Visual place recognition using vision foundation model

2024.07.05 여름 세미나

Sogang University
Vision & Display Systems Lab, Dept. of Electronic Engineering

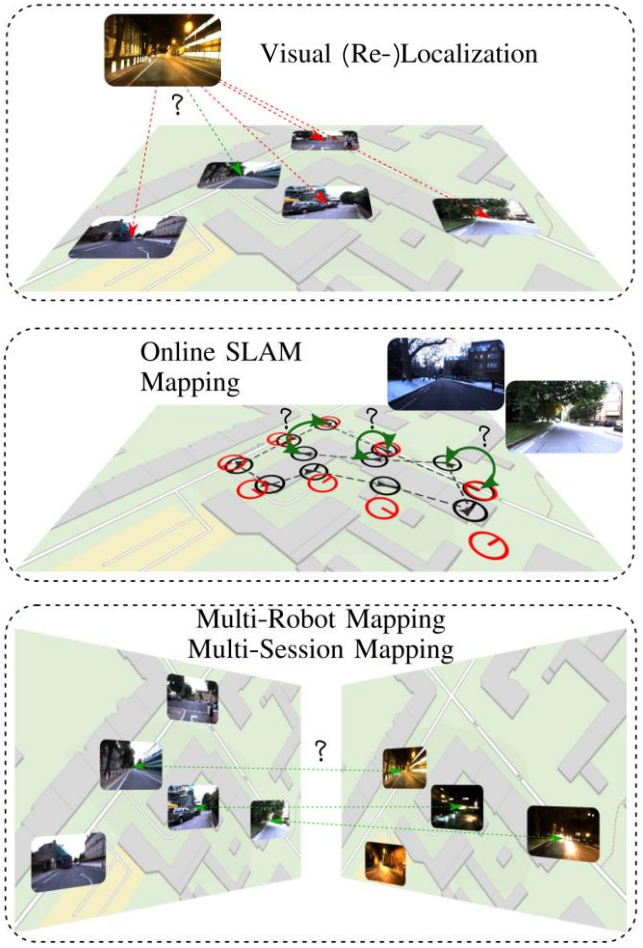
Presented By
김동규

Outline

- Background
  - What is Visual Place Recognition?
  - Self-supervised foundation models that extract task-agnostic visual features
- Paper
  - Towards Seamless Adaptation of Pre-trained Models for Visual Place Recognition (ICLR 24)
  - CricaVPR: Cross-image Correlation-aware Representation Learning for Visual Place Recognition (CVPR 2024)
  - Optimal Transport Aggregation for Visual Place Recognition (CVPR 24)

Background

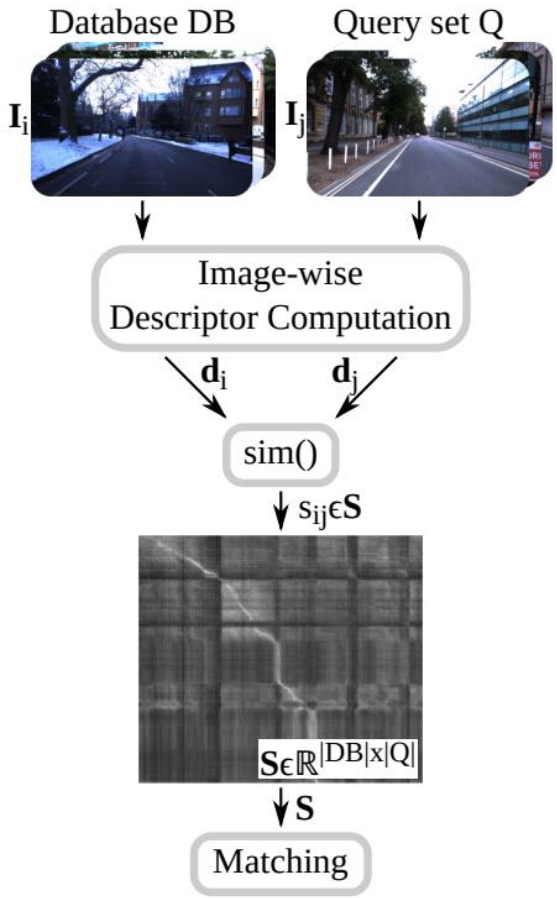
- What is Visual Place Recognition (VPR)?
  - Image를 이용하여 장소를 인식하는 기술
  - Use cases
    - Visual (Re-)Localization
    - Online SLAM Mapping
    - Multi-Robot mapping
    - Multi-Session Mapping
    - …
  - **Standard VPR**
    - Image 정보를 이용하여 database에서 가장 유사한 image를 탐색



< Overview of Visual Place Recognition use cases[1] >

Background

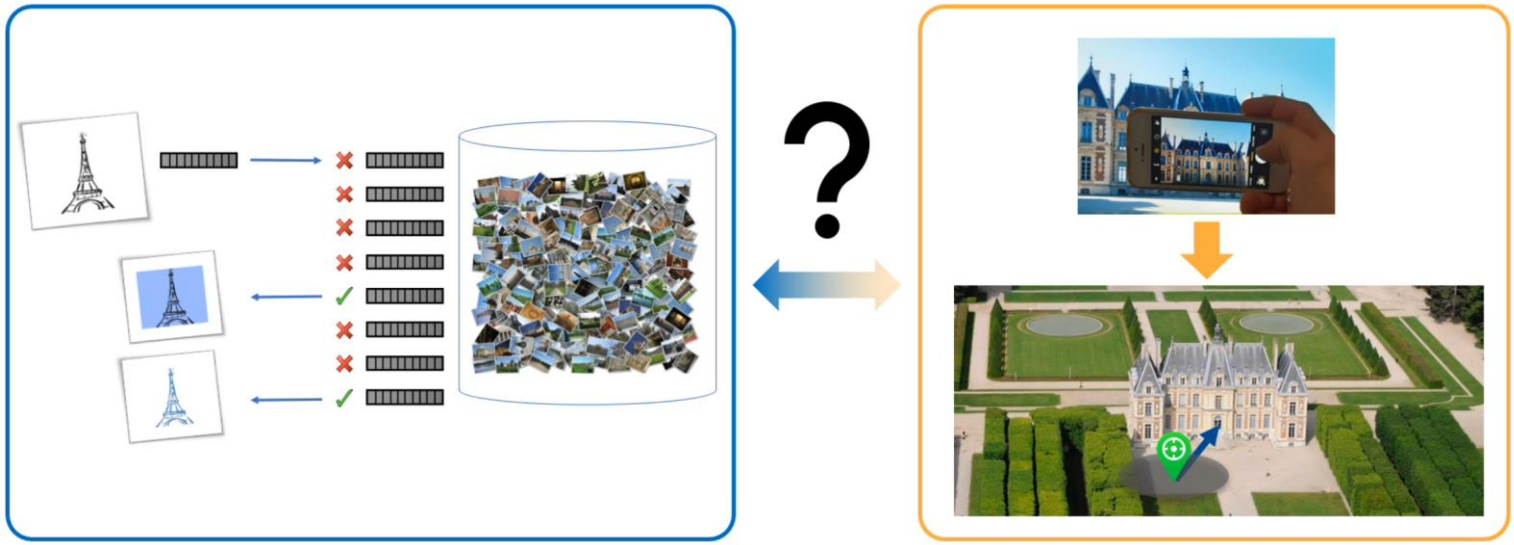
- What is Visual Place Recognition (VPR)?
  - 학습 과정
    1. Feature Extraction
       - Image data에서 유의미한 local feature extraction
    2. Place Descriptor Generation
       - Local features를 aggregation 하여 global descriptor를 생성
    3. Matching
       - Inference 시 query image의 global descriptor 를 생성하여 database안의 global descriptor 와 비교
  - 학습 목표
    - Image를 나타내는 고품질의 global descriptor를 생성
    - (Optional) Re-ranking 방법



