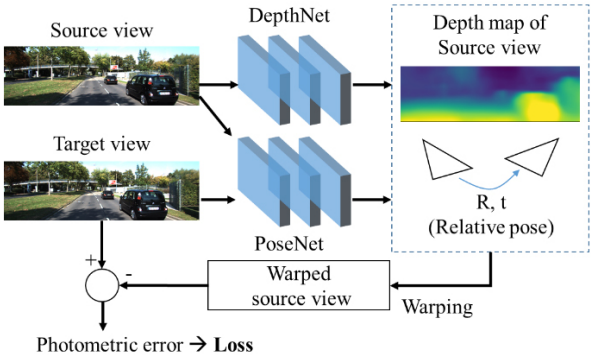
Visual SLAM 연구들의 전방(Front-end) 처리과정에는 VO 기술이 들어있다. 전문가 시스템 기반의 VO와 Visual SLAM에은 크게 특징점을 이용한 방식과 픽셀 값을 직접 사용하는 방식이 있다. ORB-SLAM[10]은 고속으로 검출 및 추출 가능한 FAST 특징점(feature)과 ORB 기술자(descriptor)를 활용한다. 현재 영상의 특징점들을 키프레임(keyframe)의 특징점들과 매칭하여 키프레임의 3차원 점 좌표들로부터 현재 영상의 상대적인 자세를 계산한다. 이를 연결하여 SLAM의 전역위치인식을 수행하며 이 과정에서 갱신한 3차원 점들의 좌표를 지역

추적(local tracking)의 지역 지도(local map)으로 사용한다. 이후 ORB-SLAM2 [16]에서는 스테레오 영상을 이용하여 스케일 모호성(scale ambiguouty)를 제거하고 성능을 크게 향상 시켰다.

반면 LSD-SLAM [11]에서 쓰이는 VO는 픽셀 값을 직접(Direct) 이용하여 광도 오차(photometric error)를 최소화하는 방식으로 영상 사이의 상대적인 자세를 계산한다. 특징점들만 활용하는 ORB-SLAM과는 달리 변화율이 있는 모든 픽셀을 사용하기 때문에 더 밀도있는 지도가 생성된다. 대표적인 두 가지 방식의 SLAM을 바탕으로 다양한 후속 연구들이 진행되었다[16-18]. 최근 Engel 등은 Direct 방식의 단점인 연산량을 줄여 고속으로 VO가 가능한 시스템을 제안하였다[17].

2.2 학습 기반 VO

딥러닝을 이용한 학습 기반 VO 방식은 최근 나온 것으로 기존 VO와 Visual SLAM 기술의 복잡한 과정을 하나의 DNN 모델로 대체하여 엔드-투-엔드(end-to-end) 학습이 가능한 것이 특징이다. 초기에는 DeepVO[3]와 같이 자세 정보가 주어진 영상 데이터셋을 기반으로 딥러닝의 지도학습을 이용한 방법이 제안되었다. 연속적인 자세를 학습하기 위해 CNN (Convolutional Neural Network)과 RNN (Recurrent Neural Network)을 결합하였으며 데이터셋에 주어진 영상의 자세를 학습하였다. 하지만 외부에서 카메라의 자세를 정밀하게 측정하는 것은 고가의 장비가 필요하고 제한된 환경에서만 가능하므로 사용할 수 있는 데이터셋이 한정되어 있다. 이후 연구에서는 영상의 자세 데이터가 없는 데이터셋을 활용해 VO와 MDE를 학습시킬 수 있는 연구들이 발표되었다[4,6,12]. [Fig. 1]은 이 과정을 표현한 것이다. [Fig. 1]에서 두 개의 네트워크가 있는데 하나는 VO 용, 하나는 MDE 용으로 각기 상대 자세와 깊이 영상 추정에 사용된다[4-7]. 그림에서 Source view 영상의 깊이와 상대 자세를 추정하면 Source view 영상을 Target view에서 본 것처럼 영상을 워핑(warping) 할 수 있다. 깊이 영상과 상대 자세 추정이 모두 정확하다면 Source view를 워핑한 영상이 Target view 영상과 같아야 한다. 차이가 있다면 깊이 추정이나 자세 추정에 오차가 있는 것이므로 이를 줄이는 방향으로 학습을 시킨다. 다음은 손실 함수를 나타낸 것이다.



