

# SLAM 불안정 환경별 ATE 변화량 실시간 예측을 위한 시계열 회귀 분석

최서정, 엄서린, 황재운, 김준영

성신여자대학교

20221439@sungshin.ac.kr, 20211365@sungshin.ac.kr, 20211423@sungshin.ac.kr, jkim@sungshin.ac.kr

## Time-Series Regression Analysis for Real-Time Prediction of ATE Variation in Unstable SLAM Environments

Seo Jeong Choi, Seo Rin Eum, Jae Yoon Hwang, Joon Young Kim

Sungshin Women's Univ.

### 요 약

본 논문은 SLAM(Simultaneous Localization and Mapping) 수행 중 발생하는 위치 오차의 변동성을 실시간으로 예측하고, 환경 특성에 따른 예측 성능 차이의 분석에 목적을 둔다. 실내 환경 중 복도와 유리 재질 환경을 시뮬레이터 기반으로 구성하였고, LiDAR 데이터 기반 시계열 특징 추출 및 window pooling 전처리 과정을 거쳐 회귀 모델의 학습 및 검증의 ATE(Absolute Trajectory Error) 변화량 예측 프레임워크를 이용하였다. 성능 평가 지표로는 정확도(MAE, RMSE)와 단일 샘플 추론 시간(ms/sample)을 함께 이용하여 모델별 trade-off를 정량적으로 분석하였다.

### I. 서 론

SLAM은 로봇이 자신의 위치를 추정하면서 동시에 주변 환경의 지도를 생성하는 기술로, 지속적으로 로봇 및 자율주행 분야내 중요 연구 주제로 다뤄져 왔다. 다만 실 환경 내 센서 노이즈, 차폐(occlusion) 등 다양한 요인으로 인해 위치 추정 정확도가 저하되며, 누적된 오차는 곧 로봇의 주행 안전성 문제와 직결된다. SLAM 성능의 정량적 평가 및 사전 예측 연구는 지속적으로 이어져 왔다. 성능 평가지표로는 ATE와 RPE(Relative Pose Error)가 널리 사용된다[1]. ORB-SLAM3의 ATE를 앙상블 회귀 모델을 이용하여 예측한 케이스도 존재하며 이외에도 SLAM 알고리즘 성능 사전 추정 등 다양한 방법이 제안되고 있다 [2][3]. 그러나 대부분의 기존 연구는 일반적인 주행 환경에서의 공개 데이터셋(KITTI, TUM-VI, EuRoC 등)에 의존하고 있기 때문에, SLAM의 불안정성을 유도하는 특수 환경(예: 긴 복도, 유리 벽)을 충분히 반영하지 못하는 한계를 가진다.

본 연구에서는 특수 환경을 직접 설계하고, Gazebo 시뮬레이터를 활용하여 직선 복도 및 유리 재질 기반의 실내 환경에서 로봇 주행 데이터를 수집하였다. 이를 통해 범용 데이터셋에 의존하지 않고, 특수 환경에 한정된 데이터를 기반으로 SLAM의 ATE 변화량을 정량적으로 예측하였으며, 다양한 회귀 모델의 성능을 비교하고 샘플 단위 예측 속도를 분석함으로써 실시간성에 대한 평가도 함께 수행하였다.

### II. 실험 환경

본 연구의 실험은 ROS 2 Humble 기반 Gazebo 시뮬레이터에서 Turtlebot3 Burger 모델을 사용하여 수행하였다. SLAM 알고리즘 성능 저하가 발생하는 상황 특정 위해 두 가지 대표적 실험 환경을 설계하였다.

- 복도 환경 (corridor): 대칭 및 비대칭 구조를 포함하는 긴 통로 환경으로, 특징점이 부족한 대칭 구간이 주를 이룸.
- 유리 환경 (glass wall): 복도 환경 구성과 동일하게 하되, 벽면을 유리 재질로 구성함으로써 LiDAR의 반사 현상이 발생하도록 함. 추가로 laser\_retro=0.1, transparency=0.6 설정으로 유리 반사/투명함 구현.

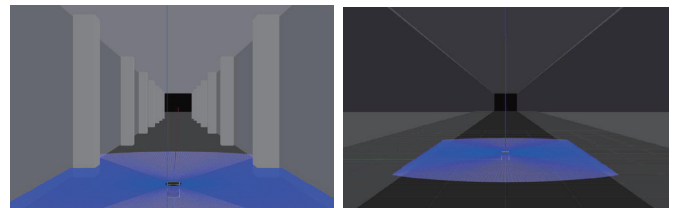


그림 1. 복도 환경과 유리 환경 구성 예시 (좌: 복도, 우: 유리)

총 6개 환경별 world 파일을 제작하였으며, 각 world내 로봇의 주행 시작 위치를 좌측, 중앙, 우측의 세 지점으로 구분하여 로그 데이터 수집을 다양화하였다. 로봇은 world의 각 위치마다 0.2m/s부터 1.0m/s까지 0.2m/s 간격으로 총 5개의 속도 조건(0.2 0.4 0.6 0.8 1.0)에서 최소 40초에서 최대 120초간 등속 직선 주행을 수행하였으며, 모든 주행은 ROS 2의 rosbag을 이용해 /scan, /tf 등의 주요 토픽을 포함하여 기록되었다.

