On the Study of Data Augmentation for Visual Place Recognition

Suji Jang[1] and Ue-Hwan Kim*

***Abstract*— In the field of robotics engineering and autonomous driving vehicles, precise estimation of positions through visual place recognition (VPR) is crucial not only for reducing localization errors caused by visual odometry but also for preventing the creation of ambiguous maps in unfamiliar environments. Despite numerous research efforts aimed at improving VPR performance by addressing challenges such as illumination variation, occlusions, and dynamic objects, contemporary approaches have primarily focused on model-based methods, with limited attention given to data augmentation (DA) methods. Therefore, there is a need to investigate the impact of DA on the generalization ability of VPR. To achieve this objective, this study compares VPR learning approaches, conducts a comprehensive empirical analysis, and presents crucial insights. The results of this study can provide useful guidance for the design of future VPR systems and contribute to the advancement of computer vision and robotics research.**

I. INTRODUCTION

The primary function of Visual Place Recognition (VPR) is to determine whether a given location has been previously visited by a robot or vision-based navigation system, thereby enabling the system to navigate and orient itself in a given environment [1], [2]. Strong performance in VPR is essential for developing service robots that can function effectively in real-life scenarios. This technique is particularly useful for performing loop closure detection in simultaneous localization and mapping (SLAM) algorithms, which can help to reduce positioning errors induced by visual odometry and prevent the building of ambiguous maps in unknown environments [3]. In multi-agent SLAM, robust VPR is crucial for estimating relative pose transformations and generating accurate global maps, as the viewpoints and scales of the places can vary depending on the paths and directions of each agent [4]. Recent literature in VPR has focused on deep learning-based methods, which have shown superiority over traditional hand-crafted methods [5].

Nonetheless, achieving the robust performance of VPR is challenging due to several factors that impede the learning process and constrain neural networks from processing information optimally. Concretely, modern VPR studies suffer from two major disadvantages as follows:

- Contemporary research in VPR generally focuses on improving the model architecture without considering data augmentation (DA) techniques—resulting in an unbalanced view between the model side and the dataset side approaches.
- Recent studies have revealed that most DA techniques are highly sensitive to the choice of augmentation parameters and datasets, and therefore fail to consistently improve performance across various datasets [6]. However, the conditions under which this lack of improvement occurs have not been sufficiently analyzed.

To overcome the limitations mentioned above, we propose to have a closer look at potential synergies between VPR models and various DA. For this, we design a comprehensive empirical study to investigate the effect of DA for VPR under various conditions. Specifically, we control the following factors to design our study: training datasets, pooling methods, loss functions, and DA configurations. Through the systematic combination of these factors, we closely examine the ramification of each factor.

As a result of our study, we uncover a number of vital insights. We summarize the most important as follows: (1) The effect of DA is not universally applicable to all VPR training approaches: even the same DA operation would not become effective depending on the training approach taken. (2) DA is also effective in datasets that include diverse appearance change scenarios and cover long-term time spans for the generalization of VPR. (3) The cross-entropy loss with DA is an effective and simple approach for optimizing VPR performance. (4) A generic DA configuration that widely enhances performance exists in the case of classification approaches; as a result, we obtained new state-of-the-art performance.

In summary, the main contributions of our work are as follows:

- We devise and perform an extensive empirical study to delve into potential synergies between VPR and DA.
- We present a generic DA configuration for VPR.
- We obtain new state-of-the-art performance.
- We make the source code of our study public[1].

II. RELATED WORK

A. Training approaches for image representations

The success of VPR in diverse environments relies on the availability of strong image representations that can perform effective image recognition. To achieve this, recent research

This work was supported in part by the Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2022-0-00907, Development of AI Bots Collaboration Platform and Self-organizing AI, No.2019-0-01842, Artificial Intelligence Graduate School Program (GIST)) and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2022R1C1C1009989).

The authors are with the AI Graduate School, Gwang-ju Institute of Science and Technology, Gwang-ju 61005, Republic of Korea (e-mail: sujijang@gm.gist.ac.kr; uehwan@gist.ac.kr).

[1]https://github.com/AutoCompSysLab/Data_Augmentation

has explored the use of deep neural networks. This technique can be categorized into two types of training approaches: contrastive learning and classification.

Contrastive learning [7]–[9] considers a pairwise loss (mostly relying on a triplet loss) and trains images with the same geographical tag in a database to be closer and images with different geographical tags to be farther based on a given query. However, this approach displays inferior performance and requires a larger amount of computational resources than classification-based approaches [10]. Further, the performance of contrastive learning heavily depends on sample mining from the training database [11].

Classification [10], [12] considers the cross-entropy loss and divides the Earth's surface into cell grids and trains a classifier to assign input images to these cells. However, this approach has the disadvantage of dividing the location into geographical areas, making it challenging to adjust the number of classes. Furthermore, the cross-entropy loss is not well correlated with geo-location information [12]. However, a recent study has enhanced the correlation between the cross-entropy loss, which does not reflect relationships between classes, and geographical location by grouping uninterrupted classes. Additionally, classes are categorized not only based on the GPS coordinates of images but also on the orientation information—addressing the limitations of the classification approach [11].

B. Inference as Image Retrieval and Challenges

VPR is the task of recognizing the depicted location in an image, which is commonly achieved through an image retrieval approach [2]. A database of images tagged with location identifiers such as landmark names or GPS coordinates, represents prior knowledge of the places of interest. When a new query image needs to be localized, the VPR system searches the database for similar images and infers the location of the query image based on the tagged location of the matched image.