[Fig. 1] Typical unsupervised learning process of DNN for VO and MDE

$$L = \sum_{p_s} |I_t(\hat{p}_t) - I_s(p_s)| \quad (1)$$

$$\hat{p}_t = K T_{s \to t} D_s(p_s) K^{-1} p_s \quad (2)$$

I_t는 Target view 영상, I_s는 Source view 영상, p_s는 I_s의 픽셀 좌표, $\hat{p}_t$는 p_s를 I_t로 사영한 픽셀 좌표, K는 카메라 행렬, D_s는 Source view의 깊이 영상이다.

Zhou 등이 발표한 SfmLearner[5]에서는 여기에 Explainability 네트워크를 추가하여 깊이 예측에 대한 신뢰도를 스스로 평가하여 움직이는 물체나 불확실한 깊이 추정이 깊이 학습 과정에 영향을 주지 않도록 하였다. Zhan 등의 연구에서는 영상을 워핑(warping)하여 얻은 오차 뿐만 아니라 특징 지도도 워핑하여 다른 영상의 특징 영상과 비교하여 오차를 줄이고자 했다. GeoNet[7]에서는 VO나 MDE 외에 옵티컬 플로우(optical flow)까지 추정하도록 네트워크를 설계하였다. 본 연구에서는 GeoNet을 기반으로 VO와 MDE 기술을 개발하고 평가할 수 있는 환경을 제공한다.

3. VODE 벤치마크

본 연구에서 소개하는 VODE (Visual Odometry and Depth Estimation) 벤치마크는 다양한 VO와 MDE 기술들의 성능을 비교할 수 있는 기능과 새로운 모델을 학습시킬 수 있는 기능이 있다. 벤치마크에서 제공하는 단계별 기능은 다음과 같다.

1. 데이터셋 준비: 'KITTI raw dataset'과 'KITTI odometry dataset'으로부터 학습과 평가에 사용할 영상들을 네트워크 입력형태로 변환해 저장한다. 이때 'train/val/test' 세 가지 split의 파일 목록도 저장한다. 이 과정은 SfmLearner나 GeoNet 등과 같다.
2. Tfrecords 준비: Tensorflow에서는 학습할 때 영상 파일과 관련 데이터를 하나씩 직접 읽어오는 것보다는 데이터셋을 하나의 'tfreocrds' 파일로 저장하여 사용하는 것을 권장한다. 벤치마크에서는 1단계에서 준비된 파일들을 tfrecords 파일로 변환해준다. VODE 벤치마크에서는 tfrecords 파일을 제공하여 1-2 단계를 생략할 수 있다.
3. 모델 학습: 기존 모델이나 새로운 모델을 학습시킬 수 있다. 학습에는 1 단계에서 만든 'tfrecords' 형식의 데이터를 사용

한다. 데이터를 불러오고 학습을 시킬때는 최신 텐서플로우에서 권장하는 'tf.data.Dataset'와 'tf.estimator.Estimator' 기능을 사용하여 기존 연구의 방식보다 편리하게 학습을 시킬수 있다.

4. 예측: 학습된 모델을 이용하여 영상에서 예측한 자세와 깊이 지도를 파일로 저장한다. 기존 방법들의 결과와 파일 형식을 맞추어 동일하게 평가가 가능하다.
5. 평가: 저장된 예측 결과를 불러와 정확도를 평가한다. 기존 연구에서 사용되던 성능 지표를 계산할 수 있고 새로운 지표를 개발하여 평가 가능하다. VODE 벤치마크에서는 기존 연구들과 성능을 쉽게 비교할 수 있도록 기존 연구들의 예측 결과를 모아서 제공한다.

