2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

VDNA-PR: Using General Dataset Representations for Robust Sequential Visual Place Recognition

Benjamin Ramtoula*, Daniele De Martini+, Matthew Gadd+, and Paul Newman
Mobile Robotics Group (MRG), University of Oxford
{benjamin,daniele,mattgadd,pnewman}@robots.ox.ac.uk

***Abstract*— This paper adapts a general dataset representation technique to produce robust Visual Place Recognition (VPR) descriptors, crucial to enable real-world mobile robot localisation. Two parallel lines of work on VPR have shown, on one side, that general-purpose off-the-shelf feature representations can provide robustness to domain shifts, and, on the other, that fused information from sequences of images improves performance. In our recent work on measuring domain gaps between image datasets, we proposed a Visual Distribution of Neuron Activations (VDNA) representation to represent datasets of images. This representation can naturally handle image sequences and provides a general and granular feature representation derived from a general-purpose model. Moreover, our representation is based on tracking neuron activation values over the list of images to represent and is not limited to a particular neural network layer, therefore having access to high- *and* low-level concepts. This work shows how VDNAs can be used for VPR by learning a very lightweight and simple encoder to generate task-specific descriptors. Our experiments show that our representation can allow for better robustness than current solutions to serious domain shifts away from the training data distribution, such as to indoor environments and aerial imagery.**

***Index Terms*— Robotics, Place Recognition, Deep Learning**

I. Introduction

Visual Place Recognition (VPR) is an important task in robotics [1]. It consists of recognising whether a place has already been observed from an image depicting it. Doing so robustly and efficiently can help find loop closures for Simultaneous Localisation And Mapping (SLAM) or directly localise a robot. However, for a VPR system to be useful, it must be robust to viewpoint and appearance changes between different observations of the same place.

This robustness has often been achieved by training systems specifically for VPR; however, recently, more robust and general place recognition has been achieved by exploiting general-purpose feature representations [2]. Another parallel avenue for improving VPR performance is to leverage an intrinsic characteristic of the robotics settings: robots continuously stream data, allowing the application of sequences of images to perform place recognition [3].

On the other hand, a recent research area is studying general tools to measure custom domain gaps between image datasets [4]. Here, the idea is to generate a general granular representation of an image dataset and to compare these representations through task-dependent distances. The two aforementioned key insights to improve VPR are naturally handled in these approaches: representations are selected to be general and robust to domain variations, and representations are generated for an arbitrary number of images.

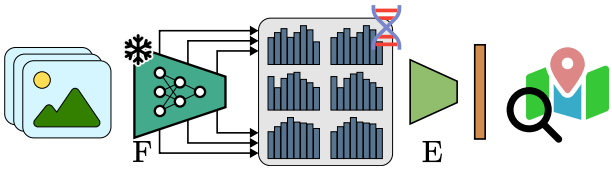

Fig. 1. *VDNA-PR overview*. As in other sequence-based works, we solve VPR by building and matching representations for image sequences along driven trajectories. We rely on VDNA representations [4], which were originally introduced to measure domain gaps between datasets. They consist of histograms that describe activations observed when passing images through a frozen self-supervised feature extractor F. Importantly, VDNAs keep track of activations for neurons throughout all layers of the network, keeping a general and granular multi-level representation. To generate more practical descriptors specifically for VPR, we propose an encoder E to encode VDNAs into descriptors that can efficiently be compared with traditional VPR techniques.

Thus, this work proposes building a VPR system using methodologies borrowed from dataset domain-gap measurements. A high-level overview is depicted in Fig. 1. Specifically, we treat sequences of consecutive images as individual datasets and generate general representations that allow for general-purpose comparisons. For VPR, however, we need to produce practical descriptors that can be compared at scale. Hence, we learn a "VPR encoder" on top of the general-purpose representation that produces a small descriptor suitable for the task. Framing the VPR problem in this fashion ensures we have access to a general, robust, and granular representation on top of which to perform VPR, and do not have any arbitrary assumption on the choice of the feature extractor layer.

Our principal contributions are

1) A novel application of deep neural dataset-to-dataset comparison mechanisms to sequence-based VPR,
2) An architecture for learning a more practical descriptor for VPR, and
3) Experiments showing improved generalisation under domain shift versus competitors.

* Corresponding author.
+ Equal contribution.

This work was supported by EPSRC Programme Grant "From Sensing to Collaboration" (EP/V000748/1), the EPSRC Centre for Doctoral Training in Autonomous Intelligent Machines and Systems [EP/S024050/1], and Oxa.

II. Related Works

VPR is classically set as an image retrieval problem; given a query image and a reference database, its closeness in geometric space to any image in the database is directly related to a similarity measure in an embedding space [1], extracted either from local or global features within the image. Similarly, images can be processed singularly or sequentially, i.e. exploiting the inherent temporal correlation to create a stronger representation [3]. This work is related to both challenges of selecting the right representation and aggregating temporal information; thus, we will frame it in both contexts in the following.