Challenges. For VPR as an image retrieval problem, there are several unique challenges that distinguish it from other retrieval approaches. These challenges stem primarily from the complexity of the scene and the ever-changing nature of the environment [13]. First, the same scene can appear significantly different because places may not always be revisited from the same viewpoint and position as before [14], [15]. Second, VPR differs from other instance search tasks, such as catalog search, in that it typically involves multiple visual elements without a single central object of interest. These elements can include cluttered and irrelevant features, such as people or vehicles, which can obscure or distract from more useful objects in the environment. Third, the variability in environmental conditions, such as lighting (day/night/shadow), weather, or season [16], can cause significant changes in the same scene. Fourth, there can be little overlap between the query and database images [17]. Fifth, two locations may contain common elements that hinder their differentiation. This challenge is exemplified by the prevalence of recurring patterns in man-made structures found in urban environments, such as building facades or fences. [18].

TABLE I

VPR CHALLENGES AND CORRESPONDING DA METHODS

Challenges	Target DA	Methods
View-point shift	Geometric transformation	Shear, Translate, Rotate, Crop
Dynamic object (clutter and occlusions)	Erasing	Cutout, Hide and Seek, Random Erasing, GridMask
Illumination change	Color processing	Brightness, Color, Contrast
Environmental conditions (weather, season)	Texture modification	Sharpness, Posterize, Solarize

C. Generalization techniques to avoid overfitting

Developing service robots in real life requires strong VPR, but there are various real-world factors that make achieving strong performance difficult, such as image variations (view variation, illumination change, camera jitter, and occlusion) and changes in the real world (different weather, dynamic objects, and domain shift). These factors significantly alter the scenes of the same location, making place recognition challenging and leading to overfitting. And various techniques have emerged to prevent overfitting and generalize the model. As discussed extensively in [19], generalizations can be achieved with respect to models and the data itself.

Model Architecture. Designing effective model architectures helps avoid overfitting and increase generalization performance. It has evolved into increasingly more complex model architectures, such as AlexNet [20], VGG16 [21], GoogleNet (Inception) [22], ResNet [23], DenseNet [24], and Vision Transformer [25]. However, despite the positive correlation between the model size and generalization ability, constraints such as computational complexity and cost limit the feasibility of using larger models for commercially available applications.

Data Augmentation (DA) diversifies the training data to various imaging conditions that are not available. As a model cannot (easily) deal with unknown operating conditions [26], DA serves the purpose of teaching those conditions even when the training data does not include them. Consequently, DA can enhance a model's generalization performance by artificially introducing various types of variations to the training dataset [19], [27], enabling the model to learn to recognize places under diverse conditions. DA offers a solution at the root of the problem—the dataset—and it is an effective approach that allows the direct utilization of techniques corresponding to the target situation.

D. Data Augmentation methods

A DA method simulates a specific situation and helps models handle such a situation in the real world. We summarize VPR challenges and corresponding DA methods in Table I and provide examples in Fig.1. First, the application of geometric transformations, such as Shear, Translate, and Rotate, serve to enhance view-point variation by enabling the manipulation of spatial relationships and geometrical attributes within a given context [19]. Second, image erasing techniques such as Cutout [28], Hide and Seek [29], Random

Fig. 1. Examples of applying each DA Method in the Pittsburgh dataset [18].

Erasing [30], GridMask [31] help to handle clutter and occlusion due to non-informational elements. The basic idea is that, by blocking out the most important parts of the image at the patch-level subregion of the pixel values and replacing them with a constant or random value, the model can learn powerful features. Third, color processing techniques such as Brightness, Color, and Contrast improve the robustness of illumination changes. Fourth, texture modifications such as adjusting Sharpness, Posterize, and Solarize are effective in enhancing the robustness of image recognition systems under environmental changes, such as weather and season because the texture is more important than shape for deep learning-based method, especially the convolutional neural networks architectures [32]. Moreover, RandAugment [33] is a method that combines geometric, color, and texture transformations, such as image resizing, rotation, and color adjustments. It involves setting the number of operations for image augmentation and the magnitude of each randomly chosen operation.

Table II is a summary of DA techniques used in the latest models for classification, object detection, and segmentation tasks, based on their performance rankings. It can be observed that in tasks other than VPR, model architecture improvements, and DA techniques have been used in conjunction to achieve performance improvements. Effective DA techniques have also been standardized. On the other hand, Table III presents a summary of the DA techniques used in the VPR task. Except for CosPlace [11], the utilization of DA is nearly absent. Even in CosPlace, only two basic operations were applied without incorporating advanced DA techniques. In summary, DA has predominantly been utilized in the context of classification, object detection, and segmentation tasks, while minimal attention is paid to its applicability in the VPR.

III. Methodology

In this section, we describe our study setup. First, we delineate the backbone architecture. Next, we introduce datasets, polling methods, and loss functions.

TABLE II

Image augmentation method applied to the latest models of Classification, Object Detection, and Semantic segmentation benchmark datasets.

