Enhancing Visual Place Recognition through Advanced Pooling Strategies and Optimization Techniques

Davide Benotto
Politecnico di Torino
s332150@studenti.polito.it

Paolo Riotino
Politecnico di Torino
s332530@studenti.polito.it

Umberto Piccardi
Politecnico di Torino
s331183@studenti.polito.it

Abstract

Visual Place Recognition (VPR) involves determining the geographic location of a query image by comparing it with a database of geotagged images. In this paper, we present advancements in visual geolocalization using deep learning techniques. Building upon the foundational work of the gsv-cities project, which leverages Google Street View imagery for urban geolocalization, we introduce a new dataset, the San Francisco XS Validation Set. Then, we conducted extensive implementations, hyperparameter tuning, and experiments with various aggregators and optimizers to enhance the model's performance. Our approach was rigorously evaluated using benchmark datasets such as Tokyo XS and the San Francisco Test Set, enabling a comprehensive comparative analysis. Additionally, we visually examined the model predictions to further validate their effectiveness. We also investigate the effects of employing MLP-mixer style aggregation. Our results demonstrate the potential for significant improvements in accuracy through these modifications.

https://github.com/paoloriotino/project_3

1. Introduction

Visual Place Recognition (VPR), also referred to as Visual Geo-localization (VG), is a fundamental task in computer vision and robotics. It involves identifying the geographical location depicted in a query image by comparing it against a database of geo-tagged images. This task is often approached as an image retrieval problem, where the goal is to identify the most visually similar images in the database and use their metadata to infer the location of the query image. Building on the work of Amar Ali-bey *et al.* [1], we conducted a first analysis on a resnet18 model truncated at the third convolutional layer, followed by an average pooling layer. We then performed extensive experiments to improve the performance of this baseline model, shown in Tab. 1.

Dataset	Recall@1	Recall@5
SF-XS-val	54.95	83.91
SF-XS-test	17.20	48.80
Tokyo-XS	29.84	61.90

Table 1. Recall comparison with a resnet18 model with avg pooling, between three different datasets

To further enhance our model beyond its baseline performance, we investigated the impact of various optimization strategies. This involved experimenting with different optimizers, learning rate schedules, and the incorporation of the MixVPR aggregation layer.

This comprehensive approach allows us to validate and test our model across diverse urban environments, ensuring its robustness and effectiveness in real-world VPR applications. Through these experiments, we aim to identify the optimal configurations and strategies for enhancing VPR systems.

2. Related Work

Visual geolocalization is often approached as an image retrieval task, where the goal is to match a query image to a database of known locations. Recent advancements in deep learning have significantly enhanced the accuracy and robustness of visual geolocalization systems.

A major advancement in this field is NetVLAD [3], a convolutional neural network (CNN) architecture specifically designed for place recognition under weakly supervised conditions. NetVLAD integrates a Vector of Locally Aggregated Descriptors (VLAD) layer into a CNN, enabling it to create compact and highly informative image representations. This architecture has consistently outperformed traditional methods in various benchmarks, demonstrating its effectiveness in capturing the essential features necessary for robust place recognition.

Another notable contribution is a method that focuses on

fine-tuning CNNs for image retrieval tasks without manual annotations [10]. By utilizing geometry verification and database mining, this approach automatically refines the network's parameters, enhancing its discriminative power and leading to improved retrieval performance. This is particularly beneficial for visual geolocalization, where precise and efficient image matching is crucial for accurate location identification.

Recent research has also emphasized the importance of large-scale, diverse datasets for training visual geolocalization models. The GSV-Cities dataset, for instance, focuses on advancing supervised place recognition by providing a comprehensive collection of urban imagery. Additionally, the introduction of MixVPR [2], a novel feature mixing approach, further refines the feature extraction process, contributing to improved recognition accuracy and robustness.

In conclusion, these advancements in deep learning, particularly in CNN architectures and training strategies, are reshaping the landscape of visual geolocalization. By leveraging innovative techniques and large-scale datasets, researchers are pushing the boundaries of what is achievable in terms of accuracy and robustness, paving the way for more reliable and practical visual geolocalization systems in real-world applications.

3. Methodology

3.1. Datasets

3.1.1 Training Dataset

For training our Visual Place Recognition (VPR) model, we utilize GSV XS Cities, a subset of the GSV Cities dataset proposed in the GSV-Cities paper [1]. This dataset provides extensive geographic coverage across over 23 cities worldwide over a 14-year period, introducing a wide variety of perceptual changes ideal for robust VPR models. It includes highly accurate ground truth data, such as precise GPS coordinates and viewing directions, which simplifies mini-batch creation and eliminates weak supervision bottlenecks. The diversity and accuracy of GSV-Cities significantly enhance VPR performance and ensure our model generalizes well to real-world scenarios, improving accuracy even on existing methods.