< Pipeline of standard VPR[1)] >

Background

- What is Visual Place Recognition (VPR)?
  - Image Retrieval and Visual Place Recognition
    - Global descriptor, aggregation 방법 등 Image Retrieval 에서 파생된 기술이 존재
    - 해당 object 에 주목, 현재 위치에 주목
    - Dataset, 확장성 문제



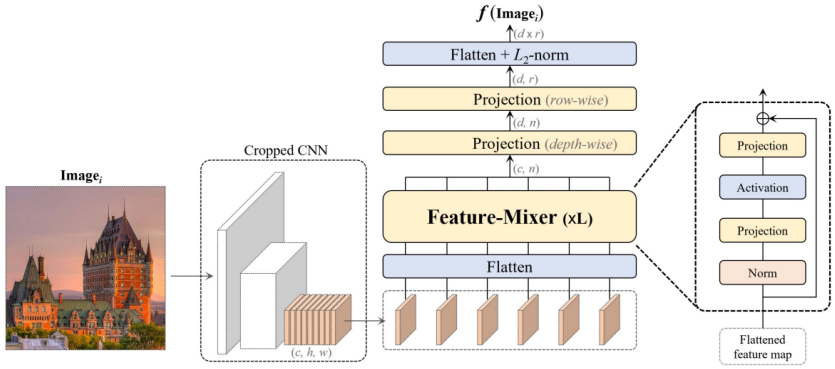
< Image Retrieval and Visual Place Recognition[1)] >

1) Ali-Bey, A., Chaib-Draa, B., & Giguere, P. (2023). Mixvpr: Feature mixing for visual place recognition. *In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 2998-3007).
2) Zhu, S., Yang, L., Chen, C., Shah, M., Shen, X., & Wang, H. (2023). R2former: Unified retrieval and reranking transformer for place recognition. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 19370-19380).

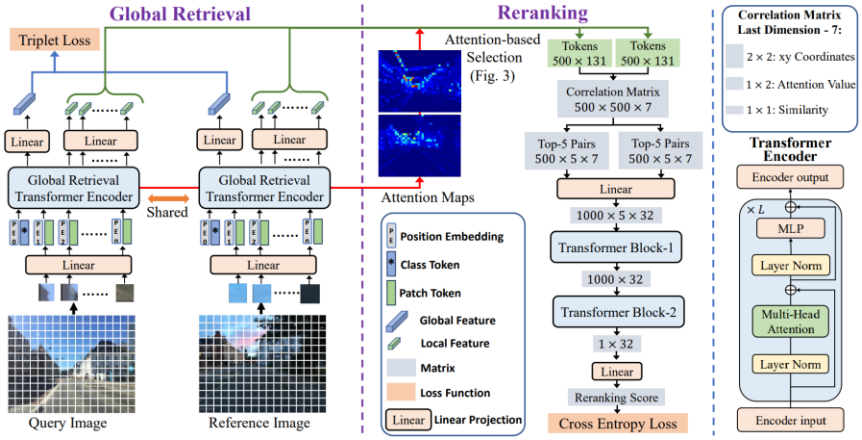
Background

- What is Visual Place Recognition (VPR)?
  - 전통 VPR
    - SIFT, SURF와 같은 local feature matching 을 이용
    - 대규모 dataset에 부적합
  - 딥러닝 활용 VPR
    - CNN과 ViT를 이용하여 다양한 방법이 등장
    - 날씨, 낮과 밤, 계절, view-point 변화 등의 환경 변화등에 초점

해당 모델들도 **범용성 부족**



< CNN based model : MixVPR[1] >

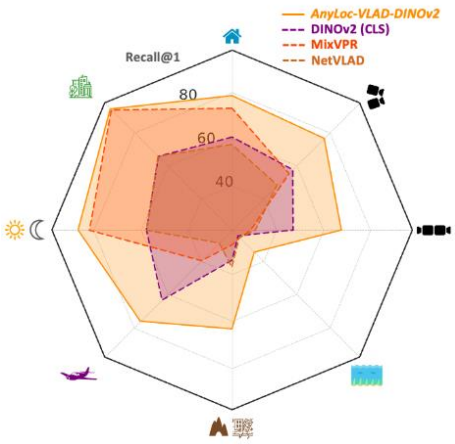


< ViT based model : R2Former[2] >

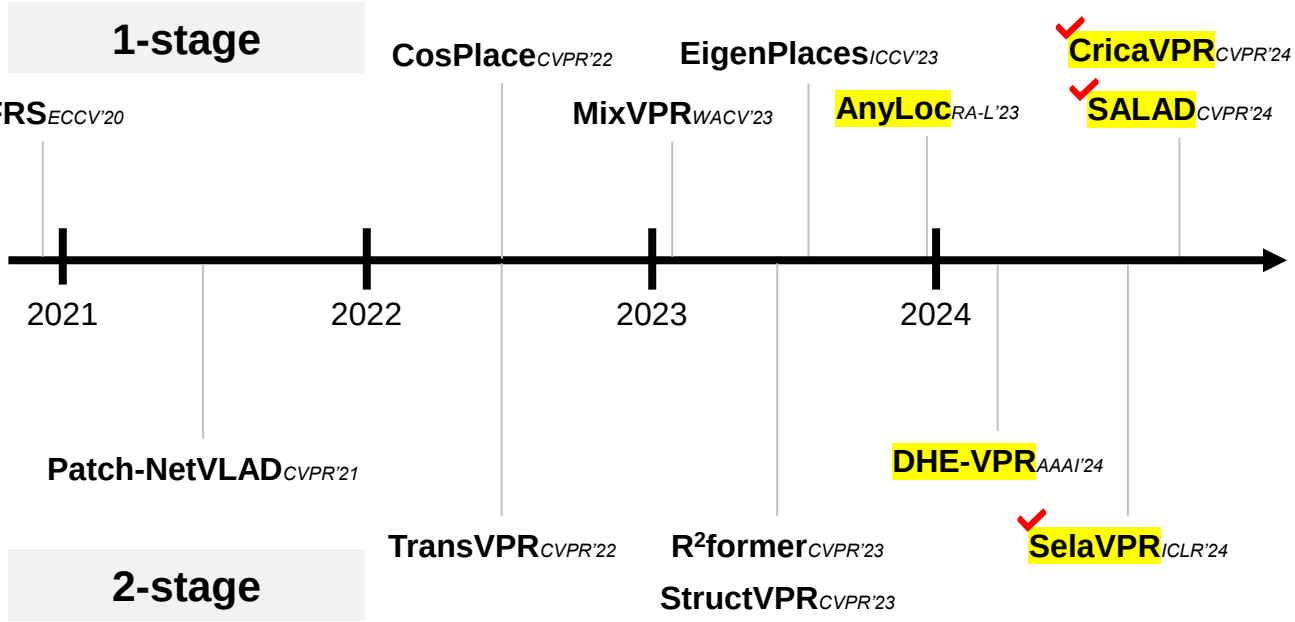
1) Keetha, Nikhil, et al. "Anyloc: Towards universal visual place recognition." *IEEE Robotics and Automation Letters* (2023).

Background

- Self-supervised foundation models that extract task-agnostic visual features
  - CLIP, DINO과 같은 visual foundation model이 등장
  - VPR에서도 visual foundation model을 사용한 방법 등장
    - AnyLoc에서 DINOv2 의 VPR에서의 사용가능성을 검증
      - DINOv2를 통해 Pre-trained 없이 다양한 환경에서 적용가능한 범용적인 solution 제시



< AnyLoc Performance[1] >



Lu, Feng, et al. “Towards Seamless Adaptation of Pre-trained Models for Visual Place Recognition” The Twelfth International Conference on Learning Representations (ICLR), 2024.