	Ranking	Model	Image augmentation methods
Image Classification (ImageNet ReaL)	1	Model soups [34]	Crop, Flip, AutoAugment, Mixup CutMix, Erasing, ColorJitter
	2	ViTAE [35]	Crop, Flip, ColorJitter, Grayscale GaussianBlur, Erasing, RandAugment, Mixup, CutMix
	4	Meta Pseudo Labels [36]	Crop, Large Scale Jitter, RandAugment CutMix, Mixup, Erasing, Grayscale
	8	FixEfficientNet [37]	Crop, Flip, ColorJitter
Object Detection (COCO test-dev))	1	InternImage [38]	Crop, Flip, AutoAugment, Mixup CutMix, Erasing, ColorJitter
	2	M3I [39]	Crop, Flip, ColorJitter, Grayscale GaussianBlur, Erasing, RandAugment, Mixup, CutMix
	3	EVA [40]	Crop, Large Scale Jitter, RandAugment CutMix, Mixup, Erasing, Grayscale
Semantic Segmentation (ADE20K)	1	BEiT-3 [41]	Crop, Flip, ColorJitter
	2	FD-SwinV2 [42]	Crop, Erasing, RandAugment, Mixup, CutMix
	3	MaskDINO-SwinL [43]	Crop, Large-Scale Jittering

TABLE III

Image augmentations method applied to SotA model of VPR benchmark datasets

Datasets	State of the Art	Image augmentation method
Mapillary val	CosPlace [11]	ColorJitter, RandomResizeCrop
Nordland	Patch-NetVLAD [44]	Resize
Mapillary test	GCL_PCA [45]	Resize
Pittsburgh-250k-test	Conv-AP [9]	-
Oxford5k, Paris6k	Offline Diffusion [46]	-

A. Backbone

The backbone, which plays a crucial role in extracting highly informative feature maps from images, is a fundamental component of all VPR systems. Typically, a CNN pre-trained on image classification datasets (ImageNet-1K) is adapted for a VPR backbone by connecting a trainable aggregation layer that effectively aggregates the feature maps into discriminative representations. We use the VGG16 [21] backbone for our study. Following the common approach, we removed the last ReLU and maxpool layers, and the conv1$\sim$4 layers are frozen, while only the conv5 layer remains trainable [8], [9], [11].

B. Datasets

We utilize a total of three datasets to evaluate the performance of VPR under various conditions. Specifically, we use Pitts30k and SF-XL (small) for training and validation, and Pitts30k test, Tokyo24/7, and SF-XL test v1 for testing. Table IV compares and summarizes the characteristics of each dataset.

Pitts30k [8] comprises panoramic images of Pittsburgh extracted from Google Street View, which are cropped into

TABLE IV

Comparison of various VPR datasets

Datasets	# images (database+query)	Appearance Change Season	Viewpoint	Day/Night	Time span
SF-XL (small) [11]	train: 60k	✔	✔	✔	15 years
Pitts30k [8]	train: 17k test: 16k	✖	✔	✖	-
Tokyo24/7 [14]	test: 76k	✖	✔	✔	-
SF-XL test v1 [11]	test: 28k	✔	✔	✔	16 years

12 components at 30-degree intervals. The dataset provides two pitch images for one space. Although it promotes diversity in viewpoint through two-pitch images, it does not contain weather and seasonal information. The dataset is divided into database and query sets separated in time by about two years, and train-set, validation-set, and test-set are separated geographically.

SF-XL [11] consists of panoramic images of San Francisco extracted from Google Street View and contains a wide range of temporal changes from 2007 to 2021. It is a lot denser dataset compared to Pitts30k or Tokyo24/7 and contains images of various environments such as weather changes, year changes, and time changes. **SF-XL test v1** consists of a database and queries. The database is extracted from a subset of SF-XL, which contains 27k images from the year 2013. The queries are designed to evaluate the performance of the model on different domains. The query set includes 1k images that encompass various characteristics such as night images, grayscale images, heavy changes in viewpoint, and occlusions.

Tokyo24/7 [14] is used only for testing purposes. It comprises a Google Street View image database and query images captured at various time zones (day, sunset, and night) using phone cameras. This dataset poses a challenging scenario for VPR due to the differences in image quality, viewpoint, and illumination conditions between the database and query images.

C. Pooling Methods

In this research, we focus on two pooling methods: NetVLAD, which is utilized in the NetVLAD model, and GeM, which is utilized in CosPlace.

NetVLAD [8] uses a learnable assignment function that replaces the undifferentiable assignment function in the original VLAD [47] method—enabling end-to-end learning via backpropagation. The NetVLAD pooling ($\mathbf{a}_k$) is as follows:

$$\mathbf{a}_k = \frac{1}{N}\sum_{i=1}^{N} w_{i,k}(\mathbf{x}_i - \mathbf{c}_k), \tag{1}$$

where $\mathbf{x}_i$ is the i-th local descriptor extracted from the image. $\mathbf{c}_k$ is the k-th cluster centroid and $w_{i,k}$ is the probability weight of $\mathbf{x}_i$ belonging to the k-th cluster, which is defined as:

$$w_{i,k} = \frac{exp(\mathbf{w}_k^T \mathbf{x}_i + b_k)}{\sum_{j=1}^{K} exp(\mathbf{w}_j^T \mathbf{x}_i + b_k)}, \tag{2}$$

where K is the number of clusters, and $\mathbf{w}_k$ and b_k are the weight and bias of the k-th cluster, respectively. Using the above equation, all local descriptors are transformed into difference vectors with cluster centroids, and the NetVLAD vector is obtained by averaging them.