3.1.2 Evaluation Dataset

For validation, we use the SF-XS-Val dataset, a subset of the San Francisco eXtra Large (SF-XL) dataset from [4], created from Google StreetView imagery. SF-XL is the first city-wide, dense, and temporally variable dataset. The SF-XS-Val subset ensures that our validation process benefits from the extensive and varied nature of the dataset, enhancing the robustness and reliability of our model's performance assessment.

3.1.3 Test Datasets

For the test phase, we use two different datasets:

- SF-XS-Test: This dataset consists of 1000 images collected from Flickr, providing a wide range of viewpoint and illumination changes, including both day and night conditions.
- Tokyo-XS-Test: This dataset, a subset of the 24/7-Tokyo dataset [12], includes 315 query images of Tokyo, captured using smartphones. The images were taken from three different viewing directions and at three different times of day (day, sunset, night), introducing significant illumination and structural changes.

These diverse and challenging test datasets enable a comprehensive evaluation of our model's performance under varying conditions, ensuring its reliability and effectiveness in real-world applications.

3.2. Training Tuning

To further enhance the visual geolocalization performance, we explore the use of a Generalized Mean (GeM) pooling layer [10]. The GeM pooling layer has been shown to provide more discriminative power in deep learning models for various tasks, including image retrieval and geolocalization.

As previously mentioned, we employed a ResNet18 architecture [5] as our backbone, followed by average pooling as the aggregator, and obtained our baseline recall values on three observed datasets. Building on this setup, we replace the average pooling layer with different pooling layers. This adjustment is implemented while keeping other training parameters constant, including the same scheduler, learning rate, and weight decay. Throughout our process, we consistently use the multisimilarity loss [13], which encourages embeddings to minimize intra-class variations and maximize inter-class distances simultaneously, promoting effective clustering and discrimination in embedding spaces.

3.2.1 Pooling Layers

We experimented with the following advanced pooling layers to improve model performance:

- GeM (Generalized Mean) Pooling: The GeM pooling layer provides enhanced discriminative power by generalizing the mean pooling operation. Instead of taking the arithmetic mean of the feature maps, GeM pooling uses the p-norm, allowing for more flexibility and better feature aggregation.
- MixVPR: MixVPR is a sophisticated pooling strategy designed to combine multiple pooling operations and learn optimal feature aggregation. This layer aims to

Pooling Layer	SF-XS-val		SF-XS-test		Tokyo-XS	
	R@1	R@5	R@1	R@5	R@1	R@5
AVG	54.96	69.60	17.20	32.40	29.84	46.35
GeM	57.12	72.68	23.00	37.10	34.29	51.75
MixVPR	**78.41**	**86.28**	**57.40**	**69.10**	**72.38**	**84.13**
GeM Δ AVG	+2.16	+3.08	+5.8	+4.7	+4.45	+5.4
MixVPR Δ AVG	**+23.45**	**+16.68**	**+40.2**	**+36.7**	**+42.54**	**+37.78**

Table 2. Comparative Performance of AVG, GeM, and MixVPR Pooling Layers on Validation and Test Sets (30 Epochs)

capture richer representation by mixing various pooling techniques, thereby enhancing the model's ability to discriminate between similar images.

The results from integrating these pooling layers are presented in Tab. 2. Overall, we observed improved recall performance across all three datasets with the GeM and MixVPR pooling layers compared to the average pooling layer. This indicates the potential of these advanced pooling strategies to enhance feature aggregation and improve visual geolocalization results.

3.2.2 Optimizers

To further fine-tune our model, we experiment with various optimizers, including Adam, AdamW, and ASGD. Each optimizer has unique characteristics and potential benefits for model convergence and generalization:

- Adam (Adaptive Moment Estimation) [6]: Adam computes adaptive learning rates for each parameter and maintains two moving averages: the mean and the uncentered variance of the gradients.
- AdamW (Adam with Weight Decay) [8]: An improvement over Adam, AdamW decouples weight decay from the gradient update, providing better control over regularization and often leading to improved performance in deep learning models.
- ASGD (Averaged Stochastic Gradient Descent) [9]: An extension of SGD, ASGD averages the weights over time, which can help stabilize training and often results in a more robust model.