A. Selecting the right representation

With the advent of Deep Learning (DL) methodologies and extensive VPR datasets, we have seen an almost complete shift from handcrafted features to learned ones, with substantial performance improvements. The common approach is to utilise a feature extractor and fine-tune it while training a feature aggregator on top of the learned features to perform VPR. For instance, NetVLAD [5] follows this approach, introducing a trainable variant of the VLAD [6] descriptor. This methodology has been very successful and inspired several improvements [7], [8] and applications to different data types [9], [10].

Other methods focus on the extraction and aggregation of information from the images. Examples are MixVPR [11] and R2Former [12]. MixVPR uses flattened features from intermediate layers of a pre-trained backbone and incorporates spatial relationships through Feature-Mixer blocks. R2Former, instead, uses a transformer architecture to encode the image patches into global and local descriptors and uses the first for global retrieval and the local ones for fast reranking through a correlation operation.

GeM [13] and consequently Cosplace [14] focused on data and training procedures, simplifying the aggregation to a learned pooling operation. Indeed, the first applied a contrastive approach where positives and negatives are selected through a Structure from Motion (SfM) technique, whereas the second sets the VPR training as a classification problem on a novel extensive dataset.

Finally, a very recent work, AnyLoc [2], proposes the use of foundational models, in particular DINOv2 [15], pre-trained in a self-supervised fashion and with no prior knowledge about the task of VPR, to extract generic and robust features. These features are then exploited for the specific task of VPR using an aggregation method, such as VLAD [5] or GeM [13]. In this way, AnyLoc showed state-of-the-art generalisability on multiple domains.

In this, we are most similar to this last approach, in that we exploit a self-supervised pre-trained model to create a representation of the data robust to domain changes.

B. Using Sequential information

Sequence-based methods leverage temporal information in the query and database to discover more robust matches for VPR. We refer the reader to Mereu *et al*. for a detailed taxonomy [3]. Generally, we can divide such methods into two main families: sequence-matching and sequence-descriptor methods. The firsts use separate similarities for each query image in a query sequence to match *segments* of high similarity in the database. Methods such as SeqSLAM [16] and [17] belong to this family, where the first samples and validates feasible trajectories in the database to assess the most likely one and the second uses a matrix value decomposition to eliminate possible noisy detections.

Sequence-descriptor methods, instead, propose to generate a descriptor for an entire image sequence and directly match segments through it. TimeSformer [18] explores early-fusion approaches on image-patches descriptors through a transformer architecture. SeqNet [19], instead, has a hybrid approach as it uses a convolutional backbone to extract image descriptors, which are used to calculate a sequence-based descriptor for rough sequence matching and to refine this latter through image-to-image similarity. Mereu *et al*. [3] tackle sequential embedding through an intermediate fusion of single-frame descriptors with a SeqVLAD aggregator, a NetVLAD [5] generalisation to handle multiple images.

Our proposed approach belongs to this last family of methodologies, as we too combine the information from the whole sequence into a single descriptor, and then use it for the matching phase. We base our approach on a previous general dataset representation and comparison work.

C. Representation scope & layer combinations

Our approach benefits from combining representations *across* the feature extractor, and we show experimentally in Section V that multi-level concepts access in different layers can help generalise better across deployment domains. Related to this, is the "filter-early, match-late" work of [20] with the guiding principle that simple visual features that detract from a network's utility for place recognition across changing environmental conditions are removed and, as per [21], that late layers are more invariant to viewpoint changes. In our recent work introducing Visual Distribution of Neuron Activations (VDNA) [4] to represent image datasets, we also highlighted the power of granular representations over multiple feature extractor layers to control what concepts the representation comparison should be sensitive to, such as when measuring the realism of synthetic images. Our combination of layers is also related to a weighted concatenation of convolutional features across layers in [22].

D. Measuring domain gaps between datasets

Quantifying the domain gap between datasets is possible with techniques such as Fréchet Inception Distance (FID) [23], Kernel Inception Distance (KID) [24], bag-of-prototypes [25], and Earth-Mover Distance (EMD) [26] applied to VDNAs [4].

FID [23] embeds images with a specific layer of the Inception-v3 network [27], fitting a multivariate Gaussian, and the Fréchet Distance (FD) between such distributions from different datasets is computed. Alternatively, the