평가 시스템과 제공하는 입출력 데이터셋에 대해서는 다음 세부 절에서 설명한다. SfmLearner[5]나 Geonet[7] 등의 기존 연구와의 차이점은 최신 버전의 텐서플로우에 적합하도록 소스코드와 데이터 형식이 개선되었고, 여러 연구의 결과 데이터를 통일된 형식으로 제공하며, 기존 연구의 실질적인 성능을 평가할 수 있는 새로운 평가 지표를 통해 객관적인 비교 결과를 제공하는 것이다.

3.1 평가 시스템

기존의 딥러닝 기반 VO & MDE 기술들은 연구의 연속성과 비교의 객관성을 위해 KITTI 데이터셋에서 같은 기준으로 성능을 평가해왔다. 깊이 추정 평가를 위해서 학습과 평가 데이터를 동일하게 분리하였는데 Eigen 등의 연구[15]로부터 유래되어 이를 'Eigen split' 이라 한다. 이를 통해 모델의 깊이 추정 성능을 평가하는 지표들을 [Table 1]에 정리하였다. [Table 1]에서 p_i는 픽셀 i에 대해 모델에서 예측한 깊이고 g_i는 데이터셋에서 주어진 실제 값, δ는 p_i, g_i 중 큰 것과 작은 것의 비율이다. KITTI 데이터셋은 Velodyne LiDAR 센서로 측정한 거리 값을 제공한다. 이를 카메라 영상으로 사영(project)하면 카메라 관점에서의 깊이 지도를 얻을 수 있다. Eigen split에서 테스트 프레임들에 대한 깊이 지도를 모아서 [프레임 수 × 영상 높이 × 영상 너비] 차원의 영상을 numpy의 'npy' 포맷으로 저장하면 이를 KITTI 데이터셋의 실측값과 비교하여 성능을 측정할 수 있다.

[Table 1] Performance indices of depth estimation[15]

Index	Definition
abs_rel	$mean(\lvert p_i - g_i \rvert / g_i)$
sq_rel	$mean(\lvert p_i - g_i \rvert^2 / g_i)$
rmse	$\mathrm{sqrt}(mean((p_i - g_i)^2))$
rmse_log	$\mathrm{sqrt}(mean((log\ p_i - log\ g_i)^2))$
$\delta < 1.25$	$p(\max(p_i/g_i, g_i/p_i) < 1.25)$
$\delta < 1.25^2$	$p(\max(p_i/g_i, g_i/p_i) < 1.25^2)$
$\delta < 1.25^2$	$p(\max(p_i/g_i, g_i/p_i) < 1.25^3)$

VODE 벤치마크에서는 Eigen split의 테스트 프레임에 대한 실제 깊이 지도를 제공하며 이를 바탕으로 모델에서 추정한 깊이 지도에 대한 정확도를 [Table 1]의 평가 지표로 평가해준다.

전통적인 VO 논문에서의 성능 평가는 전체 경로의 평균 오차(Average Translational Error, ATE)를 측정한다. 하지만 학습기반 VO 연구들은 경로 오차를 짧은 단위에서만 측정하는 경향이 있다. 비슷한 VO & MDE 연구들은[5-7] 연속된 5 프레임 단위로만 경로를 추정하고 5 프레임 경로의 오차들의 평균으로 전체 성능을 측정하였다. 이는 DNN을 이용한 VO의 특성상 출력이 두 영상의 상대 자세이기 때문에 지역 지도를 만들어 연속적인 절대 경로를 측정하는 기존의 VO와는 출력이 다르다. 하지만 VO의 원래 목적은 연속적인 절대 경로를 추정하는 것이기 때문에 학습기반 VO의 결과도 절대 경로를 기준으로 성능을 평가하는 항목을 추가하였다.