Introduction

- Background
  - Foundation model의 등장으로 generalization이 잘된 model이 높은 성능을 보임
    - 이러한 model들은 동적 객체 에 취약, 일부 정적 배경을 무시

(a) Input image

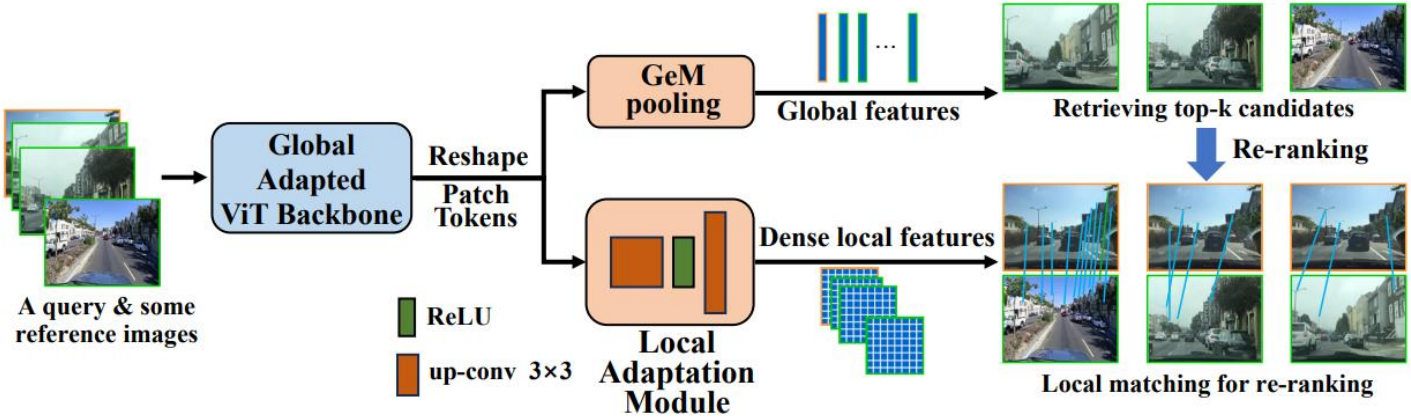
(b) Result of pre-trained model

(c) Result of our method

    - Pre-trained 된 모델과 목표 model 간의 간극이 존재
    - 하지만 fine-tunning 을 사용하면 기존에 학습된 지식을 잊어버릴 수 있음
- Contribution
  - Pre-trained model을 VPR에 적용시키기 위한 Seamless한 adaptation method
  - Global 및 local feature를 효과적으로 추출하는 hybrid 한 adaptation method
  - Stage 2 에서 효과적인 re-ranking 을 위한 loss function 추가

Methodology

- Overview
  - Global adaptation (Stage 1)
    - Global adapted ViT backbone
    - GeM (Generalized Mean) pooling
    - Retrieving top-k candidates
  - Local adaptation (Stage 2)
    - Local adaptation module
    - Local matching for re-ranking



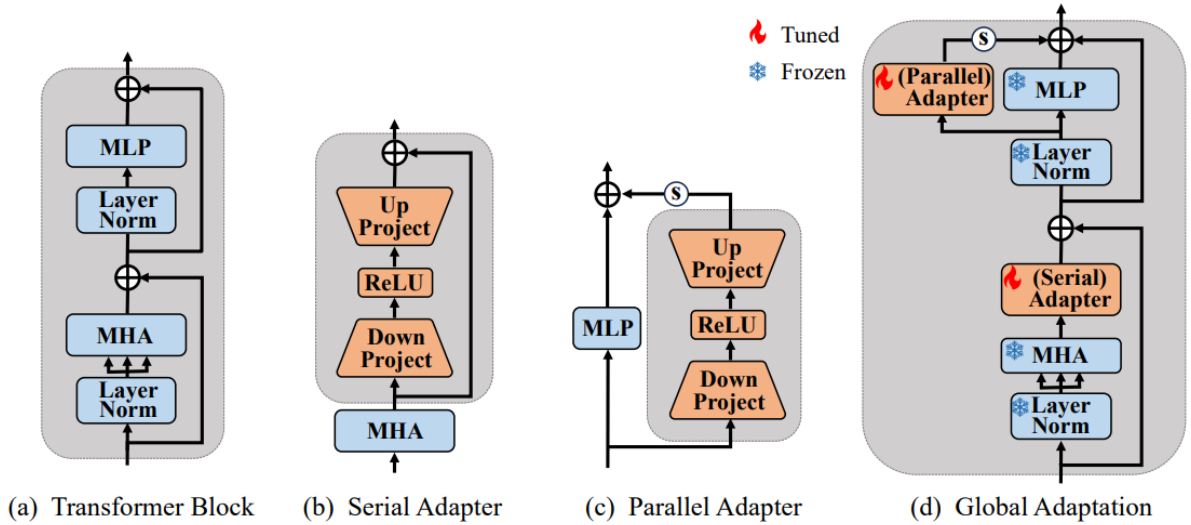
< SelaVPR의 pipeline >

Methodology

- Global Adaptation
  - 각 transformer block에 두개의 adapter 추가
    - Serial adapter
      - MHA layer 이후에 추가하며, skip connection 포함
    - Parallel adapter
      - MLP layer와 병렬로 연결, scale factor s 추가

$$x_l' = \mathrm{MHA}(\mathrm{LN}(x_{l-1})) + x_{l-1}$$
$$x_l = \mathrm{MLP}(\mathrm{LN}(x_l')) + x_l',$$

$$x_l' = \mathrm{Adapter1}(\mathrm{MHA}(\mathrm{LN}(x_{l-1}))) + x_{l-1}$$
$$x_l = \mathrm{MLP}(\mathrm{LN}(x_l')) + s \cdot \mathrm{Adapter2}(\mathrm{LN}(x_l')) + x_l'.$$



< SelaVPR의 global adapter >

- GeM pooling 적용

Methodology

- Local Adaptation
  - Stage-1 에서 top-k 의 feature 에 대하여 2번의 up-conv을 실행

$$\boldsymbol{f}^l = \text{LocalAdaptation}(fm) = \text{intraL2}(\text{up-conv2}(\text{ReLU}(\text{up-conv1}(fm))))$$

  - Feature 간의 cosine similarity를 이용하여 유사도 측정

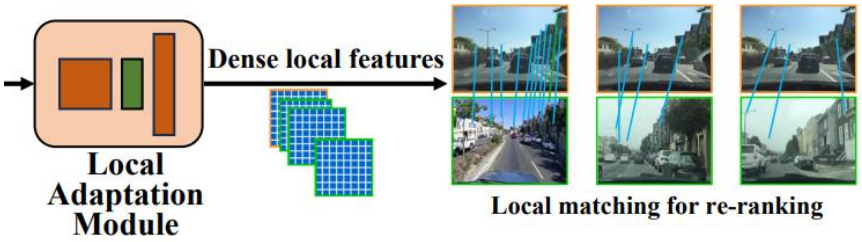
$$\boldsymbol{s}_{qc}(i,j) = \boldsymbol{f}_q^l(i) \cdot \boldsymbol{f}_c^l(j) \quad i,j \in \{1,2,...,N'\} \quad (N' = 61 \times 61)$$

Query image q의 i 번째 local feature
Candidate image c의 j 번째 local feature

  - Mutual Nearest Neighbor Matches

$$\mathcal{M} = \{(u,v) : \; u = \arg\max_i \boldsymbol{s}_{qc}(i,v), \; v = \arg\max_j \boldsymbol{s}_{qc}(u,j)\}.$$

    - query image의 u번째 feature와 candidate image의 v번째 feature가 서로에게 nearest neighbor일 때, 이 두 특징을 매칭
    - Matching된 feature 수가 많다면
      - 같은 장소일 확률이 높음



