AmsterTime: A Visual Place Recognition Benchmark Dataset for Severe Domain Shift

Burak Yildiz*, Seyran Khademi*, Ronald Maria Siebes† and Jan van Gemert*
*Delft University of Technology, The Netherlands
Emails: {b.yildiz,s.khademi,j.c.vangemert}@tudelft.nl
†Vrije Universiteit Amsterdam, The Netherlands
Email: r.m.siebes@vu.nl

***Abstract*—We introduce AmsterTime: a challenging dataset to benchmark visual place recognition (VPR) in presence of a severe domain shift. AmsterTime offers a collection of 2,500 well-curated images matching the same scene from a street view matched to historical archival image data from Amsterdam city. The image pairs capture the same place with different cameras, viewpoints, and appearances. Unlike existing benchmark datasets, AmsterTime is directly crowdsourced in a GIS navigation platform (Mapillary). We evaluate various baselines, including non-learning, supervised and self-supervised methods, pre-trained on different relevant datasets, for both verification and retrieval tasks. Our result credits the best accuracy to the ResNet-101 model pre-trained on the *Landmarks* dataset for both verification and retrieval tasks by 84% and 24%, respectively. Additionally, a subset of Amsterdam landmarks is collected for feature evaluation in a classification task. Classification labels are further used to extract the visual explanations using Grad-CAM for inspection of the learned similar visuals in a deep metric learning models.**

I. Introduction

Visual place recognition (VPR) involves inferring a geographical location of a single image with broad applications in robotics, consumer photography, social media, and archival repositories. The question of "where was this photo taken?" is answered for a query image, by retrieving the most similar match from a geo-tagged gallery of images. Thus, VPR is conveniently formulated as a content-based image retrieval problem, where the image representation is key.

The ideal image representation for VPR maps all the images capturing the same place *close* to each other, regardless of various viewpoints, illuminations, appearances, and capturing sensors. At the same time, similar places are mapped *far enough* from all other images, in an n-dimensional latent space. [1], where the quality of the representation mapping is measured against precision and recall over all queries, commonly captured in mean average precision (mAP) [2]. In this conduct, ranking perfection is achieved once all images of the same place are ranked higher than all others in the gallery. These defined criteria for the VPR task, lend themselves to a image-similarity learning problem.

In the past decade, deep similarity learning became a dominant framework by using (dis)similar image pairs to train convolutional neural networks (CNN) such as Siamese [3] and Triplet models [4], [5]. At inference time, the trained CNN model is used as a feature extractor for the image retrieval task. Similar to other learned image descriptors (features), the underlying relations in the training data determine similar visual elements. For instance, similar (positive) image pairs with different illuminations in the training, potentially leading to an illumination-agnostic model.

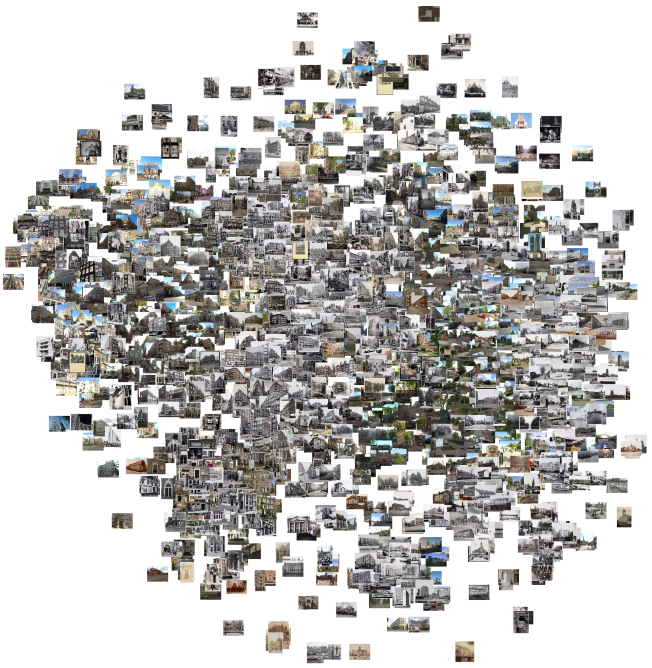

Fig. 1. The visualization of AmsterTime dataset using t-SNE

There exist numerous benchmark datasets for VPR, while none of them are directly crowd-sourced by retrieving similar visual places. A common approach to obtain positive images (most difficult) for VPR task is Geographic Information System based (GIS) annotations, e.g., from street view images with known GIS information [6]. GIS-based pairing, labels all the images with the same geographical altitude and latitude as similar, despite the fact that all the images taken from a single point do not share visual similarities, an example is orthogonal viewpoints. Others take images, taken by different people in social media or online photography platforms, from known landmarks such as Eiffel, Pyramids, etc [7], [8]. A problem with the latter is the undesired bias towards popular