GeM [7] pooling is a generalization of the Global average-pooling method and serves as a common expression for various pooling methods. The GeM pooling formula is as follows:

$$\mathbf{y}_c = \left(\frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i^c)^{\alpha}\right)^{\frac{1}{\alpha}}, \tag{3}$$

where $\mathbf{x}_i^c$ is the i-th pixel value in the c-th channel, N is the number of pixels, and α is the pooling parameter. The above formula works by taking the average of each channel's pixel values raised to the power of α and then taking the α-th root. We generally set $\alpha = 3$, and the CosPlace architecture connects GeM pooling and fully connected layers to handle large-scale datasets [6].

D. Loss Functions

The main difference between the contrastive learning and classification approaches of VPR lies in their loss functions. Contrastive learning-based methods employ a pairwise loss such as the weakly supervised triplet ranking loss [8], which requires database and query on a training dataset and involves sample mining. On the other hand, classification-based methods utilize the cross-entropy loss such as the large margin cosine loss [48] and require datasets of multiple classes.

Weakly Supervised Triplet Ranking Loss (WSTRL) [8] utilizes training data in the form of triplets (q, p_i^q, n_j^q), where q denotes a query image, p^q represents the potential positive image (i.e., an image taken within 10m of the query image), and n^q is the definite negative image (i.e., an image taken from 25m away). WSTRL is defined by

$$L_{WSTRL} = \sum_j l(\min_i d_\theta^2(q, p_i^q) + m - d_\theta^2(q, n_j^q)), \tag{4}$$

where l is the hinge loss $l(x) = max(x, 0)$, m is a constant parameter giving the margin, and d_θ is the distance measure between the feature representations of the data points.

Large Margin Cosine Loss (LMCL) [48] aims to improve the discriminative power of feature embeddings; features from the same class are grouped closely together in the embedding space, while features from different classes are well separated from each other. By fixing the size of feature x to $||x|| = s$ and adding a margin m to the cosine value itself, LMCL minimizes intra-class variation and maximizes inter-class variation utilizing the Softmax loss as follows:

$$L_{LMCL} = \frac{1}{N}\sum_i -\log\frac{e^{s(cos(\theta_{y_i,i})-m)}}{e^{s(cos(\theta_{y_i,i})-m)} + \sum_{j\neq y_i} e^{cos(\theta_{j,i})}}, \tag{5}$$

where N is the number of training samples, and θ_i is the angle between the learned feature vector and the weight vector of class i.

E. Augmentation Techniques

We design our experiments examining the following DA as a solution to the aforementioned VPR challenges. We apply Cutout (CO) [28] to overcome the problem that non-informational elements such as people or vehicles can distract or even occlude unique elements of the environment, RandomResizeCrop (RRC) to overcome the problem that the same scene may look quite different from different perspectives, ColorJitter (CJ) to overcome the problem of changing environmental conditions, such as lighting (day/night/shadow), weather, or season, where the same scene can differ significantly. We adopted RandomResizeCrop and ColorJitter from CosPlace, while complex erasing

techniques [30], [49], [50] led to performance degradation, thus we applied a simple form of Cutout with a smaller range. In addition, we take RandAugment (RA) [33], which combines geometric, color, and texture transformation, into account.

F. Training Configuration

We investigate the impact of DA on two base architectures that share the VGG16 backbone [21]: NetVLAD [8] and CosPlace [11]. On one hand, the NetVLAD model is trained on the Pitts30k dataset [8], and feature maps are collected using NetVLAD pooling, while the model is updated using the WSTRL loss [8]. The CosPlace model, on the other hand, is trained on the subset of SF-XL dataset [11], and feature maps are collected using GeM pooling [7] and fully connected layers, while the model is updated using the LMCL loss [11]. The details of each training configuration can be found in the first and fourth rows of Table V.

Pooling Replacement Study. This study investigates the impact of different pooling techniques used in two models on the relationship between DA and VPR. We conduct an experiment applying DA to the switched structures of the basic models, in which the pooling technique used in the basic NetVLAD structure is replaced with GeM, and the pooling technique used in the CosPlace structure is replaced with NetVLAD. The details of each training configuration can be found in the second and fifth rows of Table V.

Datasets Replacement Study. The datasets used for NetVLAD and CosPlace, i.e., Pitts30k and SF-XL, respectively, exhibit different characteristics as shown in Table IV. Therefore, we conduct experiments to replace the datasets in order to investigate the influence of the datasets on the relationship between DA and VPR. The details of each training configuration can also be found in the third and sixth rows of Table V.

IV. Experiments

A. Settings

In our experiment, we employ the identical hyperparameter settings as those presented in NetVLAD [8] and CosPlace [11]. We utilize a single NVIDIA Tesla A30 GPU for all experiments. To ensure convergence for each setting, we terminate training when the validation recall does not improve in 10 consecutive iterations.

Metric. We quantitatively measure the VPR performance with the standard recall@N metric following the standard evaluation protocol [1]. The metric considers a query image to be correctly localized if at least one of the top N retrieved database images is within $d = 25$ meters from the query's ground truth position.

Datasets. To train SF-XL (small) images using the NetVLAD model, approximately 60k images need to be divided into database and query. To match the number of queries in Pitts30k, images captured before April 2019 are used as a database, and those captured after that are used as queries. As a result, out of 59,650 images, 51,455 are used as a database, and 8,154 are used as queries. To train Pitts30k images using the CosPlace model, approximately 17k images need to be classified into classes. Since Pitts30k images do not have directional information and the number of images is relatively small, the directional information is ignored and the images are classified based on their classes without grouping them. As a result, a total of 455 classes are generated, each containing 16-168 images.