3.2.3 Schedulers

In addition to optimizer tuning, we experiment with various learning rate schedulers to optimize the training process. The schedulers adopted in this tuning phase are:

- MultiStepLR: The original scheduler used in all previous training phases. This scheduler decreases the learning rate by a factor of gamma at specified milestones.
  $milestone = [5, 10, 15],\ gamma = 0.3$
- ReduceLROnPlateau: Reduces the learning rate when a metric has stopped improving, which can help in fine-tuning and achieving better convergence.
  $mode = min,\ min\_lr = 0,\ monitor = loss$
- CosineAnnealingLR [7]: Decreases the learning rate following a cosine function, allowing for smoother reductions and potentially better performance towards the end of training.
  $T\_max = 5$
- CyclicLR (mode set to "triangular2") [11]: This scheduler cycles the learning rate between two boundaries with a constant frequency, which can help in escaping local minima.
  $base\_lr = 1e^{-3},\ max\_lr = 0.1,\ step\_size\_up = 1$

The learning rate trends of each scheduler are displayed in Fig. 1. By systematically experimenting with these optimizers and schedulers, we aimed to identify the most effective strategies for our visual geolocalization task. The following sections will detail the performance results and comparative analysis of these configurations.

4. Results

4.1. Comparative Analysis

Throughout our experiments the three datasets have shown very diverse results. Tokyo-XS and SF-XS-test proved particularly challenging due to their diverse composition, as previously illustrated in Sec. 3.1. These disparities between the datasets have been evident since the initial analyses, and the objective of our work has been to address and overcome these challenges.

4.1.1 Pooling layers comparison

As we have shown in the Tab. 2 our main improvement related to the pooling layers where reached using the MixVPR aggregator. This enhancement is particularly evident when testing on SF-XS-test and Tokyo-XS, where we observed a substantial increase of over 40 percentage points compared to the performance of the GeM pooling layer. This trend seems related to the presence in those two datasets of images with a variety of occlusions, viewpoint and illumination change. We highlight the different behaviours in the Fig. 4 by presenting some significant examples. We can notice that:

- Average pooling is susceptible to occlusion caused by natural and artificial objects, such as trees and cars. .

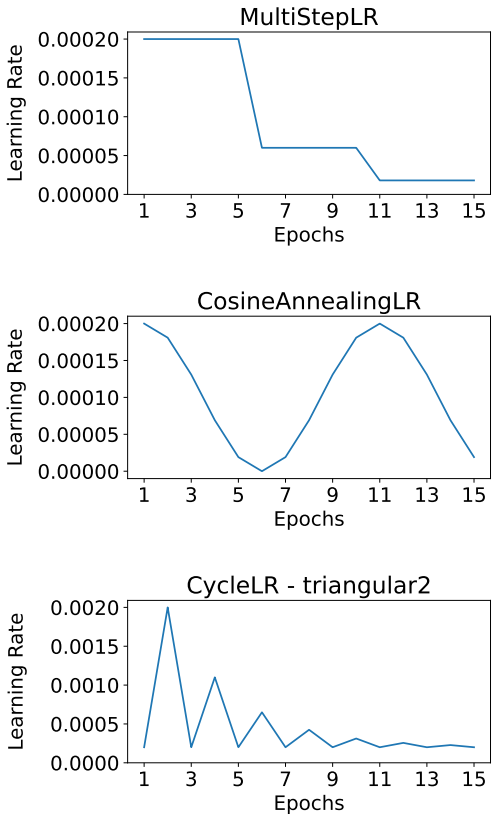

Figure 1. Comparison of the Learning Rates values variation across the epochs for MultiStepLR, CosineAnnealingLR and CycleLR(triangular2)

This problem arises because they treat all input pixels equally which sometimes lead to loss of important spatial information.

- GeM pooling represents an initial upgrade. Introducing a new parameter p provides several benefits. Firstly, it can dynamically adjusts its pooling strategy to mitigate the effects of occlusions. It prioritizes the most relevant features, similar to max pooling, while still maintaining the ability to generalize like average pooling.

- MixVPR demonstrates impressive performance on the most challenging queries which display the aforementioned characteristics. Indeed, one of the main advantages of the MixVPR aggregator is its robustness to occlusions or variations in lightning.

4.1.2 Optimizers comparison

To further enhance performance We implement different optimizers and we investigate about their hyperparameters.