3.2 입출력 데이터셋

VODE 벤치마크는 VO & MDE 모델의 학습과 성능평가를 편리하게 할수 있도록 입출력 데이터셋을 제공한다. 입력 데이터셋은 KITTI 데이터셋에서 학습과 평가에 필요한 데이터를 추출하고 가공하여 학습에 바로 적용할 수 있는 데이터를 제공한다. KITTI 데이터셋의 구조가 복잡하고 다양한 캘리브레이션 설정 파일을 해석하고 Velodyne 데이터를 영상의 깊이 지도로 변환하는데 많은 노력이 필요하다. 그래서 SfmLearner 등 기존 연구에서도 데이터를 정리하는 스크립트를 오픈 소스로 공개하였지만 여전히 200GB가 넘는 KITTI 데이터셋 원본을 받아야하는 부담이 있고 이를 정리한 새로운 대용량의 데이터셋을 또 만들어야 하는 부담이 있다. 정리된 데이터셋이라고 할지라도 수십만개의 파일로 구성되어 있기 때문에 데이터 이동 및 복사에 크게 불리하다. 본 연구에서는 이를 단지 몇개의 tfrecords 형식으로 변환한 파일을 만들어 공개하였다. 대부분의 CNN을 응용한 연구가 텐서플로우 기반으로 개발되고 있기 때문에 텐서플로우 용 데이터 형식을 취하였고 다양한 타입과 크기의 데이터를 자유롭게 포함 할 수 있으며 하나의 파일에 다수의 학습 영상 데이터를 담을 수 있다. 최근에는 텐서플로우의 'tf.data.Dataset' 모듈과 'eager_execution' 모드를 이용하면 더욱 쉽게 tfrecords 안에 담겨 있는 데이터를 확인할 수 있게됐다.

입력 데이터셋은 VO 학습용 데이터셋과 MDE 학습용 데이터셋으로 나뉜다. 각 데이터셋은 'train/val/test'로 구분되며 모든 프레임 데이터는 다음 정보를 담고 있다.

- image: KITTI 원본 영상을 CNN 입력단에 넣을 수 있도록 각 영상을 128×412 크기로 줄였다. VO와 MDE 학습을 위해 3장 혹은 5장을 가로로 붙여서 저장한다.
- intrinsics_ms: 카메라 고유정보를 담은 카메라 행렬이며 보통 학습을 다중 스케일로 하기 때문에 4가지 스케일에 맞는 카메라 행렬을 제공한다.
- gt: 성능평가에 필요한 실제 값이다. VO 학습용의 경우 'image'에 저장된 5프레임의 경로를 제공한다. 첫번째 프레임의 자세를 원점으로 하여 나머지 프레임들의 상대적인 자세를 담고 있다. MDE 학습용의 경우 'image'에 저장된 3 프레임 중 가운데 프레임의 실제 깊이 지도를 갖고 있다.

출력 데이터셋은 모델의 성능 비교에 필요한 기존 연구들의 결과를 모은 것이다. VO의 결과는 'KITTI odometry dataset'에서 9번과 10번 시퀀스(sequence)의 경로를 추정한 결과를 제공한다. 결과는 TUM 데이터셋[9]의 형식을 따라 자세를 [timestamp, x, y, z, qx, qy, qz, qw] 형식으로 표현한다. 연속적인 절대 경로와 5 프레임 단위의 짧은 경로들도 제공한다.

MDE의 결과로는 'KITTI raw dataset'을 'Eigen split'으로 나누었을 때 각 모델이 추정한 테스트 영상의 깊이 지도를 제공한다. 각 깊이 지도는 [128 × 412] 형식의 부동소수점 이미지이고 이를 깊이 차원으로 묶어서 총 697 장의 영상에 대한 깊이 지도를 비교한다.

현재까지 VO는 5가지 모델의 결과를 비교할 수 있고 MDE은 3가지 모델의 결과를 비교할 수 있다. 모델의 종류와 비교 결과는 다음 장에서 소개한다.