DA Training Schemes. We use the Pytorch library to implement the augmentation for our experiments except for Cutout. RandomResizeCrop set the image size to 512×512 to crop a minimum of 0 and maximum 0.5 and set ColorJitter's brightness $= 0.7$, contrast $= 0.7$, hue $= 0.5$, and saturation $= 0.7$. RandAugment's magnitude is 9 and the number of operations is 2. The number of holes of Cutout was set to 1 and the length was set to 16. The augmentation is applied in the following order: RandAugment or ColorJitter is applied first, followed by RandomResizedCrop, normalization, and finally Cutout.

B. Results and Analysis

Not All Approaches Are Universal. Table V presents the experimental results for the basic structures of both models and for when pooling and datasets were changed. The first and fourth rows represent the basic NetVLAD and CosPlace structures, respectively. The results show that DA has minimal impact on NetVLAD but is effective in CosPlace. The second and fifth rows show the results when the pooling was changed, and the third and sixth rows display the results when the dataset was changed. The experimental results reveal that when the NetVLAD model was applied with GeM pooling, the model without DA outperformed those with DA. Additionally, the NetVLAD model trained on the SF-XL dataset exhibits the best performance in SF-XL test v1 and Pitts30k when using the model without DA. The Tokyo247 dataset shows only slight improvement when RandAugment was applied. In contrast, the CosPlace model shows significant improvement in performance with DA applied in all experimental results, and RandomResizeCrop and ColorJitter exhibit generally good performance. Therefore, we infer that performance improvement due to DA was observed only when using the LMCL loss, regardless of the type of pooling and dataset.

Impact on the Realistic Dataset. Various VPR datasets that provide diversity in terms of temporal and spatial dimensions have been proposed, as shown in Table IV, to utilize datasets in generalized states for learning. This is in line with the goal of DA to improve the generalization performance of VPR models. Applying DA to the SF-XL dataset, which includes various realistic scenarios for generalization, observed significant performance improvements, as shown in the fourth and fifth rows in Table V. This proves that DA is effective even on datasets that contain real-world situations.

Cross-Entropy with DA is an Effective Way for VPR. According to a recent study, DA in the VPR model shows little performance improvement and inconsistent results depending on the dataset [6]; the cause is explained through the characteristics of the dataset. However, our experiments

TABLE V

COPARISON VPR METHODS: CONTRASTIVE LEARNING AND CLASSIFICATION. †: THE CONFIGURATION OF NETVLAD [8]. ‡: THE CONFIGURATION OF COSPLACE [11]

Training configuration				Performance		
Loss	Pooling	Training Dataset	Augmentation	SF-XL test v1	Pitts30k	Tokyo247
WSTRL	NetVLAD	Pitts30k	Base	41.0†	**81.51**†	**62.86**†
			ColorJitter, RandomResizeCrop	37.1	80.34	60.63
			RandAugment	40.6	81.32	62.86
			RandAugment, Cutout	**41.1**	81.28	62.86
	GeM	Pitts30k	Base	**28.3**	**73.09**	**44.44**
			ColorJitter, RandomResizeCrop	23.0	70.14	43.17
			RandAugment	27.8	70.98	42.86
			RandAugment, Cutout	27.4	70.94	42.54
	NetVLAD	SF-XL(small)	Base	**47.9**	**76.39**	53.02
			ColorJitter, RandomResizeCrop	45.4	75.89	53.33
			RandAugment	47.3	76.38	**55.56**
			RandAugment, Cutout	47.4	76.17	54.29
LMCL	GeM	SF-XL(small)	Base	56.8	81.75	58.73
			ColorJitter, RandomResizeCrop	61.7‡	84.98‡	**72.06**‡
			RandAugment	**63.5**	86.28	69.84
			RandAugment, Cutout	63.4	**86.74**	70.79
	NetVLAD	SF-XL(small)	Base	19.2	43.10	20.63
			ColorJitter, RandomResizeCrop	55.8	**80.31**	**63.17**
			RandAugment	**57.1**	78.37	55.87
			RandAugment, Cutout	56.0	78.12	51.43
	GeM	Pitts30k	Base	16.4	58.91	41.59
			ColorJitter, RandomResizeCrop	**27.6**	**67.25**	**51.11**
			RandAugment	23.6	64.13	46.98
			RandAugment, Cutout	23.9	63.97	47.30

demonstrate that such results do not apply to classification methods. Rather, our experiments suggest that the effect of DA is determined by the type of loss, not by a specific model architecture or training dataset. The reason for these results can be explained by the characteristics of each loss. The triplet loss, which forms the foundation of WSTRL, requires a data mining procedure that mines triplets for training. There are three types of triplets based on the relationship between query image, positive sample, and negative samples. Easy triplets refer to the case where the positive is close enough and the negative is far enough, and the data is already well separated. In this case, the loss becomes zero and the problem is that learning is not possible. Hard triplets refer to the case where the positive is too far and the negative is too close, resulting in easily misleading data samples. Semi-Hard is when the negative is far from positive but not far from margin value. To train a model discriminatively, it is important to avoid easy triplets and mine hard triplets [51], [52]. However, in distance-based datasets of VPR, it becomes easier to extract easy triplets as the size and amount of data increase because positive and negative samples are less likely to share similar semantic information [11]. Furthermore, the basic triplet loss samples the most similar image as a positive sample. As a result, there is a high probability that the positive sample in each training iteration will be a sample from the same scene as the query, increasing the likelihood of extracting easy triplets [53]. Applying weak DA that is not effective in creating hard triplets may still leave them as easy triplets, rendering them unsuitable for training. On the other hand, applying strong DA that can create hard triplets may increase the likelihood of task-irrelevant [27]. Additionally, even if hard triplets are extracted, when applying DA, key information related to scene discrimination may be ignored, and irrelevant information may be considered important [54], leading to distortion of hard triplets into easy triplets.

