# NAVER LABS Mapping & Localization Challenge

**세종대학교 RCV 연구실 , 조원 ,한대찬**

{jwon,dchan}@rcv.sejong.ac.kr

## 1. 개요

본 챌린지의 인도어 데이터셋 대회는 단일 카메라(휴대용 기기, 로봇에 부착된 카메라)를 이용하여 사용자 혹은 로봇의 현재 위치(6DoF)를 인식하는 것을 목표로 한다. 한번 수집된 지도 영상(이하 DB)에는 각 영상의 절대 좌표정보가 함께 기록되어 있어, DB영상과 현재 위치 영상(이하 QR) 간 상대적인 위치 계산을 통해 QR의 절대좌표를 알 수 있다. 본 대회의 문제 해결을 위해 사용한 SejongRCV-Indoor 팀의 파이프라인은 그림1과 같다. 팀의 알고리즘은 크게 3개의 모듈로 구성되어 있으며, 첫번째 모듈은 QR 영상과 가장 유사한 장소들(이하 Top-K)를 지도 DB 내 에서 찾아내는 "Image Retrieval" 모듈, QR와 가장 기하학적으로 유사한 영상(Top-1)을 Top-K에서 추려내는 "Re-ranking" 모듈. 마지막으로 찾은 Top-1영상과 QR간 2D-3D 대응점관계를 Perspective-n-Point (PnP)로 풀어 QR의 위치와 방향 정보(6DoF)를 추정하는 "Pose Estimation"모듈이 해당된다.

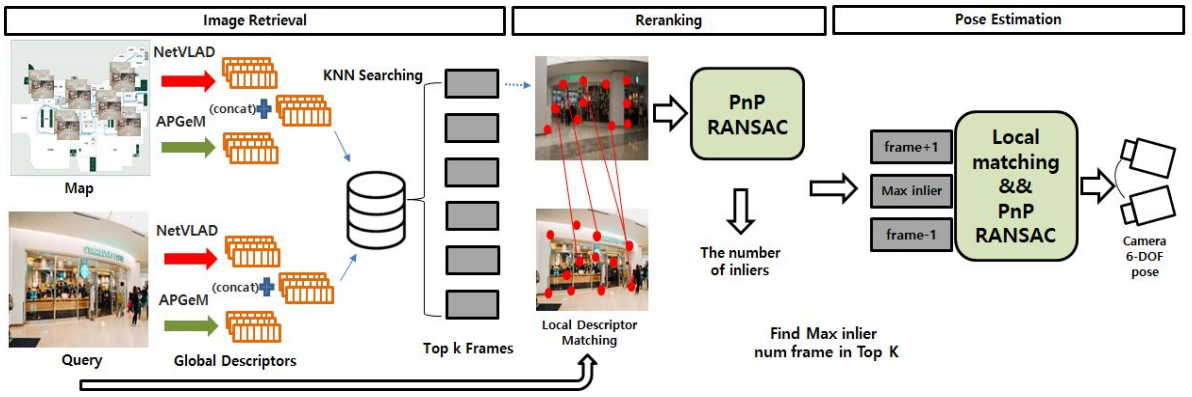


그림 1 SejongRCV indoor 팀의 문제 해결 전체 흐름도.

## 2. 제안하는 방법론

### 2.1 Top-K 후보 선정을 위한 이미지 검색(Image Retrieval)

본 연구진은 NetVLAD[1] 모델과 APGeM[2] 모델을 결합한 앙상블 모델로 영상 검색(Image retrieval)기반 위치 인식 문제를 해결하였다. 이때, NetVLAD는 챌린지에서 제공된 Indoor dataset으로 학습였고, APGeM은 ImageNet과 Google Landmark dataset으로 학습된 Pretrained weight를 사용하였다. 실험에 사용된 NetVLAD와 APGeM 모델의 환경 설정은 아래와 같다.

#### 2.1.1 NetVLAD 모델

NetVLAD 실험을 위해서 Github에 공개된 NetVLAD-pytorch 구현체[1]를 베이스로 사용하였으며, Metric Learning 학습을 위해 제공된 indoor dataset의 학습 영상을 positive와 negative를 나눠 데이터[2]를

[1] https://github.com/lyakaap/NetVLAD-pytorch

[2] 학습에 사용한 positive와 negative 설정은 github repository를 통해 공개 예정

재구성하였다. Positive 기준은 대회의 두번째 평가지표인 0.5m/10° 이내 , negative의 기준은 (카메라 좌표계 기준) 10m 이상으로 정하여 매 batch 마다 랜덤 선택하였다.

Backbone network 로는 VGG16을 선정하였고, D2Net[3]에서 제공한 MegaDepth 데이터로 학습한 파라미터를 전이학습(transfer learning)에 이용하였다. 학습 시, Backbone network를 제공된 Indoor dataset에 fine-tuning 시키기 위해 conv5 layer의 weight 만을 freeze 하지 않았다. 그리고 Triplet margin loss로 NetVLAD의 Cluster를 학습시켰다. 이때, Cluster의 개수는 64로 설정하였고 Triplet margin loss의 margin은 0.1로 설정하였다.