4. 성능 평가 결과

VODE 벤치마크는 각 모델의 출력 데이터들로부터 VO와 MDE의 각종 성능 지표들을 계산하여 csv 파일로 출력한다. 여기서는 기존 논문들에서 발표한 성능뿐만 아니라 그들이 공개한 결과 값이나 모델을 직접 평가하여 나온 성능을 비교하여 실제적인 성능을 검증한다.

4.1 VO 비교 결과

VO 성능을 비교할 모델은 'orb_short', 'orb_full', 'sfmlearner'[5], 'deepvofeat'[6], 'geonet'[7], 총 5개이다. 이 모델들은 모두 단안 카메라를 이용하기 때문에 실제 데이터를 이용한 이동 스케일 보정 후 성능을 평가한다. 이중 'orb_short', 'orb_full'은 전문가 시스템기반 Visual SLAM인 ORB SLAM [10]으로부터 나온 결과이다. 'orb_short'은 5 프레임 단위로만 SLAM을 한 결과이고 'orb_full'은 전체 시퀀스에 대해서 SLAM을 한 결과이다. ORB SLAM의 결과는 Zhou 등의 연구[5]에서 오픈소스로 공개한 결과를 사용한다. 나머지 3개의 모델은 학습기반 모델로서 최근 발표된 논문들 중 소스코드와 결과를 오픈소스로 공개한 연구들을 선택했다. 학습기반 모델들은 'KITTI odometry dataset'의 0~8번 시퀀스를 이용해 학습되었으며 9~10번 시퀀스를 이용해 평가를 한다.

[Table 2]는 각 모델의 이동 오차와 회전 오차의 평균과 분산을 보여준다. (p) 표시는 해당 논문에서 성능을 가져온 것인데 ORB SLAM의 경우 해당 성능지표가 논문에 나와있지 않으므로 SfmLearner 논문에서 제시한 성능을 가져왔다. 모델명 옆의 (d) 표시는 오픈소스와 함께 제공된 모델의 VO 출력 데이터를 가지고 다시 성능을 측정한 결과이다. (t) 표시는 모델에 입력을 넣어 직접 출력을 추론하고 이로부터 성능을 측정한 결과이다. 'orb_short'를 제외하고 대부분 논문에 제시된 성능보다 오차가 크게 나온 것을 볼 수 있다. 특히 논문 저자가 직접 제공한 출력 데이터를 저자가 오픈소스로 공개한 평가 도구를 참고하여 동일한 방법으로 측정을 했는데도 오차 값에 큰 차이가 있는 것은 설명하기 어렵다. 그렇지만 출력 데이터나 직접 추론한 출력에서 보여지는 결과의 상대적인 성능은 논문에서 제시한 것과 비슷했다. 'orb_full'의 성능이 가장 좋았고 그 다음으로 'geonet'의 성능이 좋았다. 하지만 이는 논문에서 주어진 이동 오차(translational error)만을 볼 때의 얘기고

[Table 2] Translational and rotational errors of visual odometry models. The errors are represented by 'mean + standard deviation'