On the other hand, cross-entropy has the advantage of being easy to optimize and ensuring good performance without the need for exhaustive data mining [55]. However, the cross-entropy has drawbacks, such as sensitivity to noise labels [56], presence of adversarial samples [57], and poor margins [58], [59]. Due to these drawbacks, models trained with the cross-entropy loss may have low generalization ability. However, adding inductive bias through DA assuming the target environment (VPR environment) can improve the generalization performance of the model. The results in Table V indicate that when the conditions are the same except for the loss, i.e., second and sixth rows, or third and fifth rows, the performance of the Base model of LMCL is lower than that of WSTRL on all testsets, indicating that cross-entropy has lower generalization ability. However, when DA with VPR inductive bias is added, the generalization performance improves as seen in the improved performance on all testsets.

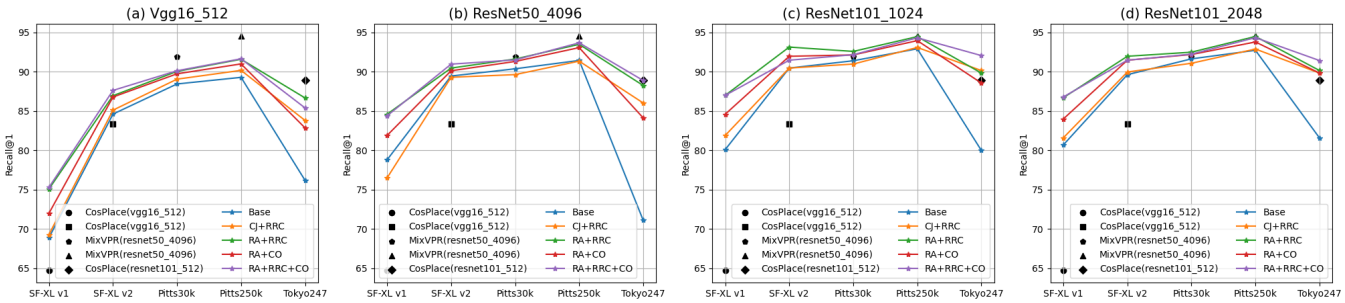

Fig. 2. Results on DA configuration of CosPlace using different backbones and descriptor dimensions.

TABLE VI

THE RECALL@1 RESULTS AND TRAINING TIME FOR THE RESNET101 MODEL TRAINED ON THE SF-XL PROCESSED DATASET.

Augmentation	SF-XL v1	SF-XL v1	Pitts30k	Pitts250k	Tokyo247	Training time
Base	80.10	90.47	91.40	92.92	80.00	1:10:56:27
CJ, RRC	81.90	90.47	90.98	93.09	90.16	2:03:20:24
RA, RRC	**87.00**	**93.14**	**92.59**	94.47	89.84	3:18:31:54
RA, RRC	84.60	91.97	92.14	93.96	88.57	2:10:57:57
RRC, RA, CO	**87.00**	91.47	92.17	94.32	**92.06**	2:05:39:45
State-of-the-Art	64.7	83.4	91.9	**94.6**	88.9	-

Generally Effective DA Configuration for VPR. The effectiveness of DA was confirmed in the classification models through our experiments. To obtain generalizable results, we conducted experiments by varying the backbones and descriptor dimensions. We trained models using the processed SF-XL dataset and applied RandAugment with a magnitude of 15 during the training process. Fig. 2 compares different augmentation configurations in respect of backbones and dimensions on the SF-XL v1, SF-XL v2, Pitts30k, Pitts250k, and Tokyo247 datasets. The black dots represent the performance of the state-of-the-art (SOTA) models for each dataset. The results indicate that the configuration of RandAugment and RandomResizeCrop led to the best performance generally. Table VI shows the quantitative results of the ResNet101 model with 1,024 dimensions, the same as Fig. 2 (c). The ResNet101 model with RandAugment, RandomResizeCrop and Cutout outperforms the SOTA performance in all datasets except for pitt250k, where it achieves the performance close to SOTA. The configuration of RandAugment and RandomResizeCrop also displays satisfactory performance that exceeds SOTA, but the training time is consuming, and the configuration of RandAugment, RandomResizeCrop and Cutout exhibits the best performance in the challenging dataset. These findings suggest that DA can improve the performance of VPR models, and the combination of RandAugment, RandomResizeCrop and Cutout is a promising choice for the VPR task.