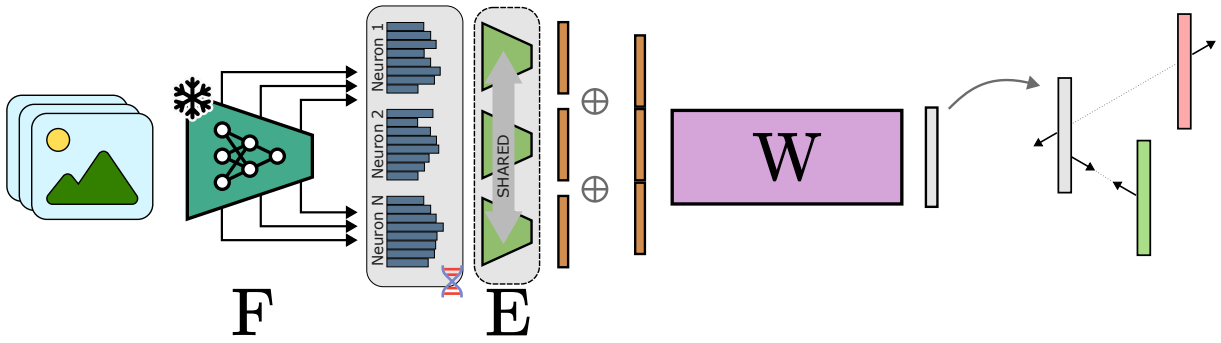

Fig. 2. *Overview of VDNA-PR training*. As in Fig. 1, as a sequence of images passes through a pre-trained frozen feature extractor F, histograms tracking neuron-wise activations constitute a VDNA ⧖. The histogram corresponding to each neuron has 500 bins, and a small 1D CNN encoder E maps each histogram to a lower dimensional vector of length 4 (with shared weights across the 9216 neurons). The concatenation of these length-4 features is of length 36864 and is then itself passed through a linear layer W to be reduced in dimension and to form the final representation. It is on this representation that we perform contrastive learning with triplet losses as is common in place recognition. At test-time on different domains, we remove the linear layer W which has learned specific features of the training domain, and use concatenations of encoded histogram features from selected neurons. With this training, we therefore learn neuron-wise descriptors that can be used and combined for VPR.

KID [24] computes the squared maximum mean discrepancy between Inception-v3 embeddings.

In addition to relying on a more comprehensive and granular representation, VDNA [4] was demonstrated in [28] to better predict localisation precision between experiences with serious seasonal domain shifts, particularly in comparison to FID [23], and so is the representation on which we base our proposed place recognition system. This method, as illustrated and described in Fig. 1, populates neuron-wise histograms with the activation levels of a pre-trained feature extractor as images pass through the network, across all layers. Histogram comparisons (e.g., by the EMD) can be aggregated and combined with neuron-wise emphasis (or combinatorially across neurons) to fine-tune the dataset-to-dataset comparison across attributes of interest and at various levels of features in deep networks.

III. Methodology

Expanding on Fig. 1, our *VDNA-PR* training approach is shown in more detail in Fig. 2, with a focus on the lightweight encoder module E learned to produce practical descriptors for the VPR problem.

A. Visual DNAs creation and comparisons

We chose to build our representation on top of VDNA [4] representations of images from sequences for which we want to produce a VPR descriptor.

In this, given a sequence of L images $\mathcal{I}$, where $L \geq 1$, we can pass the images through a frozen feature extractor (❄ in Fig. 2). During the forward pass, we can keep track of the outputs of multiple layers of the model, and decompose each of them into multiple 1D "neuron activations". The idea of VDNA is to form a granular representation by gathering distributions of many neuron activations. This is done by collecting neuron activations into histograms $\mathcal{H}_i$, one for each i of the N neurons across all layers. Each histogram comprises b bins, giving a total descriptor size of $N \times b$ which does not depend on the sequence length L (this being one of the core motivations of the original approach in [4]). Indeed, VDNAs can inherently represent multiple images simultaneously as they are designed as compact representations of datasets.

As VDNAs are composed of histograms, the original measure of domain gaps between two datasets represented as VDNAs is based on the Earth-Mover Distance (EMD) [26]. Unfortunately, while this distance has long been useful in image retrieval [26], no efficient implementation is present in commercial-grade image-retrieval database technologies, beyond approximations as in [29].

For these reasons, we propose an encoder model, trainable and applied to the task of VPR, detailed in the following. The output of this module is a lower-dimensional vector upon which we can compute L_2 distances, at scale.

B. VPR Encoder Model

In typical foundational models, the number of neurons, N, may be extremely large. For instance, DINOv2's Vision Transformer (ViT) used in this work contains 9216 neurons[1]. This fact further supports our need not to directly use the histograms $\mathcal{H}_i$. Indeed, this descriptor dimension challenges caching it in memory and thus efficient comparison, essential in real-time applications such as robotics. To tackle these issues and those raised in Section III-A, we design an encoder model to encode the histograms, normalised, into a latent space where we can efficiently use a vector distance.

This encoder network is shown as the shared modules in Fig. 2. We treat each histogram $\mathcal{H}_i$ as sequential data and apply a 1D Convolutional Neural Network (CNN) composed of six convolutional layers followed by three linear layers. This reduces the dimensionality of each histogram from b to h. After extensive experimentation, we set h to 4. After encoding, the extracted embeddings are normalised singularly and concatenated into an embedding e of dimension Nh. Now, this descriptor can more efficiently be stored in memory and compared.

[1]For comparison, *LeNet5* consists of only 84 neurons which with 500 bins per histogram as in Section IV-A already leads to a 42000-dimensional vector, in excess of VGG-16/NetVLAD [30], [5]

TABLE I. Details of datasets used. MSLS (train) and (val) are used during training, and all others are used for evaluation. The numbers of database and query sequences are given for a sequence length of 5 frames.