geographical hotspots since many common architectural forms and typologies are excluded from the dataset. Other datasets use vehicle trajectories to capture the same scenes in different time frames ranging from seasonal to yearly intervals, tapering the scope of the VPR task to appearance-invariant learning and evaluation [9], [10]. All of these semi-automated pair mining methods, even though efficient for learning relevant visual features, are either unfaithful to visual similarity notion or relatively facile to trustfully benchmark the VPR task. In this paper, we introduce the first crowd-sourced benchmark dataset for the VPR task based on a visual search to match an archival query with street view images in the Mapillary navigation platform[1]. In turn, all the matching pairs are verified by a human expert to verify the correct matches and evaluate the human competence in the VPR task for further references. The properties of our dataset referred to as *AmsterTime*[2] are summarized as:

- 1200+ license-free images from the Amsterdam city Archive, representing urban places in the city of Amsterdam, captured in the past century by many photographers.
- All archival queries are matched with street view images from Mapillary.
- All matches are verified by architectural historians and Amsterdam inhabitants.
- Image pairs are archival and street views capturing the same place with different cameras, time lags, structural changes, occlusion, viewpoint, appearance, and illuminations.
- The dataset exhibits a domain shift between query and the gallery due to significant difference between scanned archival and street view images.

We embrace data scarcity as a realistic setting and we purposely limit AmsterTime dataset for evaluating the VPR baselines rather than training. We also add visual similarity learning baselines with the latest self-supervision frameworks and visual inspection with Grad-CAM [11] model to qualitatively evaluate the learned visual features. We list the contributions as:

1) Various baselines including recent self-supervised SimSiam [12] model is evaluated on AmsterTime dataset.
2) VGG-16 [13] and ResNet-50 [14] models are trained on a very large Google Landmarks dataset [8] for visual similarity learning with a self-supervised framework.
3) Relevant landmarks from Amsterdam city are collected into a new classification dataset, from Google Landmarks dataset, to evaluate the learned similarity features, using class activation mapping frameworks such as Grad-CAM [11].
4) Visual explanations are generated using Grad-CAM model to inspect the visual similarities learned in the self-supervised models.

AmsterTime covers lifelong temporal coverage of Amsterdam city with severe domain shift between query and gallery which is uniquely challenging to benchmark VPR models as the baseline results indicate. t-SNE visualization of all the images in the dataset is given in Fig. 1 and some example image pairs are also given in Fig. 2. The dataset and the evaluation code are available at the project repository.[3]

TABLE I
RECENT VPR DATASETS (LEFT) WITH THE CORRESPONDING DATA CAPTURING AND ANNOTATION MEDIUM (RIGHT). AMSTERTIME COMBINES TWO IMAGE DOMAINS TO REPRESENT THE SAME PLACE.

Datasets	Imagery
Oxford RobotCar [15]	Car Traverse
Berlin Kudamm [10]	Train Traverse
Mapillary SLS [16]	Mapillary Street View
Pittsburgh-30k [17]	Google Street View
Tokyo247 [18]	Google Street View
Nordland [9]	Train Traverse
Garden points [19]	Car Traverse
AmsterTime (ours)	Mapillary Street View + Archive

II. RELATED WORK

A. Datasets

There are valuable survey papers that may be consulted for broader discussion over developments of VPR models and applications [20], [21]. This work is based on [22] that introduces unsupervised domain adaption and attention mechanism to solve the domain shift between query and gallery images in VPR task. Unlike [22], focusing on learning from large unpaired image sets from two domains of archival and street views, we develop a benchmark dataset of cross-domain image pairs to reliably evaluate learned image representations.

Among popular VPR datasets (Tab. I), the Berlin Kudamm dataset [10], exhibit extreme viewpoint variation in the query and reference traverses. This dataset contains recurring and upfront dynamic objects which are uncommon to any other VPR dataset. Nordland dataset [9] sample images are one of the highly seasonally variant datasets and have manually introduced lateral viewpoint variation. Gardens Point dataset [19] images are presented here highly illumination variant and accompanied with lateral viewpoint variation. In contrast to these works, our dataset is unique in that it offers all the possible image variations including viewpoints, illuminations, appearances, and capturing sensors resulting in domain shift effect between the image pairs and thus extremely challenging.

B. Self-supervised representation learning

In addition to the dataset, we investigate the performance of self-supervised similarity learning models for solving the VPR task, which is relevant because training data is scarce. In practice, constructing tuple training data and hard negative mining for deep visual similarity learning turned into a scalability bottleneck for suitable training datasets. Recently, Chen et al [12] introduced a promising self-supervised framework based on Siamese networks that are trained only with positive image pairs, discarding altogether learning from dissimilar

[1] https://www.mapillary.com

[2] This project is partly funded by ArchiMediaL project.