< SelaVPR의 stage 2 >

Methodology

- Loss
  - Global loss (stage 1)
    - Triplet loss

q: query
p: positive
n: negative

$$L_g = \sum_j l(\| \boldsymbol{f}_q^g - \boldsymbol{f}_p^g \| + m - \| \boldsymbol{f}_q^g - \boldsymbol{f}_{n_j}^g \|)$$

l: hinge loss margin

  - Local loss (stage 2)
    - Mutual Nearest Neighbor local feature loss

$$L_l = \sum_j l(-\frac{\sum_{(u,v)\in\mathcal{M}} \boldsymbol{s}_{qp}(u,v)}{|\mathcal{M}|} + \frac{\sum_{(u',v')\in\mathcal{M}'} \boldsymbol{s}_{qn_j}(u',v')}{|\mathcal{M}'|})$$

  - Total loss

$$L = L_g + \lambda L_l$$

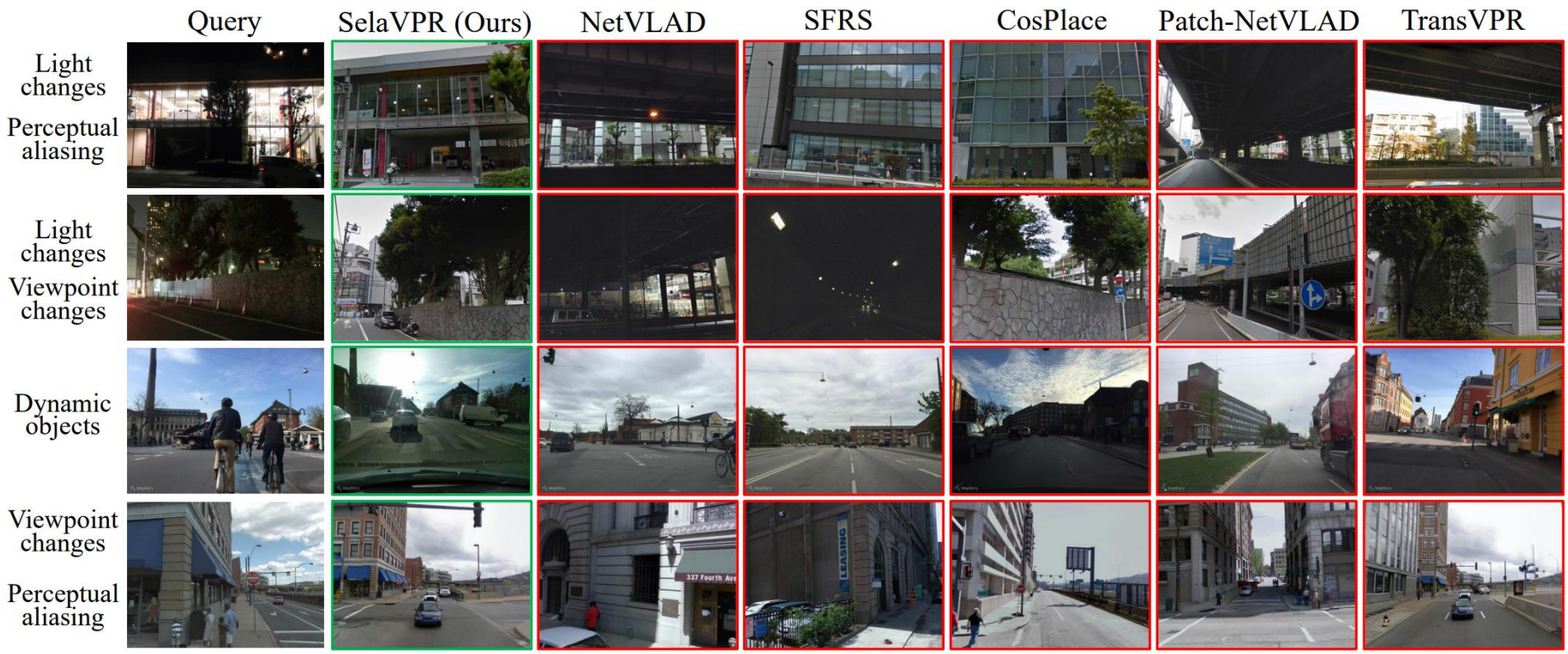
Experiments

- Quantitative Results

	Method	Tokyo24/7			MSLS-val			MSLS-challenge			Pitts30k-test		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
1 stage	NetVLAD	60.6	68.9	74.6	53.1	66.5	71.1	35.1	47.4	51.7	81.9	91.2	93.7
	SFRS	81.0	88.3	92.4	69.2	80.3	83.1	41.6	52.0	56.3	89.4	94.7	95.9
	CosPlace	81.9	90.2	92.7	82.8	89.7	92.0	61.4	72.0	76.6	88.4	94.5	95.7
	MixVPR	85.1	91.7	94.3	88.0	92.7	94.6	64.0	75.9	80.6	91.5	95.5	96.3
	SelaVPR(global)	81.9	94.9	96.5	87.7	95.8	96.6	69.6	86.9	90.1	90.2	96.1	97.1
2 stage	SP-SuperGlue	88.2	90.2	90.2	78.1	81.9	84.3	50.6	56.9	58.3	87.2	94.8	96.4
	Patch-NetVLAD-s	78.1	83.8	87.0	77.8	84.3	86.5	48.1	59.4	62.3	87.5	94.5	96.0
	Patch-NetVLAD-p	86.0	88.6	90.5	79.5	86.2	87.7	48.1	57.6	60.5	88.7	94.5	95.9
	TransVPR	79.0	82.2	85.1	86.8	91.2	92.4	63.9	74.0	77.5	89.0	94.9	96.2
	StructVPR	-	-	-	88.4	94.3	95.0	69.4	81.5	85.6	90.3	96.0	97.3
	R^2Former	88.6	91.4	91.7	89.7	95.0	96.2	73.0	85.9	88.8	91.1	95.2	96.3
	SelaVPR (ours)	**94.0**	**96.8**	**97.5**	**90.8**	**96.4**	**97.2**	**73.5**	**87.5**	**90.6**	**92.8**	**96.8**	**97.7**

Experiments

- Qualitative Results



Experiments

- Ablation study

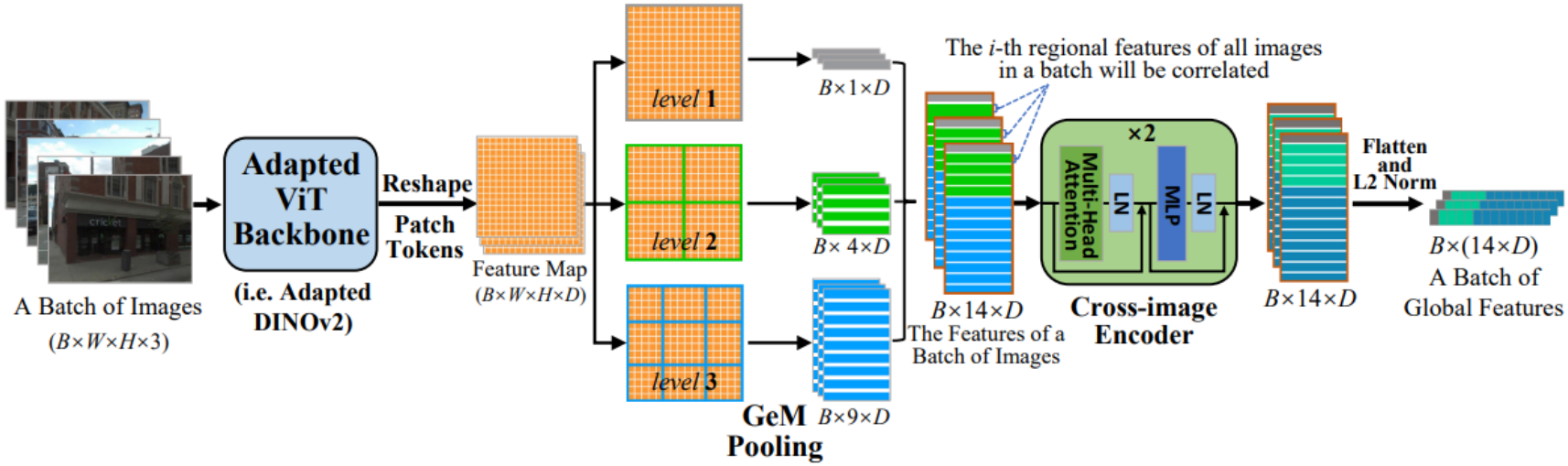
Ablated version	Pitts30k-test			Tokyo24/7			MSLS-val		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
DINOv2-GeM	81.3	91.0	93.8	67.3	85.1	89.8	44.7	55.9	59.3
Tuned-DINOv2-GeM	85.3	92.7	94.7	65.7	78.1	83.8	79.7	90.3	92.2
Global-Adaptation	87.3	94.6	96.6	77.8	87.6	91.7	87.4	95.9	96.9
Local-Adaptation	87.2	93.9	96.1	87.6	94.6	**95.9**	67.2	75.0	76.6
SelaVPR	**91.4**	**96.5**	**97.8**	**93.3**	**95.2**	95.6	**90.8**	**96.4**	**97.2**