Name	Domain	# of db seq.	# of query seq.	Loc. req.
MSLS (train)	Urban	733048	393201	25 meters
MSLS (val)	Urban	8125	5752	25 meters
MSLS (test)	Urban	13584	7964	25 meters
St Lucia	Urban	1545	1460	25 meters
Pitts-30k	Urban	6664	4542	25 meters
RobotCar 1	Urban	3615	3292	25 meters
RobotCar 2	Urban	3919	3687	25 meters
RobotCar 3	Urban	3628	3917	25 meters
Baidu Mall	Indoor	677	1356	10 meters
Gardens Point	Indoor	196	196	2 frames
17 Places	Indoor	334	334	5 frames
VPAIR	Aerial	2702	2702	3 frames

C. Training approach

Given a labelled dataset of image sequences, we aim to train the encoder model to achieve high-grade accuracy while generalising to different domains. To achieve this, we take inspiration from CosPlace training [14], which uses data-specific linear layers on top of the representation of interest that are then discarded after training. We apply a linear layer W, parameterised by a trainable weight matrix $\mathtt{W} \in Nh \times d$, to project the descriptor into a lower dimensionality. We use these descriptors to produce triplet losses [31], which serve to train the encoder E and linear layer W. We use a triplet loss as it is popular for VPR, but would expect contrastive losses [32], [33] to also work well.

This training will encourage the 1D CNN to produce good encodings of histograms that can be reused on different domains while learning the specificities of the training domain in the linear layer that can be discarded later.

IV. Experimental Setup

A. VDNAs

We use DINOv2 ViT-B/14 [15] with 12 layers of 768 neurons each to generate VDNAs. For each neuron, we collect a histogram of $b = 500$ bins, which we find to be sufficient to finely approximate distributions of activations. We chose this model as AnyLoc [2] showed that DINOv2 achieves great generalisability to different scenes and domains thanks to its training regime.

B. Data

To validate our approach, we use one training and several testing datasets summarised in Table I to evaluate our generalisation capabilities. In particular, we selected MSLS [34] to train our neuron encoder module as it contains a large scale of training samples from several urban environments worldwide and with diverse conditions. Moreover, the data samples are collected as sequences, so they apply to our proposed methodology.

We evaluate VPR performance on a diverse set of domains, including the **Baidu Mall Benchmark** [35], **VPAIR** [36], **17 Places** [37], and **Gardens Point** [38]. Figure 3 shows examples from the datasets used, which show a high degree of domain variability, from indoor to urban to overhead

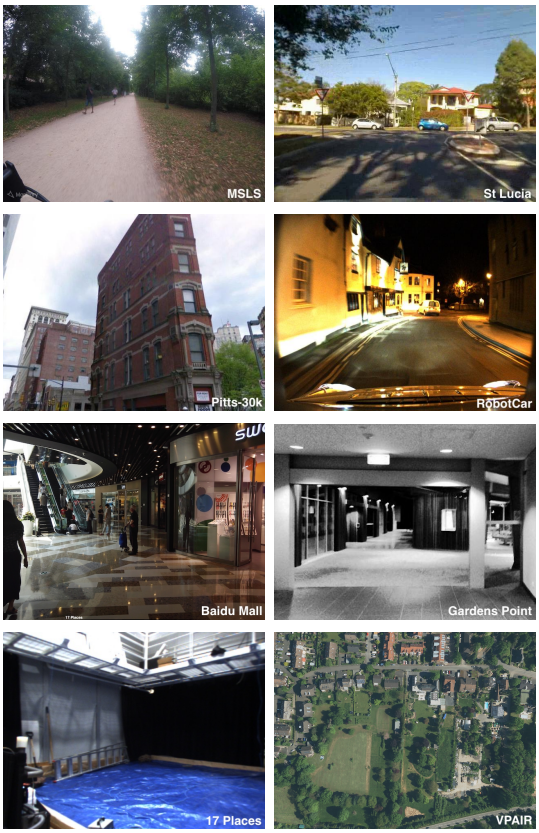

Fig. 3. Example images from datasets used in this study. We evaluate generalisation capability from training VPR approaches on MSLS, and using them in other urban and indoor environments, as well as on aerial imagery.

imagery. We consider **Pittsburgh-30k** [39], **St Lucia** [40], and **Oxford** [41] to have significant domain overlap with **MSLS** [34] as they also contain images from urban environments. In **Oxford**, we use the three train/val/test splits used in [3], corresponding to:

- **RobotCar1**: queries: 2014-12-17-18-18-43 (winter night, rain); database: 2014-12-16-09-14-09 (winter day, sun).
- **RobotCar2**: queries: 2015-02-03-08-45-10 (winter day, snow); database: 2015-11-13-10-28-08 (fall day, overcast).
- **RobotCar3**: queries: 2014-12-16-18-44-24 (winter night); database: 2014-11-18-13-20-12 (fall day).

For the training and evaluation datasets, splits and labelling are taken from SeqVLAD [3] and Anyloc [2]. Particularly, we did not consider the distractor images in **VPAIR** as they are not collected as sequences.

In our experiments, we employ a sequence length of 5 images, which we keep consistent with all our baselines. Nevertheless, we ablate over this number to investigate how a network pre-trained on five images will behave when used with a different number.