[3] https://github.com/seyrankhademi/AmsterTime

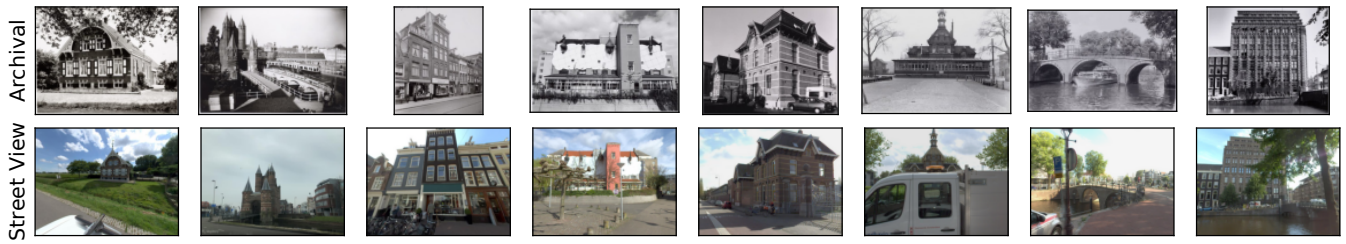

Fig. 2. Sample image pairs from AmsterTime dataset. Challenges are extreme occlusions, view point changes, camera lens distortions, color changes.

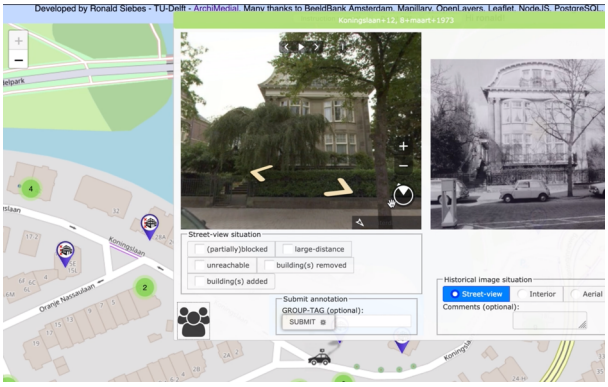

Fig. 3. Screenshot of the ArchiMediaL annotator app

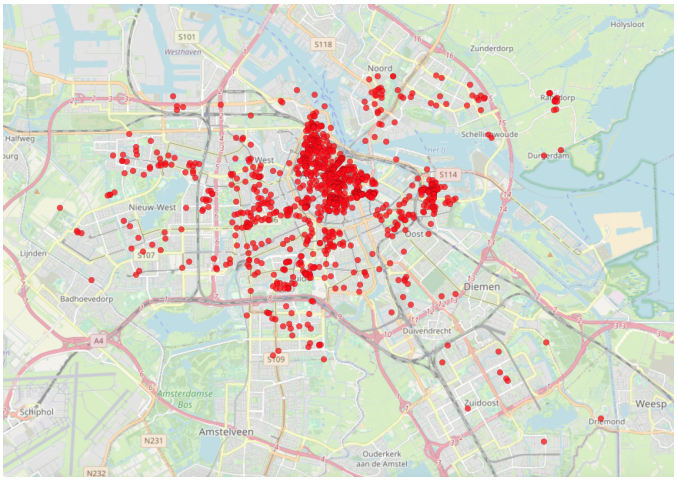

Fig. 4. The urban distribution of AmsterTime dataset shows the concentration of data at the center of Amsterdam following the archival imagery pattern.

pairs. We are inspired by [12], that significantly reduces the combinatorial complexity of contrastive learning.

In general, self-supervised learning is used for task-agnostic representation learning [23], [24], [25], [26], [27], [12] commonly by a contrastive learning framework and Siamese networks. Interestingly, competitive performance is reported in the literature for self-supervised learning compare to the supervised learning models [23], [24], [25], [26], [27], [12]. Among the self-supervised models, SimCLR [25] needs both negative and positive pairs with large batch sizes. While, SimSiam [12] has an extra predictor module on one branch of its network which provides asymmetry and it prevents collapsing even with relatively small batch sizes in absence of negative pairs. Moreover, Barlow Twins [27] aims redundancy reduction in the representations by using a loss function on the cross-correlation matrix of the embedding which also prevents trivial solutions without the need for asymmetry in the Siamese setting.

