

Term Project Report

- Build a visual localization pipeline -

Name : 남형진 (Hyeongjin Nam)

1. Introduction

이번 프로젝트에서는 사진 한 장만으로 위치와 포즈를 추정할 수 있는 Visual Localization 알고리즘을 개발하였다. 사용된 NAVER LABS Indoor dataset에는 timestamp마다 image, point cloud이 존재하며, camera parameter, camera pose, lidar pose로 구성되어 있다. 하나의 이미지를 입력으로 받아, 이미지 촬영자의 위치를 추측하는 것이 이 프로젝트의 목표이다. 이 알고리즘은 그림 1의 파이프라인을 따르고 있다. 먼저, 입력 이미지에 대해 global descriptor를 계산한다. 계산한 global descriptor를 토대로 미리 구성한 database에서 이미지를 검색하고 유사한 30개의 image를 얻는

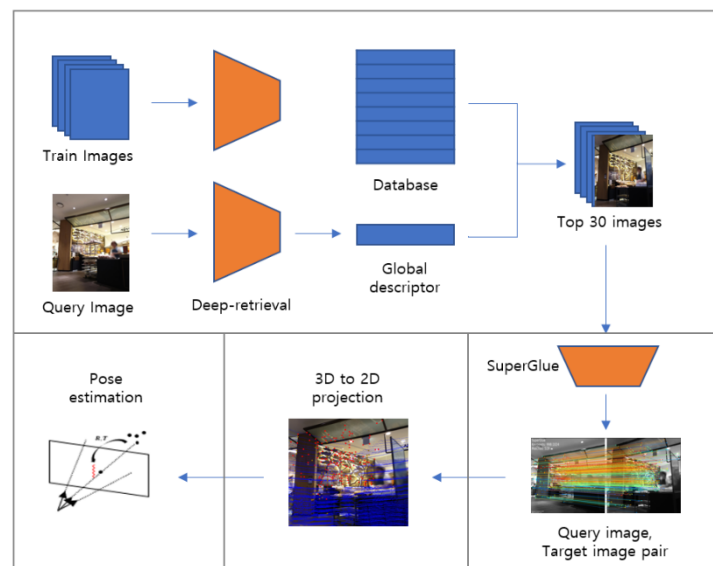


그림 1 Visual localization pipeline

다. 그리고 local descriptor를 계산하여 match point가 많은 이미지를 하나 뽑는 reranking 작업을

진행한다. 찾은 image에서 가까운 timestamp에서 기록한 point cloud를 image에 projection하여 point cloud – feature point 쌍을 생성한다. 생성된 쌍을 토대로 query image와 point cloud 간의 pose를 계산한다. 구현한 알고리즘의 코드는 이 url에서 확인할 수 있다. (https://github.com/hygenie1228/visual_localization) 본 레포트에서 사용된 이미지는 github 코드 상의 visual_localization/results에 있고, 지하 1층 data에 대한 실험 결과는 visual_localization/results/result.json에 저장되어 있다.

2. Problem Approach

A. Image retrieval

입력된 이미지의 global descriptor를 이용하여 가장 유사한 30개의 이미지를 검색한다. Global descriptor를 추출하는 방법으로, NetVLAD 방법을 사용한다. NetVLAD로 NAVER LABS Europe에서 개발한 Deep-retrieval 모델을 사용하였다 [2]. 추가 학습이 없이 pretrained model을 사용하였고, 성능을 위해 입력 이미지의 너비, 높이를 1/2로 축소시키는 preprocessing 작업을 거친다. 먼저, 검색을 위해서 train dataset에 대해 global descriptor를 계산하여 database를 미리 구축한다. 그런 다음 query 이미지의 global descriptor와 cosine similarity가 높은 30개의 이미지를 출력한다. 그림 2는 하나의 이미지에 대해 검색된 상위 30개 이미지 중 5개만 도시한 것이다. 그 결과, query 이미지의 장소와 같은 장소의 사진이 검색됨을 볼 수 있다.





Query Image	검색된 상위 5개의 Image
	    

그림 2 Image retrieval 결과

B. 2D feature matching & Reranking

검색된 image pair사이의 match point를 찾기 위해 두 이미지의 local descriptor를 추출하고 두 descriptor를 매칭한다. 매칭을 위해 SuperGlue 모델을 사용하였다 [3]. 검색된 30개의 이미지에 대해 매칭 결과를 비교한다. Confidence가 0.7 이상인 point를 inlier로 간주하고, inlier가 가장 많은 검색 이미지를 추려낸다 (reranking). 본 구현에서는 reranking의 최상위 1개의 이미지만 활용한다. 그림 3은 그림 2의 검색 결과에서 inlier수가 가장 많은 이미지이다. 그림 4는 query image와 target image 간의 match point를 visualize한 것이다. 선이 붉을수록 confidence가 높다.