V. CONCLUSIONS

In this work, we examined the DA impact per model architectures for VPR from previously unexplored angles. By designing a thorough empirical study, we revealed multiple essential insights: (1) The effect of data augmentation is not universally applicable to all VPR models, and even with the same DA, performance may or may not be improved depending on the learning method. (2) Even if the dataset, which already contains various realistic constraints, is used for learning, applying data augmentation in classification-based learning methods greatly improves performance. (3) Using VPR-inductive biased DA in conjunction with cross-entropy loss makes it easier to optimize VPR performance. Moreover, we obtained new state-of-the-art performance as a result of our study. Finally, we expect our extensive investigation has revealed crucial insights for future research directions for the corresponding research community.

REFERENCES

[1] X. Zhang, L. Wang, and Y. Su, "Visual Place Recognition: A Survey from Deep Learning Perspective," *Pattern Recognition*, vol. 113, p. 107760, 2021.

[2] C. Masone and B. Caputo, "A Survey on Deep Visual Place Recognition," *IEEE Access*, vol. 9, pp. 19 516–19 547, 2021.

[3] P. Newman and K. Ho, "SLAM-Loop Closing with Visually Salient Features," in *proceedings of the 2005 IEEE International Conference on Robotics and Automation*. IEEE, 2005, pp. 635–642.

[4] M. U. M. Bhutta, M. Kuse, R. Fan, Y. Liu, and M. Liu, "Loop-Box: Multiagent Direct SLAM Triggered by Single Loop Closure for Large-Scale Mapping," *IEEE transactions on cybernetics*, vol. 52, no. 6, pp. 5088–5097, 2020.

[5] N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the Performance of ConvNet Features for Place Recognition," in *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2015, pp. 4297–4304.

[6] G. Berton, R. Mereu, G. Trivigno, C. Masone, G. Csurka, T. Sattler, and B. Caputo, "Deep Visual Geo-Localization Benchmark," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5396–5407.

[7] F. Radenović, G. Tolias, and O. Chum, "Fine-tuning CNN Image Retrieval with No Human Annotation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 7, pp. 1655–1668, 2018.

[8] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN Architecture for Weakly Supervised Place Recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5297–5307.

[9] A. Ali-bey, B. Chaib-draa, and P. Giguère, "GSV-Cities: Toward Appropriate Supervised Visual Place Recognition," *Neurocomputing*, vol. 513, pp. 194–203, 2022.

[10] T. Weyand, I. Kostrikov, and J. Philbin, "PlaNet-Photo Geolocation with Convolutional Neural Networks," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*. Springer, 2016, pp. 37–55.

[11] G. Berton, C. Masone, and B. Caputo, "Rethinking Visual Geo-localization for Large-Scale Applications," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4878–4888.

[12] M. Izbicki, E. E. Papalexakis, and V. J. Tsotras, "Exploiting the Earth's Spherical Geometry to Geolocate Images," in *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, Proceedings, Part II*. Springer, 2020, pp. 3–19.

[13] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual Place Recognition: A Survey," *ieee transactions on robotics*, vol. 32, no. 1, pp. 1–19, 2015.

[14] A. Torii, R. Arandjelovic, J. Sivic, M. Okutomi, and T. Pajdla, "24/7 Place Recognition by View Synthesis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1808–1817.

[15] F. Maffra, Z. Chen, and M. Chli, "Viewpoint-Tolerant Place Recognition Combining 2D and 3D Information for UAV Navigation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2542–2549.

[16] D. Olid, J. M. Fácil, and J. Civera, "Single-View Place Recognition under Seasonal Changes," *arXiv preprint arXiv:1808.06516*, 2018.

[17] T. Kanji, "Self-localization from Images with Small Overlap," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4497–4504.

[18] A. Torii, J. Sivic, T. Pajdla, and M. Okutomi, "Visual Place Recognition with Repetitive Structures," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 883–890.

[19] M. Xu, S. Yoon, A. Fuentes, and D. S. Park, "A Comprehensive Survey of Image Augmentation Techniques for Deep Learning," *Pattern Recognition*, p. 109347, 2023.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[21] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-scale Image Recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[22] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going Deeper with Convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[24] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[25] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *arXiv preprint arXiv:2010.11929*, 2020.

[26] A. Kortylewski, B. Egger, A. Schneider, T. Gerig, A. Morel-Forster, and T. Vetter, "Empirically Analyzing the Effect of Dataset Biases on Deep Face Recognition Systems," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 2093–2102.

[27] C. Shorten and T. M. Khoshgoftaar, "A Survey on Image Data Augmentation for Deep Learning," *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.

[28] T. DeVries and G. W. Taylor, "Improved Regularization of Convolutional Neural Networks with Cutout," *arXiv preprint arXiv:1708.04552*, 2017.

[29] K. K. Singh and Y. J. Lee, "Hide-and-Seek: Forcing a Network to be Meticulous for Weakly-supervised Object and Action Localization," in *2017 IEEE international conference on computer vision (ICCV)*. IEEE, 2017, pp. 3544–3553.

[30] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random Erasing Data Augmentation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 13 001–13 008.

[31] P. Chen, S. Liu, H. Zhao, and J. Jia, "Gridmask Data Augmentation," *arXiv preprint arXiv:2001.04086*, 2020.

[32] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "ImageNet-trained CNNs are Biased Towards Texture; Increasing Shape Bias Improves Accuracy and Robustness," *arXiv preprint arXiv:1811.12231*, 2018.

[33] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "RandAugment: Practical Automated Data Augmentation with a Reduced Search Space," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 702–703.