III. Crowdsourcing AmsterTime

A. Data Collection

Crowdsourcing is a popular way to gather training and evaluation data for deep learning models. Well-known crowdsourcing platforms such as Amazon Mechanical Turk [28] and AutoML [29], are well suited for general crowdsourcing, yet, they are not adequate for our data collection application due to the complexity of the implementation of GIS layers. Therefore, we developed a custom crowdsourcing web application (Fig. 3). This annotation tool shows a participant the combination of an archival image and a 3D street-view navigator from the Mappilary platform. The navigator is positioned close to the expected location where the archival image is being taken, according to the available metadata, allowing the user to expand or zoom and match the archival and contemporary image in a game-like fashion. The tool also provides an evaluation interface, where administrators can manually verify or deny submissions. This task takes only a fraction of the time per image in comparison to the annotation task itself as it is a binary classification task of acceptance or rejection [30].

The selection of archival images originates from a fairly well-documented area of architectural and urban history (Fig. 4), in the Beeldbank repository of the Amsterdam City Archives[4] – the world's largest city archive. Moreover, the annotators are familiar with the place, from which the data is collected.

B. Benchmark Tasks

AmsterTime includes 1231 matched archival and corresponding street view image pairs. We used these pairs to create

[4]Beeldbank Stadsarchief Amsterdam. The Beeldbank contains several hundred thousand images taken in the streets of Amsterdam since the nineteenth century, among them many images of facades, buildings, and streets. https://archief.amsterdam/beeldbank/.

TABLE II

RESULTS FOR VERIFICATION AND RETRIEVAL TASKS. THE BACKBONE ARCHITECTURES ARE GIVEN IN THE PARENTHESES. THE MODELS ARE TRAINED ON THE GIVEN DATASET AND EVALUATED ON AMSTERTIME DATASET. SIFT AND LIFT FEATURES ARE CONVERTED TO 128 DIMENSIONAL BOVW. THE FOLLOWING THREE CNN ARCHITECTURES ARE IMAGENET-PRETRAINED MODELS USED ONLY FOR EVALUATION. NETVLAD AND AP-GEM ARCHITECTURES ARE ALSO PRE-TRAINED MODELS AND USED ONLY FOR EVALUATION. EXCEPT THE FIRST SIMSIAM MODEL, WE TRAINED THE REST OF THEM WITH SELF-SUPERVISION WITH THE COMBINATION OF 2 BACKBONES AND 3 DATASETS. THE FIRST SIMSIAM MODEL IS THE PRE-TRAINED MODEL FROM THE ORIGINAL PAPER [12] AND USED ONLY FOR EVALUATION. BOLD NUMBERS DENOTE THE BEST SCORES FOR EACH COLUMN.

		Verification					Retrieval		
Method	**Train Dataset**	**Precision**	**Recall**	**F1**	**Acc**	**ROC AUC**	**mAP**	**Top1**	**Top5**
SIFT [31] w/ BoVW	N/A	0.57	0.65	0.61	0.58	0.61	0.03	0.01	0.04
LIFT [32] w/ BoVW	Piccadilly	0.56	0.60	0.58	0.57	0.59	0.03	0.01	0.04
VGG-16 [13]	ImageNet	0.75	0.63	0.68	0.71	0.78	0.18	0.13	0.23
ResNet-50 [14]	ImageNet	0.63	0.66	0.65	0.64	0.69	0.06	0.04	0.08
ResNet-101 [14]	ImageNet	0.63	0.67	0.65	0.64	0.69	0.05	0.03	0.07
NetVLAD (VGG-16) [6]	Pittsburgh250k	0.83	**0.80**	0.82	0.82	0.90	0.26	0.17	0.33
AP-GeM (ResNet-101) [2]	Landmarks	**0.88**	0.78	**0.83**	**0.84**	**0.92**	**0.35**	**0.24**	**0.48**
SimSiam (ResNet-50) [12]	ImageNet	0.75	0.76	0.75	0.75	0.83	0.19	0.12	0.26
SimSiam (ResNet-50)	GLDv2	0.80	0.79	0.80	0.80	0.86	0.23	0.15	0.32
SimSiam (ResNet-50)	AmsterTime	0.72	0.75	0.73	0.73	0.81	0.19	0.12	0.26
SimSiam (VGG-16)	ImageNet	0.63	0.72	0.67	0.65	0.71	0.10	0.06	0.14
SimSiam (VGG-16)	GLDv2	0.63	0.77	0.70	0.66	0.75	0.12	0.07	0.18
SimSiam (VGG-16)	AmsterTime	0.77	0.70	0.73	0.74	0.81	0.16	0.10	0.22

both the *verification* and *retrieval* tasks that are closely related.

Verification is a binary classification (auxiliary) task to detect a pair of archival and street view images of the same place. The verification task for AmsterTime dataset has all of the crowdsourced image pairs as positive labeled, where the same number of negative samples are generated by randomly pairing archival and street view images summing up to a total of 2,462 pairs in the verification task.