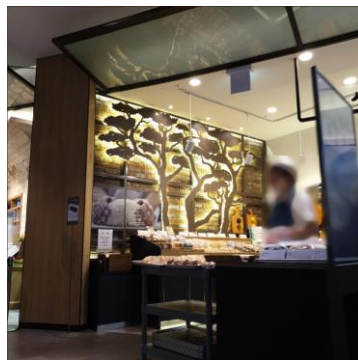


그림 3 Reranking 결과, 선택된 최종 이미지

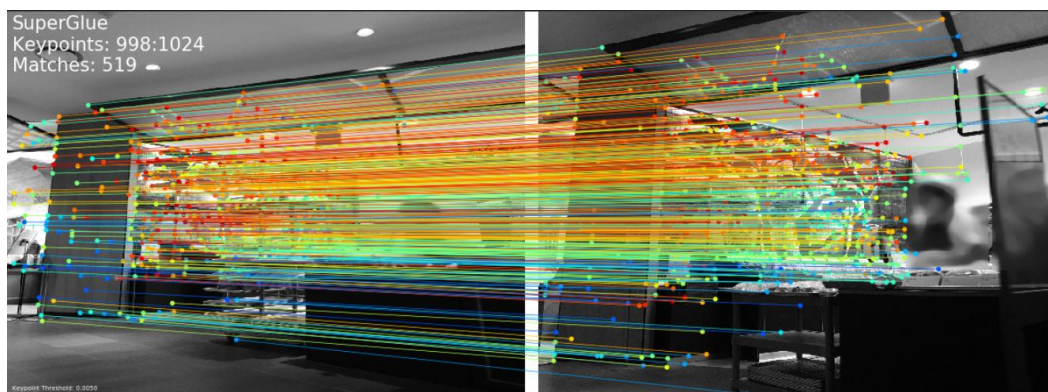


그림 4 입력 image와 검색된 image의 feature matching

C. 3D point projection

위의 과정을 통해 얻어낸 query image와 matching된 target image를 사용하여 pose를 추정한다. 추정에 앞서, 해당 장소에 맞는 point cloud를 projection한다. Target image의 timestamp를 기준으로 주변 timestamp에 얻은 16개의 lidar point cloud를 load한다. Train dataset에 있는 camera pose, lidar pose를 사용하여 target image에 projection한다. 그림 5에서 빨간 점은 Reranking 과정에서 얻은 match point를 나타낸다. 파란 점은 point cloud data를 target image에 projection한 결과이

다. Match point와 projection point간의 간격이 7.5 pixel 미만이면서 가장 가까운 점을 매칭한다. 즉, 해당 3D point를 projection 했을 때, match point가 된다는 의미이다. 그러한 점을 초록색 점으로 나타냈다. 매칭된 3D point를 3D 공간 상에 표현한 것이 그림 6이다.

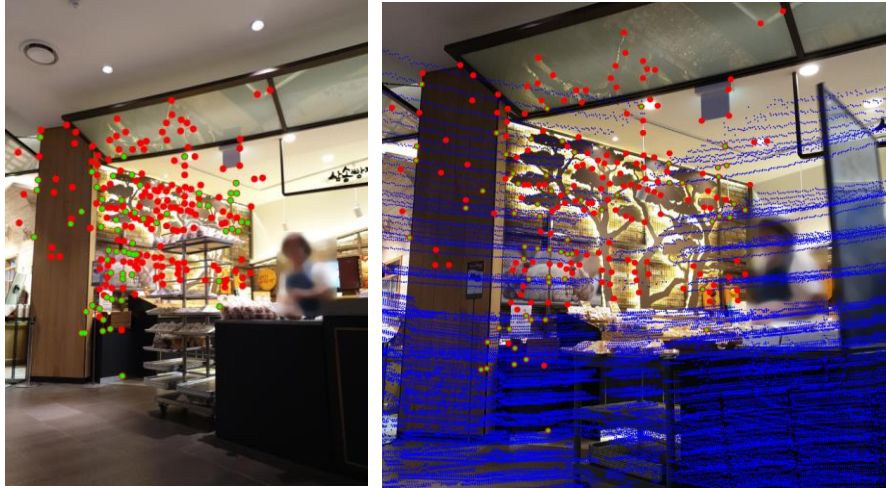


그림 5 Query image의 feature point – point cloud pair (좌), Retrieval image의 point cloud projection 결과 및 feature point – point cloud pair (우)