[34] M. Wortsman, G. Ilharco, S. Y. Gadre, R. Roelofs, R. Gontijo-Lopes, A. S. Morcos, H. Namkoong, A. Farhadi, Y. Carmon, S. Kornblith *et al.*, "Model Soups: Averaging Weights of Multiple Fine-tuned Models Improves Accuracy Without Increasing Inference Time," in *International Conference on Machine Learning*. PMLR, 2022, pp. 23 965–23 998.

[35] Q. Zhang, Y. Xu, J. Zhang, and D. Tao, "ViTAEv2: Vision Transformer Advanced by Exploring Inductive Bias for Image Recognition and Beyond," *International Journal of Computer Vision*, pp. 1–22, 2023.

[36] H. Pham, Z. Dai, Q. Xie, and Q. V. Le, "Meta Pseudo Labels," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 11 557–11 568.

[37] H. Touvron, A. Vedaldi, M. Douze, and H. Jégou, "Fixing the Train-test Resolution Discrepancy," *Advances in neural information processing systems*, vol. 32, 2019.

[38] W. Wang, J. Dai, Z. Chen, Z. Huang, Z. Li, X. Zhu, X. Hu, T. Lu, L. Lu, H. Li *et al.*, "InternImage: Exploring Large-scale Vision Foundation Models with Deformable Convolutions," *arXiv preprint arXiv:2211.05778*, 2022.

[39] W. Su, X. Zhu, C. Tao, L. Lu, B. Li, G. Huang, Y. Qiao, X. Wang, J. Zhou, and J. Dai, "Towards All-in-one Pre-training via Maximizing Multi-modal Mutual Information," *arXiv preprint arXiv:2211.09807*, 2022.

[40] Y. Fang, W. Wang, B. Xie, Q. Sun, L. Wu, X. Wang, T. Huang, X. Wang, and Y. Cao, "EVA: Exploring the Limits of Masked Visual Representation Learning at Scale," *arXiv preprint arXiv:2211.07636*, 2022.

[41] W. Wang, H. Bao, L. Dong, J. Bjorck, Z. Peng, Q. Liu, K. Aggarwal, O. K. Mohammed, S. Singhal, S. Som *et al.*, "Image as a Foreign Language: BEiT Pretraining for All Vision and Vision-Language Tasks," *arXiv preprint arXiv:2208.10442*, 2022.

[42] Y. Wei, H. Hu, Z. Xie, Z. Zhang, Y. Cao, J. Bao, D. Chen, and B. Guo, "Contrastive Learning Rivals Msked Image Modeling in Fine-tuning via Feature Distillation," *arXiv preprint arXiv:2205.14141*, 2022.

[43] F. Li, H. Zhang, S. Liu, L. Zhang, L. M. Ni, H.-Y. Shum *et al.*, "Mask DINO: Towards a Unified Transformer-Based Framework for Object Detection and Segmentation," *arXiv preprint arXiv:2206.02777*, 2022.

[44] S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer, "Patch-NetVLAD: Multi-scale Fusion of Locally-global Descriptors for Place Recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 141–14 152.

[45] M. Leyva-Vallina, N. Strisciuglio, and N. Petkov, "Generalized Contrastive Optimization of Siamese Networks for Place Recognition," *arXiv preprint arXiv:2103.06638*, 2021.

[46] F. Yang, R. Hinami, Y. Matsui, S. Ly, and S. Satoh, "Efficient Image Retrieval via Decoupling Diffusion into Online and Offline Processing," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 9087–9094.

[47] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating Local Descriptors into a Compact Image Representation," in *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 2010, pp. 3304–3311.

[48] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "CosFace: Large Margin Cosine Loss for Deep Face Recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5265–5274.

[49] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6023–6032.

[50] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond Empirical Risk Minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[51] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[52] X. Wang, X. Han, W. Huang, D. Dong, and M. R. Scott, "Multi-Similarity Loss with General Pair Weighting for Deep Metric Learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5022–5030.

[53] Y. Ge, H. Wang, F. Zhu, R. Zhao, and H. Li, "Self-supervising Fine-grained Region Similarities for Large-scale Image Localization," in

Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16. Springer, 2020, pp. 369–386.

[54] Y. Tian, C. Sun, B. Poole, D. Krishnan, C. Schmid, and P. Isola, "What Makes for Good Views for Contrastive Learning?" *Advances in neural information processing systems*, vol. 33, pp. 6827–6839, 2020.

[55] M. Boudiaf, J. Rony, I. M. Ziko, E. Granger, M. Pedersoli, P. Piantanida, and I. B. Ayed, "A Unifying Mutual Information View of Metric Learning: Cross-entropy vs. Pairwise Losses," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI*. Springer, 2020, pp. 548–564.

[56] Z. Zhang and M. Sabuncu, "Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels," *Advances in neural information processing systems*, vol. 31, 2018.

[57] K. Nar, O. Ocal, S. S. Sastry, and K. Ramchandran, "Cross-Entropy Loss and Low-Rank Features Have Responsibility for Adversarial Examples," *arXiv preprint arXiv:1901.08360*, 2019.

[58] G. Elsayed, D. Krishnan, H. Mobahi, K. Regan, and S. Bengio, "Large Margin Deep Networks for Classification," *Advances in neural information processing systems*, vol. 31, 2018.

[59] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-Margin Softmax Loss for Convolutional Neural Networks," *arXiv preprint arXiv:1612.02295*, 2016.