Retrieval is the main task corresponding to VPR, in which a given query image is matched with a set of gallery images. For the retrieval task AmsterTime dataset offers 1231 query images where the leave-one-out set serves as the gallery images for each query.

IV. BASELINE EXPERIMENTS AND RESULTS

A. Experimental Setup

We take the pairwise distance between two high-dimensional feature vectors corresponding to all the images in AmsterTime. The average of all distance values generated by pairwise comparisons in the dataset is used as a threshold distance to classify positive (similar) or negative (dissimilar) pairs. None of the baseline models that we trained uses the dataset labels. Self-supervised models only use AmsterTime images but does not use the pairing annotations.

We calculate mean average precision mAP, $Top1$ and $Top5$ accuracy metrics for retrieval task using $cosine$ distance. For a given query archival image, we first sort all street view images by the distance between the archival image and the street view images in ascending order then the metrics are calculated.

B. Local Image Features

We investigated how local image features perform on AmsterTime dataset. The SIFT [31] descriptors are used to extract local features and they were then aggregated into one global feature per image using bag-of-visual-words encoding (BoVW) [33]. The process repeated for the descriptors extracted with the LIFT [32] trained on *Piccadilly* dataset [34]. The bag size is chosen 128 which performs best among others. The results for the verification and retrieval tasks using BoVW are given in Tab. II. Accuracy for verification task for SIFT is 58% (8% above the random baseline) and for LIFT 57%.

C. Off-the-shelf (pre-trained) CNN models

We evaluated the performance of image features extracted from commonly used CNN models pre-trained on different datasets and tasks including image classification and visual place recognition (VPR), on AmsterTime dataset.

The models VGG-16 [13], ResNet-50 [14], and ResNet-101 [6] are pre-trained on ImageNet [35] for image classification and used directly from PyTorch's library. The CNN models are only used for extracting features of the images in AmsterTime dataset. The features are obtained from the last convolutional layer followed by a ReLU and a max-pooling layers for VGG-16 model and from adaptive average pooling layer for ResNet models. The features are then utilized to calculate scores for verification and retrieval tasks. The results are given in Tab. II. One noticeable point is that VGG-16 works better than both ResNet-50 and ResNet-101 on verification task. That margin is much bigger on retrieval task as VGG-16 has 13% top-1 accuracy while ResNet-50 has 4% and ResNet-101 has 3% top-1 accuracy.

In the next step, we used NetVLAD [6] pre-trained on *Pittsburgh250k* [18] for VPR task as a close match to our task of image retrieval. In addition, we evaluated AP-GeM [2] pre-trained on *Landmaks-clean* dataset [7] on AmsterTime dataset. The NetVLAD model has a VGG-16 backbone while AP-GeM has ResNet-101 backbone. Neither of them are trained further than the publicly available model weights. As usual, the

TABLE III
RESULTS OF SUPERVISED TRAINED RESNET-18 WITH TRIPLET LOSS [5] ON AMSTERTIME DATASET. THE NUMBERS ARE MERELY TO QUANTIFY THE SUPERVISION GAP COMPARED TO UNSUPERVISED MODELS IN TAB. II

Verification				Retrieval		
Precision	**Recall**	**Acc**	**ROC AUC**	**mAP**	**Top1**	**Top5**
0.85	0.89	0.87	0.93	0.42	0.30	0.53

models are used to extract image features for both verification and retrieval task. AP-GeM and NetVLAD models result in 84% and 82% accuracies on verification task, respectively. The pre-trained AP-GeM achieves the best performance among all the baselines as it leverages the largest training dataset which is very similar to images in AmsterTime dataset.

D. Self-supervised Baseline

Due to the limited size of AmsterTime, self-supervision is a suitable option to exploit data without labels. SimSiam [12] is a recent method that combines self-supervision and similarity learning without needing for neither negative samples nor large batches. We evaluated six SimSiam models with different data and architectures presented in the last row section in Tab. II. The list starts from ResNet-50 model trained on ImageNet[5]. We trained two more ResNet-50 models on Google Landmarks (GLDv2) and AmsterTime datasets with same settings except that the batch size is 128 in our trainings. Since ImageNet and GLDv2 are relatively large datasets and AmsterTime is limited dataset, the model trained on GLDv2 is trained for 100 epochs and the model trained on AmsterTime dataset is trained for 10000 epochs to equalize the number of used mini-batches during training. Moreover, three VGG-16[6] models are trained on the same datasets (ImageNet, GLDv2 and AmsterTime) with the same settings as bare supervised VGG-16 has better results than ResNet-50 remarked in Sec. IV-C. The models started the self-supervised training from scratch (with random parameters) and after the self-supervised training are completed, the trained models used as usual to extract features from the images in AmsterTime dataset. The results are presented in Tab. II. Contrary to the better results for pre-trained VGG-16 model mentioned in Sec. IV-C, ResNet-50 outperformed in self-supervised learning.