- DINOv2-GeM
  - 일반화 성능이 낮음
- Tuned-DINOv2-GeM
  - Tokyo24/7에서는 성능이 저하
- Global-Adaptation
  - 모든 데이터셋에서 성능이 향상
- Local-Adaptation
  - 특히 조명 변화가 큰 Tokyo24/7에서 높은 성능 향상

Lu, Feng, et al. “CricaVPR: Cross-image Correlation-aware Representation Learning for Visual Place Recognition” The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024.

Methodology

- Overview
  - Adapted ViT backbone
    - Multi-scale Convolution-enhanced adaptation
  - Cross-image Correlation-aware Place Representation
    - 3 levels split feature maps
    - Cross-image encoder

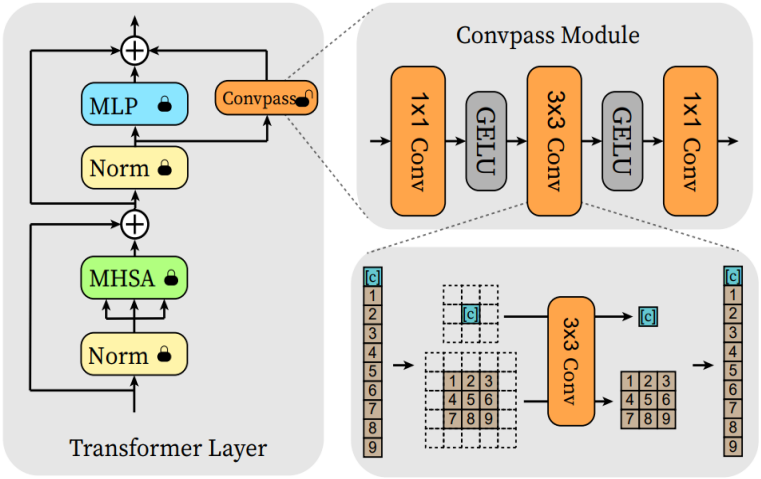


< CricaVPR의 pipeline >

1) Shibo Jie and Zhi-Hong Deng. Convolutional bypasses are better vision transformer adapters. *arXiv preprint arXiv:2207.07039*, 2022.

Methodology

- Multi-scale Convolution-enhanced Adaptation
  - Convpass[1] (arXiv 2022)에서 제안한 convolution을 사용한 adapter
    - Convolutional layer를 사용하여 image token과 class token에 convolution 수행
      - Convolutional layers의 고유한 지역성 특성을 활용
    - 약 0.33M 의 parameters (ViT-B parameters : 86M)
    - Multi head self-attention 또는 MLP block 과 병렬로 삽입되는 것이 가장 좋은 성능을 보임

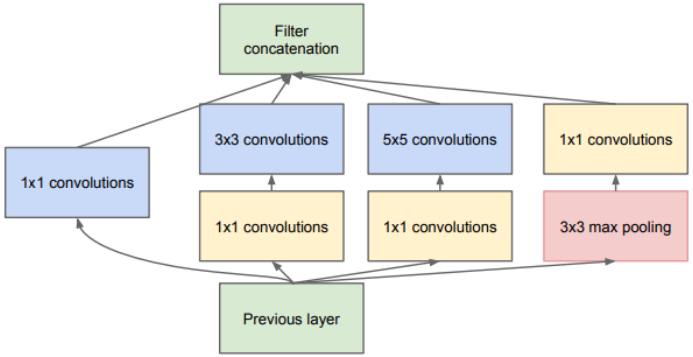


< Convpass의 convpass adapter[1] >

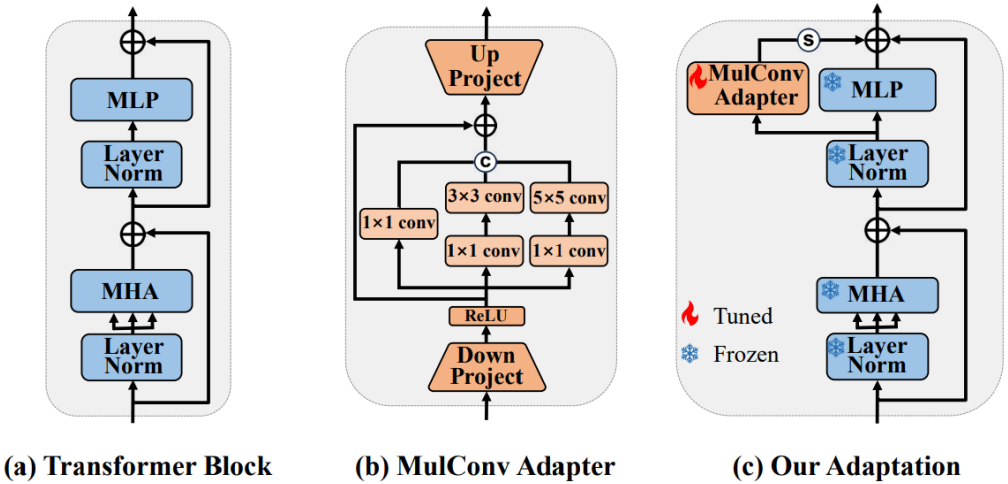
Methodology

- Multi-scale Convolution-enhanced Adaptation
  - GoogLeNet[1]의 inception module 방법을 모방하여 MulConv adapter 구축
    - 차원을 줄이기 위해 1x1 conv 추가

$$z_l' = \mathrm{MHA}\left(\mathrm{LN}\left(z_{l-1}\right)\right) + z_{l-1},$$
$$z_l = \mathrm{MLP}\left(\mathrm{LN}\left(z_l'\right)\right) + s \cdot \mathrm{Adapter}\left(\mathrm{LN}\left(z_l'\right)\right) + z_l'.$$