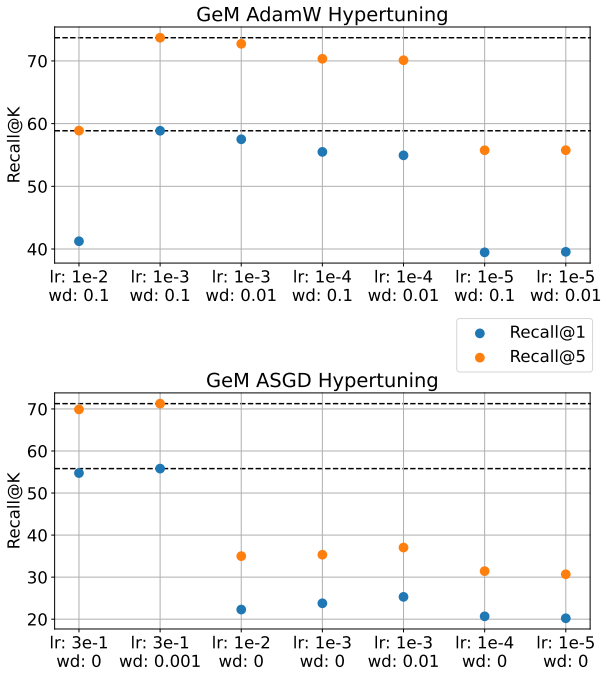

Figure 2. Hyperparameter Tuning for GeM Pooling with AdamW and ASGD Optimizers. The best recall values were achieved with the AdamW optimizer using a learning rate of 1×10^{-3} and a weight decay of 0.1, and with the ASGD optimizer using a learning rate of 3×10^{-1} and a weight decay of 0.001.

Pooling				SF-XS-val		SF-XS-test		Tokyo-XS	
Layer	Optimizer	LR	WD	R@1	R@5	R@1	R@5	R@1	R@5
GeM	Adam	2e-4	0	56.82	72.33	21.60	36.90	**35.24**	**53.02**
GeM	AdamW	1e-3	0.1	**58.85**	**73.70**	**23.10**	35.70	33.33	52.38
GeM	ASGD	3e-1	0.001	55.84	71.24	22.20	**37.30**	30.48	**53.02**
MixVPR	Adam	2e-4	0	**78.28**	85.86	**57.70**	**69.40**	**73.65**	**84.13**
MixVPR	AdamW	1e-3	0.1	78.23	**86.15**	53.60	66.90	64.44	78.73
MixVPR	ASGD	3e-1	0.001	77.13	85.30	52.80	67.00	61.27	74.29

Table 3. Comparative Performance (15 Epochs) of GeM and MixVPR Pooling Layers with Different Optimizers: Adam, AdamW, ASGD

Starting from our standard architecture with a GeM pooling layer, we focus on these optimizers: Adam, AdamW and ASGD. In particular, we emphasize on finding the best learning rate and weight decay for AdamW and ASGD, as illustrated in Fig. 2. On the base of these results, we tested our best configurations on both test datasets, adopting GeM and MixVPR. Outcomes are shown in Tab. 3

Model		SF-XS-val		SF-XS-test		Tokyo-XS	
Aggregator	Scheduler	R@1	R@5	R@1	R@5	R@1	R@5
GeM	MultiStepLR	56.82	72.33	21.60	36.90	**35.24**	**53.02**
GeM	ReduceLROnPlateau	58.25	73.29	**23.70**	37.90	32.38	50.16
GeM	CosineAnnealingLR	57.58	72.74	21.40	**38.80**	34.29	**53.02**
GeM	CyclicLR_triangular2	**58.26**	**73.36**	20.80	37.10	29.52	51.11
MixVPR	MultiStepLR	78.28	**85.86**	**57.70**	69.40	**73.65**	**84.13**
MixVPR	ReduceLROnPlateau	**78.66**	85.83	55.50	68.40	67.62	80.63
MixVPR	CosineAnnealingLR	78.27	85.37	56.10	**69.70**	71.75	83.17
MixVPR	CyclicLR_triangular2	77.46	85.31	50.90	62.70	70.79	80.63

Table 4. Impact of Learning Rate Schedulers on GeM and MixVPR Aggregators (15 Epochs)

4.1.3 Schedulers comparison

To complete our analysis which aims to find the best hyperparameters configuration, we conduct a deeper exploration on schedulers. Starting from the MultistepLR adopted in our standard configuration, we experiment the following schedulers: ReduceLROnPlateau, CosineAnnealingLR, CyclicLR (triangolar2 mode). A more detailed study is shown in Tab. 4. As we can notice with GeM aggregator some of the adopted schedulers perform better. For the SF-XS-val the CycicleLRtrinagular 2 has the higher recall@1, overtaking MultiStepLR by 1.5. Performance of this scheduler sharply decreases on SF-XS-test and Tokyo-XS. In contrast, ReduceLROnPleatue reaches the highest recall@1 in SF-XS-test, whereas on Tokyo-XS the MultiStepLR scheduler is still the most precise. Regarding MixVPR aggregator, we observe consistently strong results across different schedulers. In Fig. 3 is possible to observe the trend of the recall@1 throughout the training epochs.