C. Training details

For training our encoder, we use similar mining methods from previous works [3], [34]. We cache 1000 query sequences alongside 5000 negatives to perform hard example mining. We also include the 500 hardest previous triplets and

refresh the cache every 1500 triplets. Each triplet contains a query, its positive, and 5 negatives. We optimise the encoder E and linear layer W weights using the AdamW [42] optimiser with weight decay.

D. Baseline

All approaches considered rely on the DINOv2 ViT-B/14 [15] backbone. For all *SeqVLAD* variants, we use features from layer 12. Our baselines and variants include:

1) `SeqVLAD`: The method and general architecture as described in [3], but with DINOv2 substituted as the backbone. In this variant, the backbone and the SeqVLAD layer are both fine-tuned on MSLS [34], as is done in [3].
2) $\texttt{SeqVLAD}_{\texttt{frozen}}$: The same setup as `SeqVLAD`, but in this case, only the SeqVLAD aggregation layer is fine-tuned. The backbone is kept frozen.
3) $\texttt{SeqVLAD}_{\texttt{calib.}}$: Using SeqVLAD calibrated to the evaluation domain. This is performed similarly to the use of Hard Assignment VLAD [43] in AnyLoc [2] but with the SeqVLAD aggregation over multiple images. The backbone is not fine-tuned, and centroids are initialised using clustering on the database images.
4) $\texttt{VDNA-PR}_{\texttt{W}}$: Using the output of the linear layer W used during the training of *VDNA-PR*.
5) $\texttt{VDNA-PR}_{\texttt{all}}$: Using the output of the *VDNA-PR* encoder E (i.e. with the W linear layer omitted) when *all* 9216 neurons have been projected to length 4 and concatenated. This is equivalent to $\texttt{VDNA-PR}_{\texttt{W}}$ with the linear layer used during training on MSLS removed.
6) $\texttt{VDNA-PR}_{12}$: Using the *VDNA-PR* encoder E output but only keeping descriptors from neurons in the twelfth layer.
7) $\texttt{VDNA-PR}_{9:12}$, $\texttt{VDNA-PR}_{10:12}$, and $\texttt{VDNA-PR}_{11:12}$: Similar to $\texttt{VDNA-PR}_{12}$ but in these cases with a *combination* of layers as subscripted.

Baselines (1-3) produce a descriptor of length 49152, (4) produces a descriptor of length 128, and (5-7) produce descriptors of length $3072\times\#\texttt{layers}$.

E. Performance metrics

To assess place recognition performance, we use the Recall@N (R@N) metric. It consists in the percentage of successful queries in the set of query images which result in at least 1 correct localisation result when the N closest reference images are retrieved from the reference map or database according to VPR descriptors. "Correctness" of a localisation match is determined by a ground-truth distance – these thresholds are detailed in Table I.

V. RESULTS

A. Effect of layer selection

In Fig. 4, we first observe how using a *VDNA-PR* descriptor composed of histogram encodings of neurons from each backbone layer affects performance on all datasets considered.

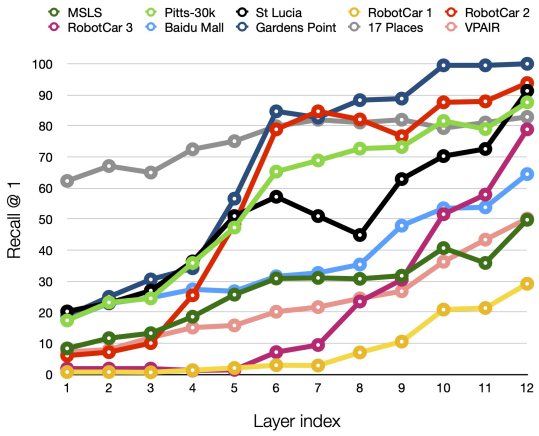

Fig. 4. Recall@1 when evaluating VPR performance using *VDNA-PR* descriptors from each layer of DINOv2 for all datasets considered.

We obtain large variations depending on the datasets. For example, decent performance on **17 Places** can be obtained even with early layers, but not on **RobotCar** sequences. Moreover, a later layer does not always lead to improved performance. For example, we obtained better results on **St Lucia** with neurons from layer 6 than from layer 8. These observations support the value of having a descriptor containing information from all layers, as not all datasets will do best with the same levels of features. Overall, the last layers consistently lead to the best results for all datasets. Hence, in our following experiments, we include results based on neurons from combinations of the last layers.

B. Benchmarking with domain shifts

a) Training domain: Table II (*Urban*) shows the R@1 and R@5 performance of our baselines and approaches in the same domain as they were trained, in urban environments. On the test set of MSLS, `SeqVLAD` with its fine-tuned backbone performs best. $\texttt{SeqVLAD}_{\texttt{frozen}}$ and $\texttt{VDNA-PR}_{\texttt{W}}$ also perform well, thanks to their training on MSLS and despite keeping their backbones frozen. Other *VDNA-PR* variants without the linear layer and $\texttt{SeqVLAD}_{\texttt{calib.}}$ do not benefit from a strong specialisation on MSLS, and these approaches perform reasonably well, but noticeably worse than the ones specialised on MSLS.