### 2.1.2 APGeM 모델

NetVlad 만으로 부족한 Global Descriptor의 표현력을 늘리기 위해 최근 CVPR2020 Challenge에서 1등을 수상한 APGeM[2]을 추가로 사용하였다. 구현체는 저자가 운영하는 공식사이트 참고 하였고, 본 연구진은 별도의 학습 없이 저자가 공개한 AP loss로 ImageNet과 Google Landmark dataset 각각을 학습시킨 두 개의 pretrained network를 사용하였다

### 2.1.3 앙상블(Ensemble) 모델

NetVLAD를 통해 추출한 Feature에 PCA+Whitening을 적용시킨 Descriptor와, APGeM의 각기 다른 Pretrained model에서 추출한 두 Descriptor를 결합(Concat) 한 후에 정규화(L2 normalize)를 거쳐 Ensemble 형태의 Global Descriptor를 설계 후 사용하였다.

### 2.1.4 영상 검색(Searching) 방법

지도 영상(이하 DB)에서 QR와 유사한 Top-K를 찾기 위해, L2 distance 유사도 비교를 사용하였고, 효율적인 검색을 위해 k-Nearest Neighbors(kNN)의 GPU 버전[3]을 사용하였다. Top-K의 K 값은 Re-ranking 과 pose estimation 실험 결과 가장 성능이 좋은 값(k=10)으로 선정하였다.

## 2.2 Top-1 선정을 위한 후처리

### 2.2.1 Matching Points 추출

파이프라인의 2번째(Re-ranking)와 3번째(Pose Estimation)에서 필요한 DB와 QR의 Matching Points를 추출 하기 위해서 SuperPoint & SuperGlue[4]를 방법을 사용하였으며, 각 모델은 저자가 Github에서 제공하는 Indoor Dataset용 Pretrained 모델과 세부 파라미터를 사용하였다.

### 2.2.2 Re-ranking 을 통한 Top-1 영상 선정

Top-K 에서 QR와 기하학적으로 가장 유사한 영상을 찾아 위치 인식 모듈에 사용하기 위해 Re-ranking 방법론이 사용한다. 본 연구진은 다양한 방법론을 적용 끝에 가장 성능이 좋았던 PnP inlier Re-ranking 방법을 차용하였다. 이는 QR와 Top-K간의 지역불변 특징량 정합 후, PnP 알고리즘을 objective function으로 설정하고, 이를 RANSAC 알고리즘으로 최적화하는 방법론에 해당한다. 최종적으로는 RANSAC 알고리즘에서 나온 inlier의 개수가 가장 많은 영상을 Top-1 영상으로 선정한다.

## 2.3 영상 기반 3D 위치 인식 방법론

### 2.3.1 DB 데이터(RGB-Lidar) 전처리

제공된 DB의 Point Cloud Map을 활용하여 DB 영상(RGB)에 대응되는 (Pseudo) Lidar 데이터를 가공하였다. 가공을 위해 먼저 실내 Map 정보가 담겨진 PCD 파일의 모든 point cloud에서 DB 영상의 위치 정보 기준 반경 15m 내의 point cloud를 모두 찾고(octree 검색 기법), 전 방향 point cloud에서 DB 영상(RGB) 화각 내의 3차원 정보만을 찾아 필터링 하였다. 2차 필터링 된 최종 포인트들은 좀 더 빠른 연산 속도를 위해 numpy pickle 로

[3] https://github.com/facebookresearch/faiss

가공하여 저장 후 사용하였다.

### 2.3.2 QR 데이터(RGB) 전처리

제공된 모든 DB 영상은 렌즈의 왜곡이 제거된 undistorted 영상이었으나, QR 영상의 경우 일부 렌즈 왜곡이 제거되지 않은 distorted 영상이 포함되어 있었다. 이런 점은 본 연구진의 알고리즘 수행 시 3차원 위치 정보 추정에 오차를 야기하게 되며, 이런 문제는 카메라 캘리브레이션 파라미터를 통해 해결이 가능했다. 분석을 통해 Basler 카메라로 획득한 QR 영상의 왜곡이 제거되지 않고 제공된 것을 확인하였으며, 본 연구진은 문제가 되는 QR 영상의 왜곡을 제거하고 Pose estimation 과 Re-ranking을 진행하였다.

### 2.3.3 DB-QR 영상 간 Pose estimation

Outlier에 강인한 Pose Estimation 을 위해 Top-1 의 timestamp상 앞, 뒤 프레임을 추가적으로 사용하여 총 3장의 후보 영상(Candidates)과 QR영상을 비교하였다. Candidates과 QR 영상간 SuperPoint를 추출하여 SuperGlue로 매칭시킨 후, 가공해 놓은 해당 Candidates의 3D points를 해당 영상에 projection 시켜 2D point (projected 2D point)를 구한다. 그리고 매칭된 Candidates 의 local point와 L2 distance가 가장 작은 projected 2D point의 3D point를 QR 영상 local point의 3D point로 할당하였다. 그리고 매칭된 QR의 local point에도 각각 같은 3D point를 할당한 뒤, 할당된 3D point를 RANSAC으로 최적화한 PnP 알고리즘에 사용해 QR 영상의 6-DOF pose를 추정하였다.

## 3. 요약

본 챌린지를 위해 사용된 코드와 각종 설정 파라미터는 SejongRCV GitHub[4]에 공개하였다.

## 4. Reference


[1] Arandjelovic, Relja, et al. "NetVLAD: CNN architecture for weakly supervised place recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[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] Dusmanu, Mihai, et al. "D2-net: A trainable cnn for joint detection and description of local features." *arXiv preprint arXiv:1905.03561* (2019).

[4] 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.

[5] Dai, Angela, et al. "Scannet: Richly-annotated 3d reconstructions of indoor scenes." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.


[4] https://github.com/sejong-rcv/SejongRCV-Indoor