E. Supervised Baseline

To have a real supervised baseline and measure the supervision gap, we also trained a ResNet-18 model in Triplet setting [5] with grand-truth pair labels. The dataset first divided into train and test with the ratio of $4:1$. Besides ground-truth (positive) pairs, equal number of pseudo negative pairs are randomly generated by pairing the archival and street-view images. The model trained for 90 epochs with SGD optimizer with 128 of batch size, 0.001 learning rate (decayed by 0.1 after each 30 epochs), 0.9 momentum and 0.00001 weight decay. The model is then used to extract feature on the test set. The results for verification and retrieval are given in Tab. III. Due to the limited size of AmsterTime dataset, the supervision gap is only around 7% in *mAP*.

[5]This model is used from SimSiam authors' shared models.

[6]VGG-16 model architecture is with batch normalization which is used directly from PyTorch's library.

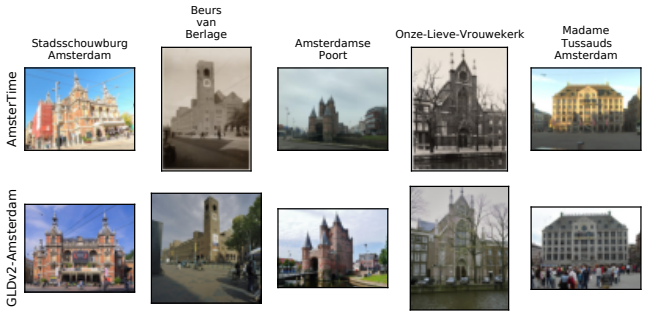

Fig. 5. Example of similar images for the same landmark classes in both AmsterTime and GLDv2-Amsterdam datasets enables visualizing learned features with Grad-CAM.

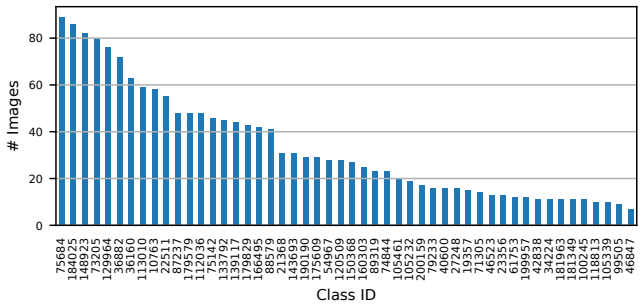

Fig. 6. Histogram of the number of images per selected 50 landmarks located in Amsterdam from Google Landmarks Dataset v2 shows that class distribution is unbalanced and solved simply by duplicating images in under-represented classes.

V. VISUAL EXPLANATIONS

We investigate the learned representation of the SimSiam [12] on AmsterTime dataset using Grad-CAM [11]. Grad-CAM requires a classification layer at the end of CNN architecture while SimSiam-trained models trained on similarity learning. To adapt Grad-CAM, (1) we add a randomly initialized liner classifier at the end of the SimSiam-trained models, (2) train the newly added classifier on a curated similar dataset with class labels (landmarks). The parameters of SimSiam-trained models are frozen after training on AmsterTime dataset.

A. Dataset for visualization

To facilitate visualization we curated a subset of Google Landmarks dataset v2 (GLDv2) [8] because it is semantically close to AmsterTime. Particularly, 50 landmarks are selected in GLDv2 which are located in the city of Amsterdam. Some of the hand-picked similar images have been given in Fig. 5. We will refer to this subset for the classification

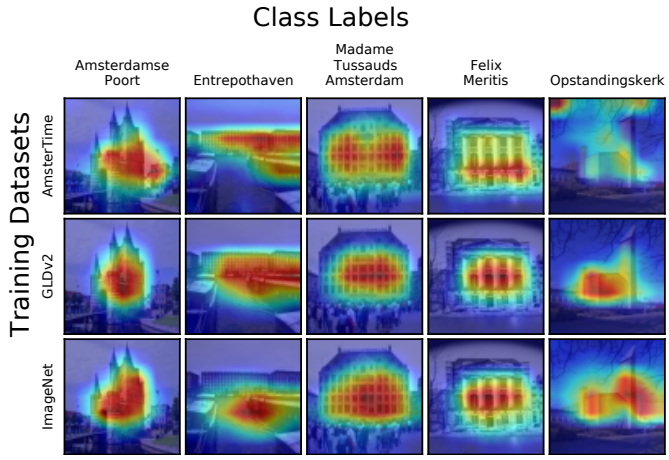

Fig. 7. Grad-CAM visualizations of three ResNet-50 models pre-trained with SimSiam [12] on three different datasets given in the labels on the left. Top labels denotes both class activation for Grad-CAM and grand-truth class labels. The visualizations suggest that models learn the structure in the images.