However, in other urban datasets, `SeqVLAD` does not dominate as strongly. It particularly struggles on **RobotCar1**, performing similarly to *VDNA-PR* techniques without W and $\texttt{SeqVLAD}_{\texttt{calib.}}$, whereas $\texttt{SeqVLAD}_{\texttt{frozen}}$ maintains good performance on other urban domain datasets. This already indicates a limitation in robustness by using an approach too strongly trained on a dataset, despite MSLS' large scale and diversity. The features from the selected layer of the backbone might already be well-suited for VPR in urban settings. In this case, keeping the backbone frozen allows $\texttt{SeqVLAD}_{\texttt{frozen}}$ to outperform `SeqVLAD` on other urban datasets by maintaining these features. If the features from a layer are already suitable for VPR within a domain, we also expect $\texttt{SeqVLAD}_{\texttt{frozen}}$ to be more descriptive than *VDNA-PR* variants as VDNAs do not have information on the joint distribution of all neuron activations within a layer.

TABLE II. Performance comparison on benchmark environments with urban and indoor environments, and aerial imagery. Please refer to the baseline/ablation definitions in Section IV-D for details of methods (row names).

	Urban												Indoor						Aerial	
	MSLS		St Lucia		Pitts-30k		RobotCar 1		RobotCar 2		RobotCar 3		Baidu Mall		Gardens Point		17 Places		VPAIR	
Methods	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5
SeqVLAD	**87.9**	**91.7**	**99.7**	**99.9**	92.8	96.1	35.5	45.1	97.0	98.1	68.3	76.9	52.1	69.5	89.8	95.9	81.1	86.8	20.5	30.5
SeqVLAD$_{\texttt{frozen}}$	82.2	87.6	99.5	**99.9**	**93.4**	**96.6**	**66.0**	**74.1**	**98.3**	**98.9**	**93.3**	**96.3**	63.5	80.4	**100.0**	**100.0**	**83.5**	**89.8**	44.4	57.2
SeqVLAD$_{\texttt{calib.}}$	44.7	57.2	62.6	76.8	76.7	89.3	33.0	51.4	86.1	95.1	58.0	77.1	50.9	73.4	97.4	**100.0**	81.7	87.7	20.7	34.5
VDNA-PR$_{\texttt{W}}$	74.1	84.0	92.1	95.3	81.2	93.2	34.7	49.9	92.2	96.8	69.8	86.2	37.1	63.8	88.3	95.4	78.7	88.6	24.8	41.7
VDNA-PR$_{\texttt{all}}$	38.9	46.0	79.2	86.1	81.7	88.7	8.6	19.6	91.0	94.5	51.9	72.4	49.7	72.8	99.0	**100.0**	80.2	**89.8**	40.4	50.4
VDNA-PR$_{12}$	49.8	58.5	91.3	95.1	87.6	94.4	29.2	40.6	93.8	96.5	78.9	88.1	64.5	**81.5**	**100.0**	**100.0**	82.9	**89.8**	50.2	63.5
VDNA-PR$_{9:12}$	45.4	52.6	86.1	91.7	85.4	92.0	30.6	42.5	94.6	97.3	78.5	88.4	66.0	81.2	**100.0**	**100.0**	82.3	89.2	**51.3**	63.2
VDNA-PR$_{10:12}$	46.0	54.1	87.1	92.2	86.0	92.6	29.9	41.4	94.9	97.4	77.9	87.8	**66.2**	81.3	99.5	**100.0**	82.0	89.2	51.1	63.7
VDNA-PR$_{11:12}$	46.7	54.5	88.6	93.6	86.2	92.9	28.3	40.9	93.7	96.7	78.3	87.9	65.4	81.4	**100.0**	**100.0**	82.9	89.5	51.0	**64.3**

TABLE III. Performance comparison on **VPAIR** with varying imbalanced test-time sequence lengths. All approaches were trained on sequences of length 5 for database and query sequences. Here we denote test-time sequence lengths as database/queries.

	1/5		5/1		5/5	
Methods	R@1	R@5	R@1	R@5	R@1	R@5
SeqVLAD	17.31	33.5	15.4	24.7	20.5	30.5
SeqVLAD$_{\texttt{frozen}}$	38.8	59.0	39.0	51.7	44.4	57.2
SeqVLAD$_{\texttt{calib.}}$	10.3	24.9	8.8	20.3	20.7	34.5
VDNA-PR$_{12}$	**45.4**	**66.8**	**44.7**	**58.0**	**50.2**	**63.5**

TABLE IV. Performance comparison on **VPAIR** with increasing test-time sequence lengths. All approaches were trained on sequences of length 5 for database and query sequences.