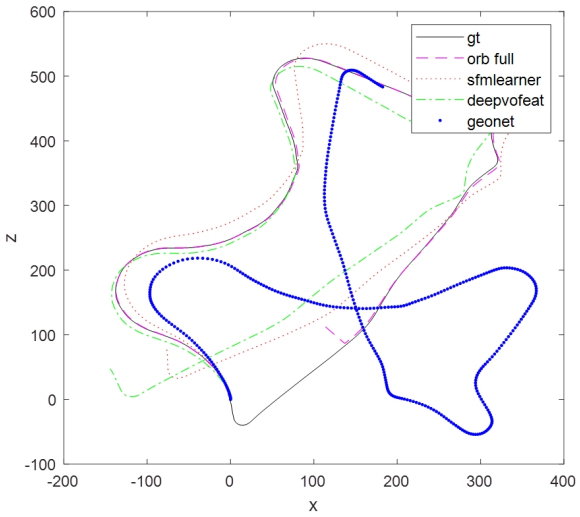
	Seq.	Translation (m)	Rotation (deg.)
orb_short (p')	9	0.064+0.141	-
	10	0.064+0.13	-
orb_short (d)	9	0.029+0.056	0.0387+0.094
	10	0.0376+0.1	0.0629+0.244
orb_full (p')	9	0.014+0.008	-
	10	0.012+0.011	-
orb_full (d)	9	0.0275+0.02	0.0275+0.014
	10	0.024+0.025	0.0328+0.018
sfmlearnner (d)	9	0.0391+0.04	0.1809+0.274
	10	0.0372+0.039	0.2996+0.397
sfmlearner (t)	9	0.0347+0.027	0.1508+0.107
	10	0.0262+0.027	0.1434+0.101
deepvofeat (d)	9	0.0294+0.025	0.2286+0.185
	10	0.0292+0.03	0.2396+0.228
geonet (p)	9	0.012+0.007	-
	10	0.012+0.009	-
geonet (t)	9	0.0271+0.023	0.1953+0.133
	10	0.0266+0.026	0.1931+0.134

회전 오차(rotational error)를 보면 학습기반 모델들의 오차가 ORB SLAM에 비해 매우 큰 것을 볼 수 있다. 회전 오차는 경로가 길어질수록 경로의 궤적을 크게 바꾸기 때문에 프레임 사이의 상대적인 이동 오차보다 전체적인 오차에 더 큰 영향을 준다. 즉 5 프레임 경로의 이동 오차로만 성능을 평가하고 이를 기반으로 ORB SLAM 보다 더 성능이 좋다는 GeoNet 논문의 주장은 설득력이 떨어진다.

이를 정확히 비교하기 위해서는 전체 시퀀스에서 절대 경로 오차를 봐야 한다. 프레임 사이의 상대 자세만 출력하는 학습기반 모델의 출력을 연결하여 전체 경로를 복원하였다. 복원된 경로의 오차를 [Table 3]에 정리하였다. 첫 번째 행의 숫자는 학습기반 VO에서 경로를 복원하기 위해 사용한 프레임 간격이다. 학습 모델들은 대부분 5 프레임씩 경로를 추론하므로 1~4 프레임 간격으로 전체 경로를 복원할 수 있다. 표의 결과를 보면 프레임 간격이 클수록 오차가 줄어드는 것을 확인할 수 있다. 무엇보다도 ORB SLAM에 비해 성능 격차가 크다는 것을 확인할 수 있다.

[Table 3] Average translational errors of full trajectories. The numbers in the top row mean the frame interval for trajectory reconstruction

	Seq.	1	2	3	4
orb_full	09	3.3			
	10	3.6			
sfmlearner (t)	09	68.0	60.6	57.6	53.1
	10	80.4	59.3	63.5	40.1
deepvofeat (d)	09	50.9	51.0	51.0	51.1
	10	58.8	58.8	58.8	58.9
geonet (t)	09	317.1	301.3	285.8	282.1
	10	190.3	169.3	152.0	137.1