< GoogLeNet의 inception module[1] >

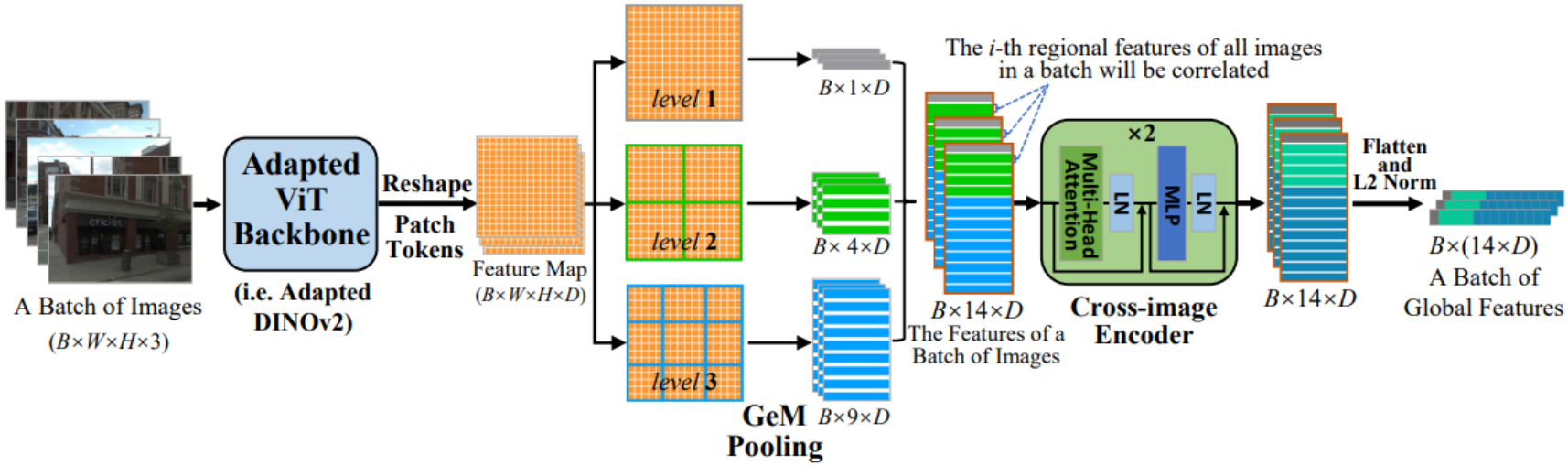


(a) Transformer Block (b) MulConv Adapter (c) Our Adaptation

< CricaVPR의 multi-scale convolution-enhanced adapter >

Methodology

- Cross-image Correlation-aware Place Representation
  - Feature map을 1x1, 2x2, 3x3 세 level 로 나누어서 split 후 GeM Pooling
    - 이때, 1x1 은 global token을 ViT의 class token 을 그대로 사용
    - 총 14개의 embedding vector 를 concatenation
  - Cross-image Encoder
    - Batch 내의 모든 image의 i 번째 local feature 간의 correlation 을 modeling
  - 최종 global descriptor 생성

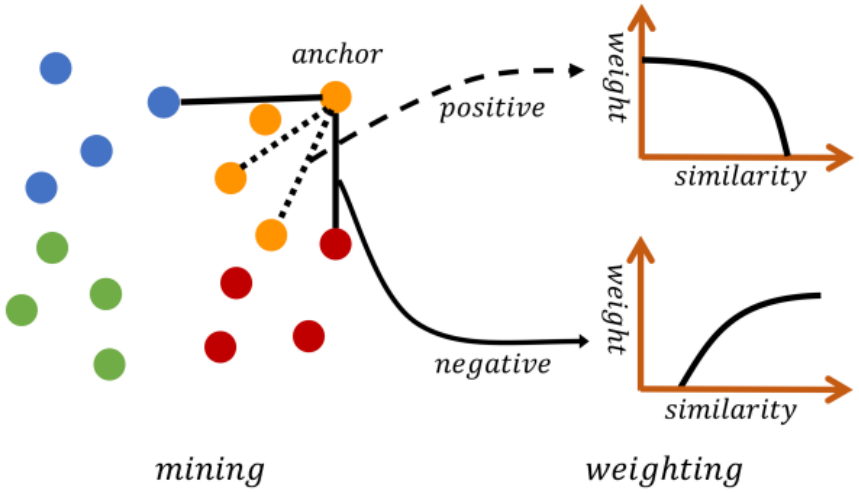


< CricaVPR의 pipeline >

Methodology

- Loss
  - Multi-Similarity (MS) loss
    - 이미지 검색 및 분류에서 주로 사용
    - Mining을 통해 유의미한 페어를 선택
      - 이때, hard sample 을 선택하는 hard mining을 진행
    - 유사한 data는 가깝게 다른 data는 멀도록 학습

$$\mathcal{L}_{MS} = \frac{1}{B}\sum_{q=1}^{B}\left\{\frac{1}{\alpha}\log\left[1+\sum_{p\in\mathcal{P}_q} e^{-\alpha(S_{qp}-\lambda)}\right] + \frac{1}{\beta}\log\left[1+\sum_{n\in\mathcal{N}_q} e^{\beta(S_{qn}-\lambda)}\right]\right\},$$



< MS loss의 방법[1)] >

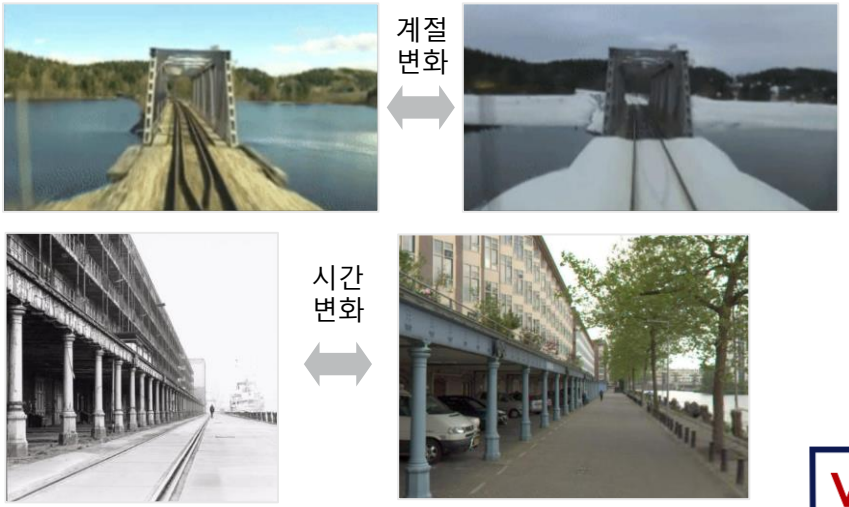
Experiments

- Quantitative Results
  - Benchmark datasets

Method	Dim	Pitts30k			Tokyo24/7			MSLS-val			MSLS-challenge		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
NetVLAD [5]	32768	81.9	91.2	93.7	60.6	68.9	74.6	53.1	66.5	71.1	35.1	47.4	51.7
SFRS [23]	4096	89.4	94.7	95.9	81.0	88.3	92.4	69.2	80.3	83.1	41.6	52.0	56.3
Patch-NetVLAD [26]	/	88.7	94.5	95.9	86.0	88.6	90.5	79.5	86.2	87.7	48.1	57.6	60.5
TransVPR [59]	/	89.0	94.9	96.2	79.0	82.2	85.1	86.8	91.2	92.4	63.9	74.0	77.5
CosPlace [8]	512	88.4	94.5	95.7	81.9	90.2	92.7	82.8	89.7	92.0	61.4	72.0	76.6
GCL [38]	2048	80.7	91.5	93.9	69.5	81.0	85.1	79.5	88.1	90.1	57.9	70.7	75.7
MixVPR [2]	4096	91.5	95.5	96.3	85.1	91.7	94.3	88.0	92.7	94.6	64.0	75.9	80.6
EigenPlaces [10]	2048	92.5	96.8	97.6	**93.0**	96.2	97.5	89.1	93.8	95.0	67.4	77.1	81.7
CricaVPR (ours)	4096	**94.9**	**97.3**	**98.2**	**93.0**	**97.5**	**98.1**	**90.0**	**95.4**	**96.4**	**69.0**	**82.1**	**85.7**

  - More challenging datasets

Method	Nordland	Amster Time	SVOX -Night	SVOX -Rain
SFRS [23]	16.0	29.7	28.6	69.7
CosPlace [8]	58.5	38.7	44.8	85.2
MixVPR [2]	76.2	40.2	64.4	91.5
EigenPlaces [10]	71.2	48.9	58.9	90.0
CricaVPR (ours)	**90.7**	**64.7**	**85.1**	**95.0**