### III. 프레임워크 구성

본 연구는 SLAM 수행 중 오차 변화량의 실시간 예측을 위해, proxy delta ATE를 예측 대상으로 설정한 시계열 회귀 파이프라인을 제시한다. 전체 과정은 1) 시계열 특징 추출, 2) Proxy ATE 계산, 3) 윈도우 기반 특징 집계, 4) 회귀 모델 학습 및 검증의 4단계로 구성된다.

1) 시계열 특징 추출: 본 연구의 ATE 변화량 예측을 위해서는 LiDAR scan 데이터의 특징만을 추출하여 진행하였다. 각 scan 프레임으로부터 평균, 분산 등의 기본 통계량뿐 아니라 이전 프레임과의 변화율 등 시계열 기반 특징도 추출하였으며, 이는 scan matching 실패로 나타나는 급격한 변화에 따른 이상치를 반영한다. 추가로 scan 대칭성 및 entropy와 같은 구조적 특징도 포함하여, 이후 윈도우 기반 집계에 활용하였다.

2) Proxy ATE 계산: SLAM 정확도 지표로 사용되는 ATE는 일반적으로 ground-truth 궤적  $p_i$  와 SLAM 추정 궤적  $\hat{p}_i$  간의 정합(alignment) 통해 얻을 수 있고, 수식 (1) 같이 평균 제곱근 오차(Root Mean Square

Error, RMSE)로 정의된다. N은 전체 프레임 개수이다.

$$ATE = \sqrt{\left(\frac{1}{N}\right) \sum_{i=1}^N \|p_i - \hat{p}_i\|^2} \quad (1)$$

그러나 본 연구는 ground-truth를 활용할 수 없는 실시간 주행 환경을 가정하여, 추정 궤적만으로 위치 오차를 근사적으로 계산한 proxy ATE를 사용하였다. 특정 시점  $t$ 에서 SLAM 추정 위치  $\hat{p}_t$ 와 초기 기준(예:  $t=0$ )에서의 추정 위치  $\hat{p}_t^{ref}$  간의 차이를 기반으로 계산된다.

$$ATE_t = \|\hat{p}_t - \hat{p}_t^{ref}\| \quad (2)$$

3) 윈도우 기반 특징 집계: 추출된 프레임 단위 특징 벡터들에 길이 W의 슬라이딩 윈도우를 적용하여 윈도우 단위로 특징을 집계하는 pooling 작업을 수행한다. 본 실험에서는 window 길이를 20, step 간격을 5로 설정하여 슬라이딩 윈도우 기반의 통계 특징을 구성하였다. 각 윈도우를 구성하는 특징 벡터들의 평균, 표준편차, 중앙값, IQR 등을 계산하였다. 또한, 과정 1에서 구한 proxy ATE 값을 이용하여 각 윈도우 구간의 시작점과 끝점의 차이를 proxy delta ATE로 정의하고, 이를 SLAM 오차의 상대적 변화량으로 해석하여 예측 대상으로 활용하였다.

4) 회귀 모델 학습 및 검증: 최종적으로 집계된 특징 벡터를 회귀 모델 학습의 입력으로 사용하였고, 타깃으로는 proxy delta ATE 값을 사용하였다. 본 연구에서는 앙상블 회귀 모델인 DecisionTree, LightGBM, RandomForest, XGBoost를 이용한 학습을 진행하였고, 환경별 특성에 따른 모델별 예측 정확도 차이와 추론 속도의 비교를 위해 복도 환경과 유리 환경을 분리해 학습과 검증을 수행하였다. 앙상블 회귀 모델은 복잡한 학습 구조를 갖는 딥러닝 모델에 비해 상대적으로 적은 양의 데이터로도 높은 예측 성능을 확보할 수 있으며, 본 연구 대상 모델이 실시간 환경에서의 빠른 추론이 요구된다는 점을 고려하여, 계산 효율성이 높은 트리 기반 모델을 중심으로 실험을 구성하였다.