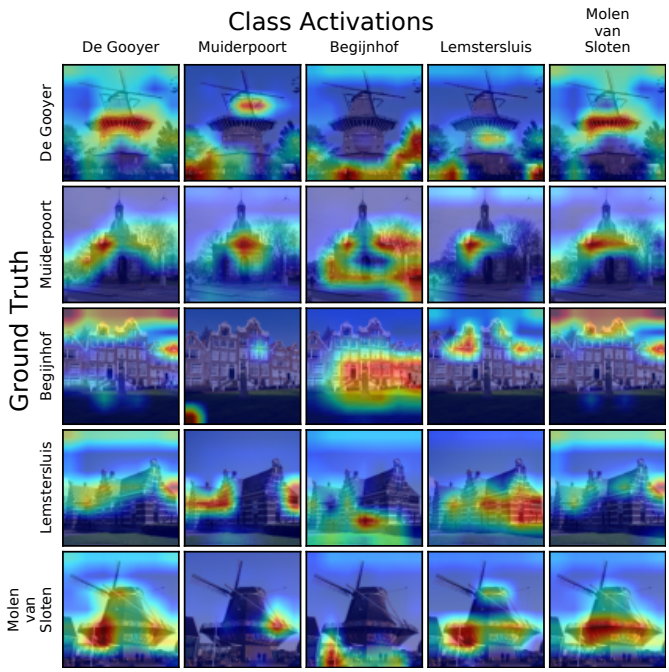

Fig. 8. Grad-CAM visualizations of ResNet-50 model pre-trained with SimSiam [12] on AmsterTime dataset. The visualizations on the diagonal and the intersection of De Gooyer and Molen van Sloten show that the model activates more when the activation class images are similar to the input images which indicates the model learned the structures in the images.

dataset as *GLDv2-Amsterdam* hereafter. The histogram of class distribution of GLDv2-Amsterdam, presented in Fig. 6, shows a highly skewed and imbalanced distribution. To alleviate the training suffering from the imbalanced dataset, of the linear classifier, underrepresented classes were simply duplicated GLDv2-Amsterdam.

B. Training the Linear Classifier

The linear classifier is trained based on [12]. A randomly-initialized linear classifier added to a frozen model is trained for 90 *epochs* with *batch size*= 256 using SGD with the parameters of cosine-decay-scheduled, initial *lr*= 30.0 , *weight decay*= 0, and *momentum*= 0.9.

C. Grad-CAM visualization

We created two visualizations: The first is to show how the same models trained on different datasets learn different visual features, and the second one is to see how a model reacts to different class activations of Grad-CAM.

Firstly, we selected a subset among the intersection of correctly classified images in GLDv2-Amsterdam dataset by three ResNet-50 models pre-trained on ImageNet, GLDv2, and AmsterTime with SimSiam self-supervision. The output of the last convolutional layer of the ResNet-50 models is visualized. The Grad-CAM visualizations for each of the selected images and for each of the models are created and imposed on the original images. In this setting, the grand-truth classes of the images are used as class activation to show the reactions of the models to the images w.r.t. grand-truth class activation. The visualizations are given in Fig. 7. The models give more weight to the landmark objects in the images, indicating that the models correctly focus on landmark features.

To illustrate the reactions of the models in comparison with other class activations besides the grand-truths, we also created a Grad-CAM visualization matrix given in Fig. 8. For this visualization matrix, we used the ResNet-50 model pre-trained on AmsterTime dataset with SimSiam self-supervision. The visualizations for images w.r.t. grand-truth class activation appears on the diagonal. This matrix shows that the model relies on the landmark object once the class activation is either the grad-truth class or the class of the images with similar landmarks such as *De Gooyer* and *Molen van Sloten*.

VI. Conclusions

We introduced AmsterTime a reliable and challenging evaluation dataset with verification and retrieval benchmark tasks for visual place recognition. AmsterTime dataset consists of $\sim$ 2500 archival and street view images matched by human annotators. Various image representation baselines including the local features, supervised and self-supervised models are tested on AmsterTime. The results suggest that supervised model trained on a large and similar dataset of *Landmarks* outperforms the self-supervised models. Oblation studies are carried out using visual explanations to investigate the learned features confirming the quality of AmsterTime dataset in learning relevant features despite its small size using self-supervised models. The code for this paper including the image features is available on a GitHub repository.

VII. Acknowledgement

This work is partially supported by Volkswagen Foundation under ArchiMediaL project. We show our gratitude to all the people who helped us to annotate data including Tino Mager, Beate Löffler, Carola Hein, and Victor de Boer.