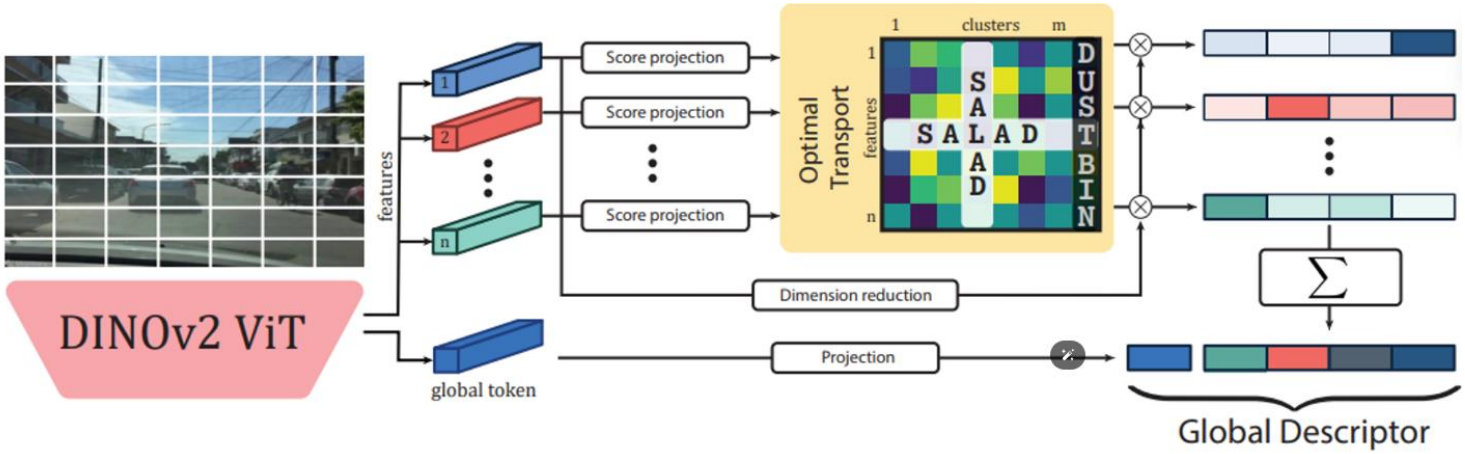
Experiments

- Qualitative Results

Izquierdo, Civera. “Optimal Transport Aggregation for Visual Place Recognition” The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024.

Methodology

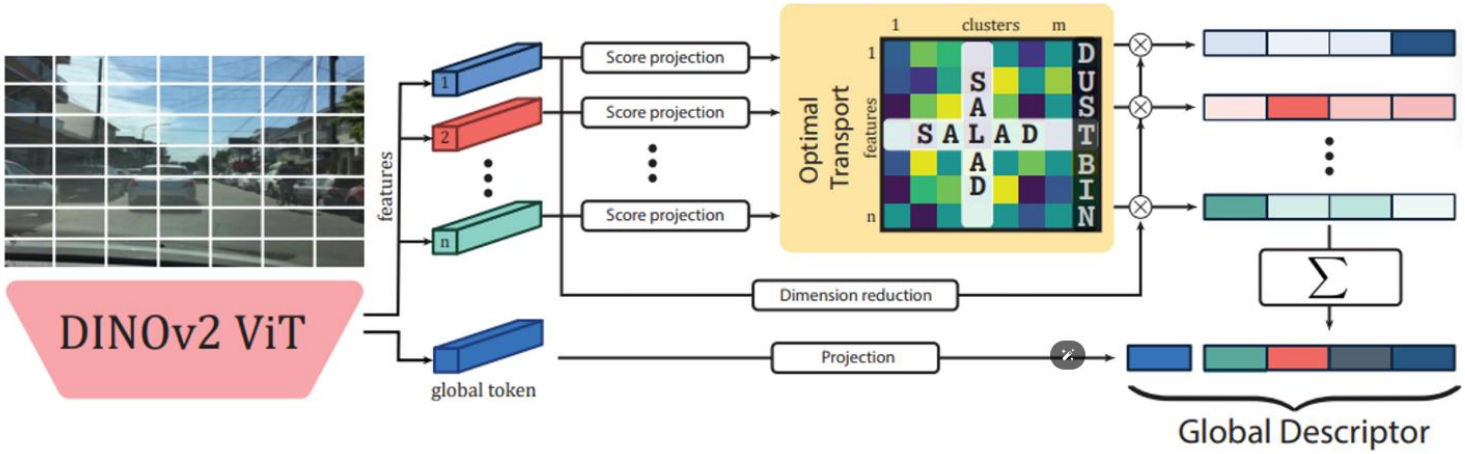
- Overview
  - Local Feature Extraction
  - Assignment & Aggregation
    - SALAD (Sinkhorn Algorithm for Locally Aggregated Descriptors)
      - Optimal assignment transport



< SALAD의 pipeline >

Methodology

- Local Feature Extraction
  - Pre-trained DINOv2 를 backbone 으로 사용
  - Fine-tunning
    - DINOv2 의 저자들은 fine-tunning 이 성능을 저하시킨다고 주장
    - VPR task에 한해서 성능개선
      - Encoder 의 마지막 block 을 unfreeze



< SALAD의 pipeline >

Methodology

- Assignment & Aggregation
  - NetVLAD[1)]
    - i 번째 Feature 가 j 번째 cluster에 assign 될 확률 a
    - 해당 cluster 의 residual vector
    - Global descriptor V 생성

xi : local feature

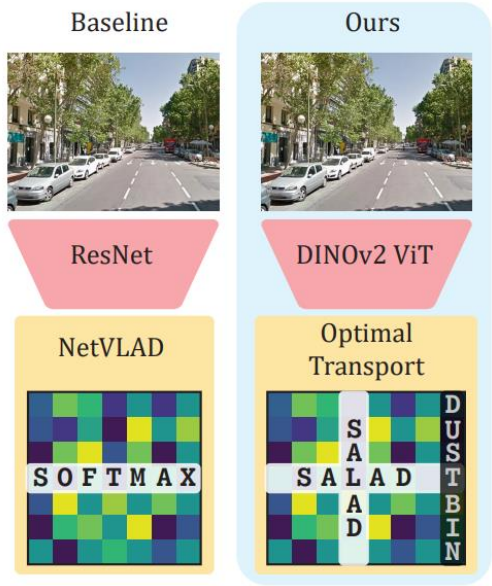
$$a_{ij} = \frac{e^{w_j^T x_i + b_j}}{\sum_{k=1}^{K} e^{w_k^T x_i + b_k}}$$