	seq. len. 5		seq. len. 15		seq. len. 25	
Methods	R@1	R@5	R@1	R@5	R@1	R@5
SeqVLAD	20.5	30.5	27.9	35.9	32.2	39.0
SeqVLAD$_{\texttt{frozen}}$	44.4	57.2	55.4	61.8	61.7	65.8
SeqVLAD$_{\texttt{calib.}}$	20.7	34.5	27.1	35.8	34.8	41.2
VDNA-PR$_{12}$	**50.2**	**63.5**	**61.2**	**66.7**	**64.5**	**68.0**

b) Shifted domains: Table II (*Indoor & Aerial*) shows the R@1 and R@5 performance of our system on domains different from training: indoors with **Baidu Mall**, **Gardens Point** and **17 Places**, and on aerial imagery with **VPAIR**. What we see is that our *VDNA-PR* variants are typically more resilient than other methods, for these datasets. In **Gardens Point** and **17 Places**, we are broadly in line with SeqVLAD$_{\texttt{frozen}}$ but outstrip it for **Baidu Mall** and **VPAIR**.

Our VDNA-PR$_{\texttt{W}}$ variant struggles in these non-urban domains. This is to be expected as we have fine-tuned the linear layer W on **MSLS** data in VDNA-PR$_{\texttt{W}}$, and as a result of this urban specialisation, VDNA-PR$_{\texttt{W}}$ suffers in non-urban environments. On the other hand, VDNA-PR$_{\texttt{all}}$, which simply removes the linear layer from VDNA-PR$_{\texttt{W}}$, performs worse in urban domains, but is much more robust to other domains. In fact, the *VDNA-PR* variants lacking the linear layer W are *the most* robust in these datasets, outperforming *SeqVLAD* baselines. Therefore, we can argue that the fine-tuning of SeqVLAD and SeqVLAD$_{\texttt{frozen}}$ on MSLS, despite the foundation model's self-supervised representations, leads to domain shift susceptibility, and in this context our VDNA representation and careful training and then dismantling of our VPR system immunises us from this to a good extent.

We also note that SeqVLAD$_{\texttt{calib.}}$ always performs worse than SeqVLAD$_{\texttt{frozen}}$, suggesting that the aggregation layer optimisation with a frozen backbone still allowed for some robustness to domain shifts.

Finally, in the case of **Baidu Mall** and **VPAIR**, these results show that incorporating information across neural network layers as is natural for *VDNA-PR* is important, where we see that VDNA-PR$_{10:12}$ is the most performant in **Baidu Mall** for R@1 while VDNA-PR$_{9:12}$ and VDNA-PR$_{11:12}$ are the most performant in R@1/5 respectively for **VPAIR**. Indeed, considering R@5 for **17 Places** and **Gardens Point**, we see that incorporating all neurons is beneficial.

Layer combinations considered in this work are arbitrarily chosen and are unlikely to focus on the best set of neurons for each domain. However, these first results are encouraging signals that future work identifying how to select relevant neurons for a given domain could make *VDNA-PR* an even stronger solution for robust VPR.

C. Test-time sequence length variations

Furthermore, we investigate the effect of having varying numbers of images for the database and query descriptors, in particular for the **VPAIR** dataset. For *VDNA-PR*, the inputs to the encoder E are normalised histograms regardless of the number of images. This allows to make descriptors consistent for arbitrary numbers of images.

As with SeqVLAD, we can verify this natural robustness when having 1 or 5 sequence lengths in Table III. The small variations in performance observed when using 1 and 5 database or query images can be attributed to less informative descriptors due to the reduced number of images used.

In Table IV, we also observe that increasing the number of frames is handled naturally and allows for improved performance thanks to more informative descriptors.

VI. CONCLUSIONS

We have presented a new approach to VPR based on a general dataset representation called Visual Distribution of Neuron Activations. This representation lends itself naturally to representing sequences and is convenient for combining representations at different layers. We showed improved resilience of localisation performance to serious domain shifts in non-urban scenes, making progress in the area of universal place recognition from a foundation model basis.

In the future, we will explore the potential of this method with unsupervised *domain calibration* – where a search for responsive neurons may replace difficult gradient-based fine-tuning of aggregation layers. This could be used on database images to calibrate attention on neurons from which to produce a VPR descriptor.

REFERENCES

[1] 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, Feb. 2016.

[2] N. Keetha, A. Mishra, J. Karhade, K. M. Jatavallabhula, S. Scherer, M. Krishna, and S. Garg, "AnyLoc: Towards Universal Visual Place Recognition," *arXiv*, 2023.

[3] R. Mereu, G. Trivigno, G. Berton, C. Masone, and B. Caputo, "Learning Sequential Descriptors for Sequence-Based Visual Place Recognition," *IEEE Robotics and Automation Letters*, 2022.

[4] B. Ramtoula, M. Gadd, P. Newman, and D. De Martini, "Visual DNA: Representing and Comparing Images Using Distributions of Neuron Activations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 11 113–11 123.

[5] 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.

[6] 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.

[7] J. Zhang, Y. Cao, and Q. Wu, "Vector of Locally and Adaptively Aggregated Descriptors for Image Feature Representation," *Pattern Recognition*, vol. 116, p. 107952, 2021.