REFERENCES

[1] R. Serajeh, S. Khademi, A. Mousavinia, and J. C. Van Gemert, "On sensitive minima in margin-based deep distance learning," *IEEE Access*, vol. 8, pp. 145 067–145 076, 2020. 1

[2] J. Revaud, J. Almazán, R. S. Rezende, and C. R. d. Souza, "Learning with average precision: Training image retrieval with a listwise loss," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 5107–5116. 1, 4

[3] G. Koch, R. Zemel, R. Salakhutdinov *et al.*, "Siamese neural networks for one-shot image recognition," in *ICML deep learning workshop*, vol. 2. Lille, 2015. 1

[4] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *SIMBAD*, 2015. 1

[5] V. Balntas, E. Riba, D. Ponsa, and K. Mikolajczyk, "Learning local feature descriptors with triplets and shallow convolutional neural networks." in *Bmvc*, vol. 1, 2016, p. 3. 1, 5

[6] 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. 1, 4

[7] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "Deep image retrieval: Learning global representations for image search," in *European conference on computer vision*. Springer, 2016, pp. 241–257. 1, 4

[8] T. Weyand, A. Araujo, B. Cao, and J. Sim, "Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2575–2584. 1, 2, 5

[9] D. Olid, J. M. Fácil, and J. Civera, "Single-view place recognition under seasonal changes," in *PPNIV Workshop at IROS 2018*, 2018. 2

[10] Z. Chen, F. Maffra, I. Sa, and M. Chli, "Only look once, mining distinctive landmarks from convnet for visual place recognition," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 9–16. 2

[11] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626. 2, 5

[12] X. Chen and K. He, "Exploring simple siamese representation learning," *arXiv preprint arXiv:2011.10566*, 2020. 2, 3, 4, 5, 6

[13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014. 2, 4

[14] 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. 2, 4

[15] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 Year, 1000km: The Oxford RobotCar Dataset," *The International Journal of Robotics Research (IJRR)*, vol. 36, no. 1, pp. 3–15, 2017. [Online]. Available: http://dx.doi.org/10.1177/0278364916679498 2

[16] 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. 2

[17] A. Torii, R. Arandjelović, J. Sivic, M. Okutomi, and T. Pajdla, "24/7 place recognition by view synthesis," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1808–1817. 2

[18] A. Torii, J. Sivic, T. Pajdla, and M. Okutomi, "Visual place recognition with repetitive structures," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 883–890. 2, 4

[19] N. Sünderhauf, F. Dayoub, S. Shirazi, B. Upcroft, and M. Milford, "On the performance of convnet features for place recognition," *CoRR*, vol. abs/1501.04158, 2015. [Online]. Available: http://arxiv.org/abs/1501.04158 2

[20] C. Masone and B. Caputo, "A survey on deep visual place recognition," *IEEE Access*, vol. 9, pp. 19 516–19 547, 2021. 2

[21] X. Zhang, L. Wang, and Y. Su, "Visual place recognition: A survey from deep learning perspective," *Pattern Recognition*, vol. 113, 11 2020. 2

[22] Z. Wang, J. Li, S. Khademi, and J. van Gemert, "Attention-aware age-agnostic visual place recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, Oct 2019. 2

[23] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *arXiv preprint arXiv:2006.09882*, 2020. 3

[24] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9729–9738. 3

[25] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. Hinton, "Big self-supervised models are strong semi-supervised learners," *arXiv preprint arXiv:2006.10029*, 2020. 3

[26] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar *et al.*, "Bootstrap your own latent: A new approach to self-supervised learning," *arXiv preprint arXiv:2006.07733*, 2020. 3

[27] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow twins: Self-supervised learning via redundancy reduction," *arXiv preprint arXiv:2103.03230*, 2021. 3

[28] K. Crowston, "Amazon mechanical turk: A research tool for organizations and information systems scholars," in *Shaping the future of ict research. methods and approaches*. Springer, 2012, pp. 210–221. 3

[29] "AutoML Vision ," https://cloud.google.com/vision/automl/docs. 3

[30] S. Khademi, T. Mager, and R. Siebes, "Deep learning from history," in *Research and Education in Urban History in the Age of Digital Libraries*, F. Niebling, S. Münster, and H. Messemer, Eds. Cham: Springer International Publishing, 2021, pp. 213–233. 3

[31] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004. 4

[32] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "Lift: Learned invariant feature transform," in *European conference on computer vision*. Springer, 2016, pp. 467–483. 4

[33] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Computer Vision, IEEE International Conference on*, vol. 3. IEEE Computer Society, 2003, pp. 1470–1470. 4

[34] K. Wilson and N. Snavely, "Robust global translations with 1dsfm," in *European conference on computer vision*. Springer, 2014, pp. 61–75. 4

[35] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255. 4