#### IV. 실험 결과

복도(corridor) 환경과 유리(glass wall) 환경으로 구분하여 로봇 주행을 통해 얻은 데이터로 4가지 회귀 모델(DecisionTree, LightGBM, RandomForest, XGBoost)의 성능을 비교하였다. 모델 성능은 예측 정확도(MAE, RMSE)와 단일 샘플 추론 시간(ms/sample)을 기준으로 평가하였다. MAE(Mean Absolute Error)는 예측값과 실제값 간 절대적 차이의 평균으로, 모든 오차를 동일하게 반영하는 평가 지표라면, RMSE(Root Mean Square Error)는 큰 오차에 더 민감하게 반응하므로 이상치가 존재하는 경우 더 큰 영향을 받는다. Ms/sample는 본 연구에서는 전처리 및 특징 추출 단계를 제외한, 준비된 특징 벡터의 입력에 대한 순수 추론 시간(inference-only)을 의미하며, 이를 통해 실시간 추론 가능성을 판단한다. 이러한 평가 지표들은 기존 SLAM 신뢰도 평가 및 failure detection 연구에서도 활용된 바 있다[1][2].

표 1. 복도 환경에서의 회귀모델별 평가 지표 결과

모델	MAE	RMSE	ms/sample
DecisionTree	<b>1.80</b>	<b>3.41</b>	<b>0.0131</b>
LightGBM	2.39	3.65	0.3177
RandomForest	<b>1.87</b>	<b>3.23</b>	2.4580
XGBoost	2.25	3.55	<b>0.0918</b>

복도 환경에서는 DecisionTree와 RandomForest가 낮은 MAE와 RMSE를 기록하며 안정적인 예측 성능을 보였다. 예측 속도(ms/sample) 측면까지 고려했을 때 DecisionTree가 가장 우수하였다.

유리 환경에서는 RandomForest가 복도 환경에서와 마찬가지로 낮은

MAE와 RMSE를 기록하여 정확도 면에서 우수하였으며, XGBoost는 DecisionTree와 함께 속도와 정확도의 균형이 잘 맞는 모델로 나타났다.

두 환경을 종합적으로 볼 때, 정확도 지표 기준으로는 RandomForest가 가장 우수했으며, 속도 평가 지표에서는 DecisionTree와 XGBoost가 가장 우수하였다. 예측 정확성과 실시간성을 모두 고려한 균형성 측면에서는 DecisionTree가 적합한 모델로 판단되었다.

표 2. 유리 환경에서의 회귀모델별 평가 지표 결과

모델	MAE	RMSE	ms/sample
DecisionTree	<b>2.45</b>	4.38	<b>0.0127</b>
LightGBM	2.74	4.25	0.1372
RandomForest	<b>2.26</b>	<b>3.97</b>	1.7766
XGBoost	2.65	<b>4.18</b>	<b>0.0647</b>

이를 실험 환경과 모델별 특성을 고려하여 해석하면, 복도 환경은 구조가 단순하고 변화가 적은 특성상, 단일 결정 트리 모델인 DecisionTree로도 충분한 성능을 낼 수 있었던 것으로 보인다. 반면 유리 환경에서는 반사나 노이즈로 인해 LiDAR scan의 변동성이 커지기 때문에, 여러 트리의 병렬 학습을 통해 평균을 내는 RandomForest가 더욱 강한 예측 성능을 보였다. 또한 XGBoost와 LightGBM은 모두 Gradient Boosting 기반 앙상블 모델로, 각각 수평, 수직 방향으로의 트리 성장 방식을 지닌다. 때문에 XGBoost이 보다 균형 있게 트리를 생성하여 과적합에 덜 민감하다는 모델 특성상 유리 환경에서도 정확도와 속도의 균형을 유지할 수 있었다.

#### V. 결론

본 연구는 SLAM의 오차 발생에 따른 ATE 값의 변화량을 예측하기 위해 일반적인 주행 데이터 기반 데이터셋이 아닌, SLAM 불안정성을 유도하는 특성을 지닌 환경 중 복도 환경과 유리 환경을 특정하여 직접 시뮬레이션 환경을 설계하고, 데이터를 수집하였다. 제시한 프레임워크 기반 데이터 전처리 및 학습을 진행하였고, 각 환경에서의 모델별 예측 성능을 정확도와 속도 지표를 통한 정량적 trade-off 분석을 수행하였다.

환경 특성에 따라 예측 모델의 성능이 상이하게 나타났다는 점, 그리고 성능평가에 있어서 실시간성을 고려한 접근이 중요함을 확인할 수 있었다. 향후에는 다양한 실내의 환경을 반영한 확장된 실험과 실제 주행 데이터 기반의 평가를 통해 예측 모델의 일반화를 시도할 수 있을 것이다.

#### ACKNOWLEDGMENT

본 연구는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (2022R1G1A1009023).

#### 참고 문헌

- [1] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A Benchmark for the Evaluation of RGB-D SLAM Systems," IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 573 - 580, 2012.
- [2] I. Ali, B. (Selina) Wan, and H. Zhang, "Prediction of SLAM ATE Using an Ensemble Learning Regression Model and 1-D Global Pooling of Data Characterization," arXiv preprint, arXiv:2303.00616, 2023
- [3] M. Luperto, F. Amigoni, and V. Castelli, "Predicting Performance of SLAM Algorithms," arXiv preprint, arXiv:2109.02329, 2021.