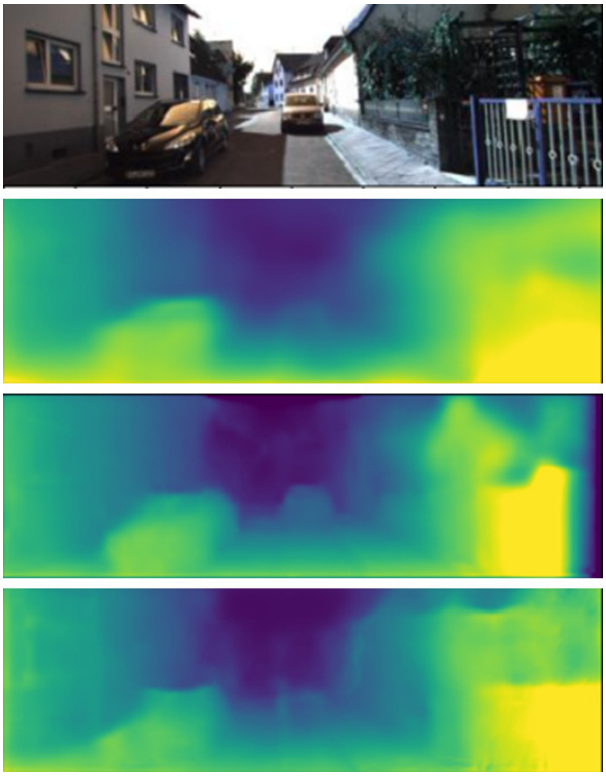


[Fig. 2] The full trajectories of the sequence 09 provided by the ground truth and the four models: orb_full, sfmlearner (t), deepvofeat (d), and geonet (t). The frame interval for the learning based models is 4

전체 경로오차는 [Fig. 2]에서도 확인할 수 있다. [Fig. 2]는 각 학습 모델을 통해 복원할 수 있는 가장 정확한 경로를 보여준다. 모든 모델에서 프레임 간격을 최대로 할 때 성능이 높고 'sfmlearner'에서는 주어진 출력보다 모델에서 직접 출력한 경로의 결과가 더 정확하다. 4개 모델의 경로를 검은 선으로 표시된 실제 경로(gt)와 비교했을 때 'geonet'의 성능이 가장 나쁘다는 것을 알 수 있다. 이는 상대적으로 초기에 높은 회전오차가 발생하여 전체적인 경로의 방향이 실제와 멀어졌기 때문이다.

4.2 MDE 비교 결과

깊이 지도 추정은 앞서 제시된 학습기반 모델에 대해서 성능을 비교한다. [Fig. 3]은 영상을 보고 각 모델이 예측한 깊이 지도이다. 깊이 추정에서도 마찬가지로 논문에서 제시하는 성능과 주어진 출력 데이터나 모델을 이용해 얻은 출력으로부터 계산한 성능들을 비교한다. 깊이 지도에 대한 성능 지표는 [Table 1]에 제시되어 있다. VODE 벤치마크에서는 표의 모든 지표를 계산할 수 있지만 여기서는 성능을 체감할 수 있는 몇



[Fig. 3] Depth map prediction examples. From the top: an RGB image, and the predicted depth maps of sfmlearner (t), deepvofeat (d), and geonet (t)

[Table 4] Depth prediction performances

model	abs_rel	rms	$\delta < 1.25$
sfmlearner (p)	0.198	6.565	0.718
sfmlearner (d)	0.198	6.565	0.718
sfmlearner (t)	0.212	7.286	0.665
deepvofeat (p)	0.135	5.585	0.820
deepvofeat (d)	0.136	5.466	0.833
geonet (p)	0.153	5.311	0.847
geonet (t)	0.186	6.439	0.734

가지 지표만 [Table 4]에 표시한다.

[Table 4]의 결과를 보면 'sfmlearner'나 'deepvofeat'의 결과는 논문의 성능과 재측정한 성능이 비슷한 것을 볼 수 있으나 'geonet'의 경우에는 차이가 있다. 논문의 결과는 'deepvofeat'과 비슷하지만 저자가 공유한 모델을 받아 직접 평가한 결과는 'sfmlearner' 정도의 상대적으로 낮은 성능이 나온다. 따라서 'deepvofeat'의 성능이 더 낫다고 볼 수 있지만 'deepvofeat'의 경우 딥러닝 프레임워크로 caffe를 사용하고 모델이 지나치게 복잡하기 때문에 이를 변형하여 응용하는 것이 쉽지 않다.