[8] J. Yu, C. Zhu, J. Zhang, Q. Huang, and D. Tao, "Spatial pyramid-enhanced NetVLAD with weighted triplet loss for place recognition," *IEEE transactions on neural networks and learning systems*, vol. 31, no. 2, pp. 661–674, 2019.

[9] Ş. Săftescu, M. Gadd, D. De Martini, D. Barnes, and P. Newman, "Kidnapped radar: Topological radar localisation using rotationally-invariant metric learning," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4358–4364.

[10] M. A. Uy and G. H. Lee, "PointNetVLAD: Deep point cloud based retrieval for large-scale place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4470–4479.

[11] A. Ali-Bey, B. Chaib-Draa, and P. Giguere, "MixVPR: Feature Mixing for Visual Place Recognition," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 2998–3007.

[12] S. Zhu, L. Yang, C. Chen, M. Shah, X. Shen, and H. Wang, "R2former: Unified retrieval and reranking transformer for place recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 19 370–19 380.

[13] 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.

[14] 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.

[15] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, "DINOv2: Learning Robust Visual Features without Supervision ," *arXiv preprint arXiv:2304.07193*, 2023.

[16] M. J. Milford and G. F. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights," in *IEEE international conference on robotics and automation*, 2012.

[17] K. L. Ho and P. Newman, "Detecting Loop Closure with Scene Sequences," *International journal of computer vision*, vol. 74, pp. 261–286, 2007.

[18] G. Bertasius, H. Wang, and L. Torresani, "Is Space-Time Attention All You Need for Video Understanding? ," in *ICML*, vol. 2, no. 3, 2021, p. 4.

[19] S. Garg and M. Milford, "SeqNet: Learning Descriptors for Sequence-based Hierarchical Place Recognition ," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4305–4312, 2021.

[20] S. Hausler, A. Jacobson, and M. Milford, "Filter Early, Match Late: Improving Network-Based Visual Place Recognition," in *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2019, pp. 3268–3275.

[21] 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.

[22] J. Mao, X. Hu, X. He, L. Zhang, L. Wu, and M. J. Milford, "Learning to Fuse Multiscale Features for Visual Place Recognition," *IEEE Access*, vol. 7, pp. 5723–5735, 2018.

[23] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.

[24] M. Bińkowski, D. J. Sutherland, M. Arbel, and A. Gretton, "Demystifying MMD GANs," *arXiv preprint arXiv:1801.01401*, 2018.

[25] W. Tu, W. Deng, T. Gedeon, and L. Zheng, "A Bag-of-Prototypes Representation for Dataset-Level Applications," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 2881–2892.

[26] Y. Rubner, C. Tomasi, and L. J. Guibas, "The Earth Mover's Distance as a Metric for Image Retrieval," *International journal of computer vision*, vol. 40, pp. 99–121, 2000.

[27] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[28] M. Gadd, B. Ramtoula, D. De Martini, and P. Newman, "What you see is what you get: Experience ranking with deep neural dataset-to-dataset similarity for topological localisation," in *International Symposium on Experimental Robotics (ISER)*, 2023.

[29] K. Atasu and T. Mittelholzer, "Linear-Complexity Data-Parallel Earth Mover's Distance Approximations," in *International Conference on Machine Learning*. PMLR, 2019, pp. 364–373.

[30] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

[31] E. Hoffer and N. Ailon, "Deep metric learning using Triplet network," in *Similarity-Based Pattern Recognition: Third International Workshop, SIMBAD 2015, Copenhagen, Denmark, October 12-14, 2015. Proceedings 3*. Springer, 2015, pp. 84–92.

[32] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation Learning with Contrastive Predictive Coding ," *arXiv preprint arXiv:1807.03748*, 2018.

[33] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations ," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[34] F. Warburg, S. Hauberg, M. Lopez-Antequera, P. Gargallo, Y. Kuang, and J. Civera, "Mapillary Street-Level Sequences: A Dataset for Lifelong Place Recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[35] X. Sun, Y. Xie, P. Luo, and L. Wang, "A Dataset for Benchmarking Image-Based Localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7436–7444.

[36] M. Schleiss, F. Rouatbi, and D. Cremers, "VPAIR–Aerial Visual Place Recognition and Localization in Large-scale Outdoor Environments," *arXiv preprint arXiv:2205.11567*, 2022.

[37] R. Sahdev and J. K. Tsotsos, "Indoor Place Recognition System for Localization of Mobile Robots," in *2016 13th Conference on computer and robot vision (CRV)*. IEEE, 2016, pp. 53–60.

[38] A. Glover, "Day and Night, Left and Right," Mar. 2014. [Online]. Available: https://doi.org/10.5281/zenodo.4590133

[39] 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.

[40] M. Warren, D. McKinnon, H. He, and B. Upcroft, "Unaided Stereo Vision Based Pose Estimation," in *Proceedings of the 2010 Australasian Conference on Robotics and Automation*. Australian Robotics & Automation Association, 2010, pp. 1–8.

[41] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The Oxford RobotCar dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 2017.

[42] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," in *International Conference on Learning Representations*, 2019.

[43] R. Arandjelovic and A. Zisserman, "All About VLAD," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1578–1585.