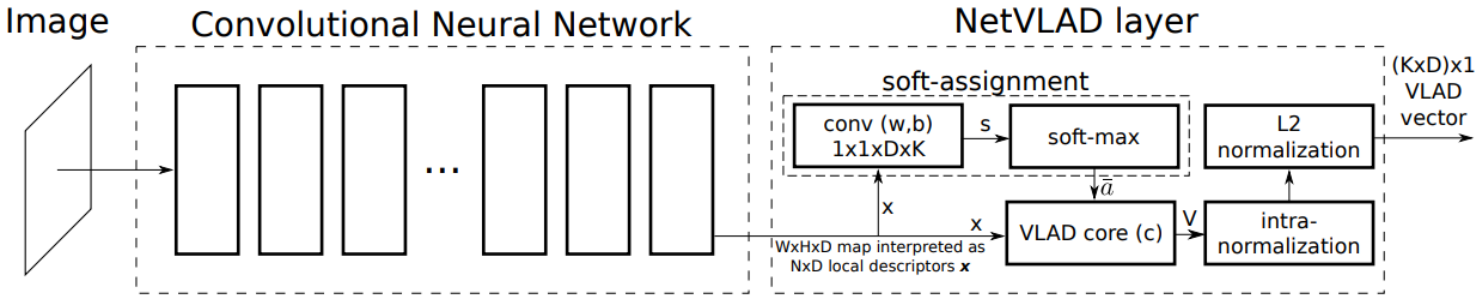
cj : cluster j 의 중심

$$v_j = \sum_{i=1}^{N} a_{ij}(x_i - c_j)$$

$$V = [v_1^T, v_2^T, \ldots, v_K^T]^T$$

concat

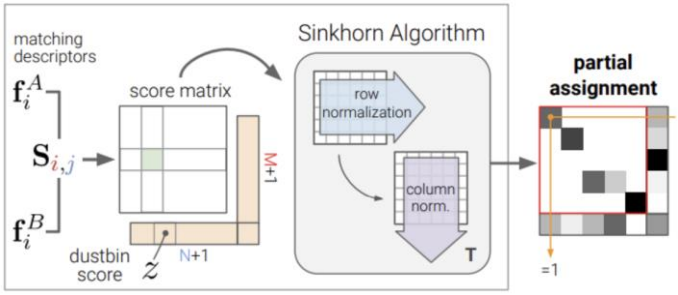




< NETVLAD의 pipeline[1)] >

Methodology

- Assignment & Aggregation
  - SALAD
    - Score matrix 생성
    - dustbin 추가
    - Optimal assignment
    - Aggregation
    - Global descriptor 생성



< Superglue의 optimal matching method[1] >

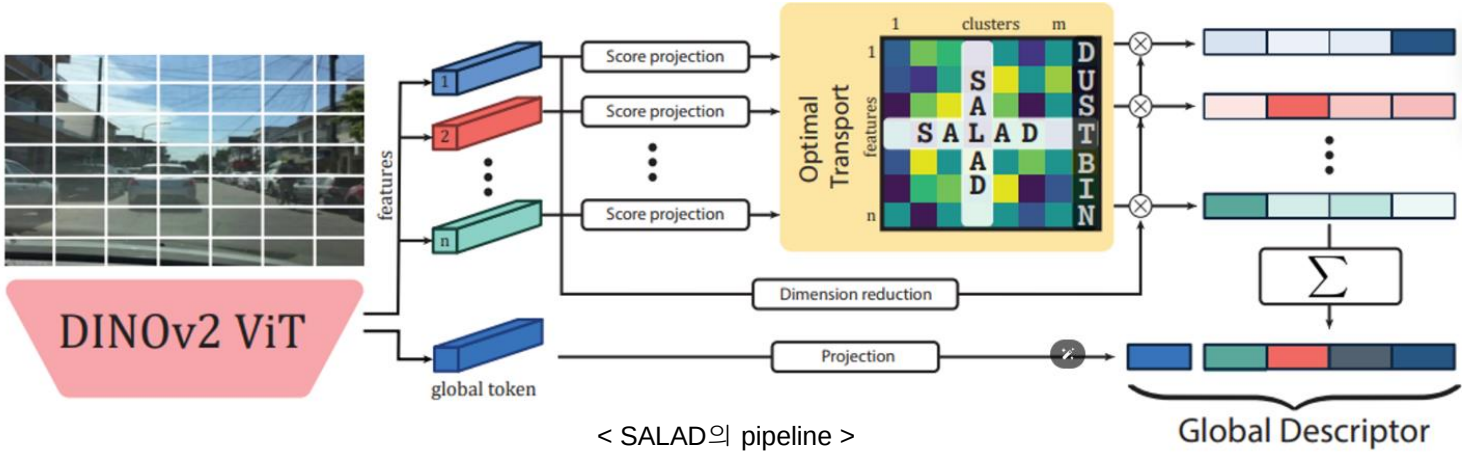
$$\mathbf{S} \in \mathbb{R}^{n \times m}_{>0}$$

$$\bar{\mathbf{S}} = [\mathbf{S},\ \bar{\mathbf{s}}_{i,m+1}] \in \mathbb{R}^{n \times m+1}_{>0}$$

$$\mathbf{P} = \mathrm{Sinkhorn}(\exp(\bar{\mathbf{S}}))$$

$$V_{j,k} = \sum_{i=1}^{n} P_{i,k} \cdot f_{i,k}$$

$$\mathbf{D} = [\mathbf{V}, \mathbf{g}]$$ concat



< SALAD의 pipeline >

Experiments

- Quantitative Results

Method	Desc. size	Latency (ms)	MSLS Challenge		MSLS Val		NordLand		Pitts250k-test		SPED	
			R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5
NetVLAD [4]	32768	1.41	35.1	47.4	82.6	89.6	32.6	47.1	90.5	96.2	78.7	88.3
GeM [45]†	1024	1.14	49.7	64.2	78.2	86.6	21.6	37.3	87.0	94.4	66.7	83.4
Conv-AP [1]	8192	1.22	54.2	66.6	83.1	90.3	42.7	58.9	92.9	97.7	79.2	88.6
CosPlace [5]	2048	2.59	67.2	78.0	87.4	93.0	44.2	59.7	92.1	97.5	80.1	89.6
MixVPR [2]	4096	1.37	64.0	75.9	88.0	92.7	58.4	74.6	94.6	98.3	85.2	92.1
EigenPlaces [6]	2048	2.65	67.4	77.1	89.3	93.7	54.4	68.8	94.1	98.0	69.9	82.9
DINOv2 SALAD	512 + 32	2.33	70.8	83.6	89.3	94.9	61.2	78.9	93.0	97.4	88.5	94.7
DINOv2 SALAD	2048 + 64	2.35	73.7	85.9	90.5	95.4	70.4	85.7	94.8	98.3	89.5	94.9
DINOv2 SALAD	8192 + 256	2.41	**75.0**	**88.8**	**92.2**	**96.4**	**76.0**	**89.2**	**95.1**	**98.5**	**92.1**	**96.2**

- Ablation study

Method	Desc. size	MSLS Challenge			MSLS Val			NordLand			Pitts250k-test			SPED		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
ResNet NetVLAD [4]	32768	35.1	47.4	51.7	82.6	89.6	92.0	32.6	47.1	53.3	90.5	96.2	97.4	78.7	88.3	91.4
DINOv2 AnyLoc [28]	49152	42.2	53.5	58.1	68.7	78.2	81.8	16.1	25.4	30.4	87.2	94.4	96.5	85.3	94.4	95.4
ResNet SALAD	8192	57.4	70.8	74.9	83.2	89.5	91.8	33.3	49.6	55.8	91.4	96.9	97.9	75.0	86.7	89.8
ConvNext [34] SALAD	8192	63.9	75.2	80.1	85.5	92.4	94.5	47.8	64.3	70.3	93.9	97.9	98.8	83.5	90.9	92.9
DINOv2 GeM	4096	62.6	78.3	83.0	85.4	93.9	95.0	35.4	52.5	59.6	89.5	96.5	98.0	83.0	92.1	93.9
DINOv2 MixVPR	4096	72.1	85.0	88.3	90.0	95.1	96.0	63.6	80.1	84.6	94.6	98.3	**99.3**	89.8	94.9	96.1
DINOv2 NetVLAD	24576	**75.8**	86.5	89.8	**92.4**	95.9	96.9	71.8	86.5	90.1	**95.6**	**98.7**	**99.3**	90.8	95.7	**96.7**
DINOv2 NetVLAD (dim. red.)	8192	73.3	85.6	88.3	90.1	95.4	96.8	70.1	86.5	90.2	95.4	98.4	99.1	90.6	95.4	**96.7**
DINOv2 SALAD (ours)	8192 + 256	75.0	**88.8**	**91.3**	92.2	**96.4**	**97.0**	**76.0**	**89.2**	**92.0**	95.1	98.5	99.1	**92.1**	**96.2**	96.5

Experiments

- Qualitative Results

감사합니다