5. 결 론

본 논문에서는 학습기반 VO를 비교하여 평가할 수 있고 이를 바탕으로 새로운 모델을 쉽게 개발할 수 있는 VODE 벤치마크를 소개하였다. 벤치마크를 통해 기존 연구들의 성능을 검증하여 각 모델에 대한 객관적인 성능 지표를 제시하였다. 성능을 확인한 결과 학습기반 VO는 회전오차로 인해 자세 추정 오차가 경로가 길어질수록 크게 벌어진다는 것을 확인할 수 있었다. MDE에서는 SfmLearner와 DeepVOFeat 논문과 결과가 비슷하게 나왔지만 GeoNet의 경우 상당한 차이가 있었다.

이 연구는 향후 학습 VO & MDE 연구를 함에 있어서 새로운 모델을 학습하고 검증하는데 필요한 데이터와 도구를 제공하므로 후속 연구자들은 모델의 설계에만 집중할 수 있다. 이를 바탕으로 학습기반 시스템의 장점을 살리면서 이미 발전되어 있는 전문가 시스템기반 VO의 정확도를 따라잡을 수 있는 모델을 연구할 것이다.

References

[1] L. Deng and D. Yu, "Deep Learning: Methods and Applications," *Now Foundations and Trends*, vol. 7, no. 3, pp. 197-387, 2014.

[2] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436-444, 2015.

[3] S. Wang, R. Clark, H. Wen, and N. Trigoni, "Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks," *2017 IEEE International Conference on Robotics and Automation (ICRA)*, Singapore, pp. 2043-2050, 2017.

[4] R. Li, S. Wang, Z. Long, and D. Gu, "Undeepvo: Monocular visual odometry through unsupervised deep learning," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia, pp. 7286-7291, 2018.

[5] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, DOI: 10.1109/CVPR.2017.700.

[6] H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, and I. M. Reid, "Unsupervised Learning of Monocular Depth Estimation and Visual Odometry with Deep Feature Reconstruction," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City*, UT, USA, pp. 340-349, 2018.

[7] Z. Yin and J. Shi, "GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, DOI: 10.1109/CVPR.2018.00212.

[8] J. Fuentes-Pacheco, J. Ruiz-Ascencio, and J. M. Rendón-Mancha, "Visual simultaneous localization and mapping: a survey," *Artificial Intelligence Review*, vol. 43, no. 1, pp. 55-81, Jan., 2015.

[9] T. Taketomi, H. Uchiyama, and S. Ikeda, "Visual SLAM algorithms: a survey from 2010 to 2016," *IPSJ Transactions on Computer Vision and Applications*, vol. 9, no. 16, 2017, DOI: DOI 10.1186/s41074-017-0027-2.

[10] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147-1163, Oct., 2015.

[11] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," *European Conference on Computer Vision*, pp. 834-849, 2014.

[12] R. Mahjourian, M. Wicke, and A. Angelova, "Unsupervised Learning of Depth and Ego-Motion from Monocular Video Using 3D Geometric Constraints," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, pp. 5667-5675, 2018.

[13] A. Saxena, S. H. Chung, and A. Y. Ng, "Learning depth from single monocular images," *18th International Conference on Neural Information Processing Systems*, Vancouver, British Columbia, Canada, pp. 1161-1168, 2005.

[14] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231-1237, 2013.

[15] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," arXiv: 1406.2283 [cs.CV], 2014..

[16] R. Mur-Artal and J. D. Tardós, “Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras,” *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255-1262, Oct., 2017.
[17] J. Engel, V. Koltun, and D. Cremers, “Direct sparse odometry,” arXiv:1607.02565 [cs.CV], 2016.
[18] J. Engel, J. Stückler, and D. Cremers, “Large-scale direct SLAM with stereo cameras,” *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, pp. 1935-1942, 2015.
[19] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, “A benchmark for the evaluation of RGB-D SLAM systems,” *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, Portugal, pp. 573-580, 2012.

최 혁 두

2009 연세대학교 전기전자공학과 (학사)
2014 연세대학교 전기전자공학과 (박사)
2017 LG전자 CTO부문 선임연구원
2018~현재 순천향대학교 조교수

관심분야: 로봇, 인공지능, 딥러닝, 컴퓨터 비전