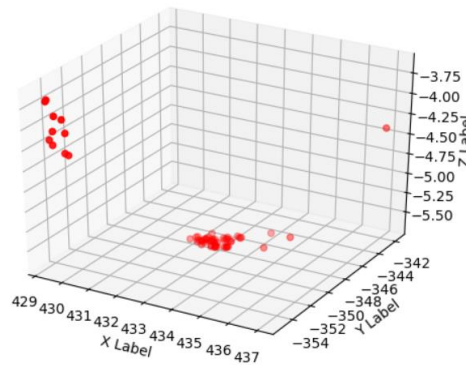


그림 6 선택된 point cloud의 3차원 분포

D. Pose estimation

위에서 얻은 최종 match point를 사용하여 3D point와 2D feature point간의 pose를 추정한다. RANSAC 기법을 사용하여 rotation matrix, translation matrix를 계산한다. 이 때, opencv의 패키지를 사용하며, inlier threshold는 16으로 둔다. 만약 match point가 8개 미만일 경우, RANSAC 알고리즘을 수행하기 매우 부족한 data 수이므로 estimation을 하지 않는다. 이 과정에서 query image

의 camera parameter도 함께 사용한다. 그림 7은 위에서 입력한 이미지의 pose estimation 결과이다. Test dataset의 모든 이미지에 대해 A~D 과정을 반복한다.

```
{ "floor": "b1", "name": "AC01324955_156
626716070000.jpg", "qw": 0.77732897963
16094, "qx": 0.5287398084649076, "qy":
-0.1842662364716348, "qz": -
0.28677487070165847, "x": 434.817126955
94723, "y": -346.90935658699686, "z": -
6.461227669866862 }
```

그림 7 Pose estimation 결과

3. Result & Conclusion

NAVER LABS Indoor dataset에서 지하 1층 data image에 대해서 visual localization 실험을 진행하였다. 표 1은 1961 이미지에 대해 예측 결과를 NAVER 테스트 서버로 보내 얻은 정확도이다. Pose의 위치, 각도 오차가 다음 3단계의 threshold 이하일 때 정답으로 간주한다.

	(0.25m, 10.0°)	(0.5m, 10.0°)	(5.0m, 10.0°)
B1 dataset accuracy (%)	15.91	26.82	34.53

표 1 NAVER LABS Indoor dataset 지하 1층 테스트 결과

정확도를 저해하는 이유를 분석한 결과, point cloud와 feature point matching 작업이 정확도에 많은 영향을 끼친다. Pose estimation에서 전체 point 대비 inlier 수가 적음을 확인할 수 있는데, 이 경우 RANSAC은 좋지 못한 성능을 보인다. 그렇기 때문에 많은 inlier수를 확보하는 것이 중요하다는 것을 알 수 있다. Reranking을 진행한 뒤, 최상위 이미지 1개만 사용했는데, 여러 개의 이미지에 대해 point cloud를 projection하여 많은 point를 얻어낼 수 있을 것이다. 또한, 하나의 feature point에 대해 7.5 pixel 내에 여러 projected point가 있는데, 여기서 RANSAC등을 통해 잘못 projection 된 3D point를 제거하면 좋은 성능이 나올 것으로 보인다. 또한 matching model을 NAVER LABS Indoor dataset으로 fine tuning하여 confidence가 높은 많은 match point를 얻는 것도 좋은 방안이 될 수 있다. 이러한 여러가지 방법으로 개선하여 알고리즘을 계속 개선할 예정이다.

4. Reference

- [1] Gordo, Albert, et al. "End-to-end learning of deep visual representations for image retrieval." *International Journal of Computer Vision* 124.2 (2017): 237-254.
- [2] Revaud, Jerome, et al. "Learning with average precision: Training image retrieval with a listwise loss." *Proceedings of the IEEE International Conference on Computer Vision*. 2019.
- [3] Sarlin, Paul-Edouard, et al. "Superglue: Learning feature matching with graph neural networks." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020.