5. Conclusion

In this work, we present a comprehensive investigation into enhancing the accuracy and robustness of Visual Place Recognition (VPR) systems. Through extensive experimentation and analysis, we have demonstrated that advanced pooling strategies, such as the Generalized Mean (GeM) pooling and the MixVPR aggregator, significantly enhance VPR performance. Notably, MixVPR excels in handling challenging scenarios with occlusions, varying viewpoints, and illumination changes. Our findings also highlight the importance of carefully selecting and tuning hyperparameters, such as the optimizer and learning rate scheduler, to achieve optimal results. The combination of them resulted in consistent performance gains across diverse datasets.

Future work will focus on further refining these techniques and exploring additional datasets to continue advancing the field of visual place recognition.

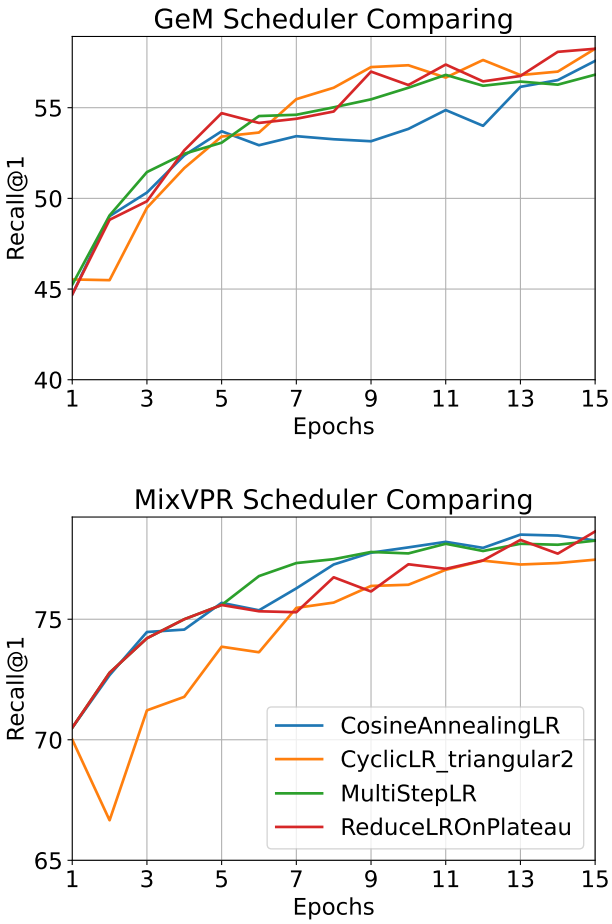

Figure 3. Comparison of scheduler tuning over 15 epochs. The GeM aggregator achieved the best recall values with the CycleLR scheduler (mode set to triangular 2), while the MixVPR aggregator yielded similar results using both the CosineAnnealingLR and ReduceLROnPlateau schedulers.

References

[1] Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguère. Gsv-cities: Toward appropriate supervised visual place recognition. Neurocomputing, 513:194–203, Nov. 2022. 1, 2

[2] Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguère. Mixvpr: Feature mixing for visual place recognition, 2023. 2

[3] Relja Arandjelović, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition, 2016. 1

[4] Gabriele Berton, Riccardo Mereu, Gabriele Trivigno, Carlo Masone, Gabriela Csurka, Torsten Sattler, and Barbara Caputo. Deep visual geo-localization benchmark, 2022. 2

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 2

Figure 4. Qualitative comparisons of retrieved images for each dataset using AVG, GeM, and MixVPR methods.

[6] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 3

[7] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts, 2017. 3

[8] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. 3

[9] B.T. Polyak. Some methods of speeding up the convergence of iteration methods. USSR Computational Mathematics and Mathematical Physics, 4(5):1–17, 1964. 3

[10] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Fine-tuning cnn image retrieval with no human annotation, 2018. 2

[11] Leslie N. Smith. Cyclical learning rates for training neural networks, 2017. 3

[12] Akihiko Torii, Relja Arandjelović, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. 24/7 place recognition by view synthesis. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1808–1817, 2015. 2

[13] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. Multi-similarity loss with general pair weighting for deep metric learning. In IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 5022–5030, 2019. 2