Patch-DrosoNet: Classifying Image Partitions With Fly-Inspired Models For Lightweight Visual Place Recognition

Bruno Arcanjo[1], Bruno Ferrarini[1], Michael Milford[2], Klaus D. McDonald-Maier[1] and Shoaib Ehsan[1,3]

Abstract— Visual place recognition (VPR) enables autonomous systems to localize themselves within an environment using image information. While Convolutional Neural Networks (CNNs) currently dominate state-of-the-art VPR performance, their high computational requirements make them unsuitable for platforms with budget or size constraints. This has spurred the development of lightweight algorithms, such as DrosoNet, which employs a voting system based on multiple bio-inspired units. In this paper, we present a novel training approach for DrosoNet, wherein separate models are trained on distinct regions of a reference image, allowing them to specialize in the visual features of that specific section. Additionally, we introduce a convolutional-like prediction method, in which each DrosoNet unit generates a set of place predictions for each portion of the query image. These predictions are then combined using the previously introduced voting system. Our approach significantly improves upon the VPR performance of previous work while maintaining an extremely compact and lightweight algorithm, making it suitable for resource-constrained platforms.

I. INTRODUCTION & BACKGROUND

Visual place recognition (VPR) is an essential component of mobile robotics, as it allows the system to localize itself in the runtime environment using only image data [1]. The affordability and variety of camera sensors makes VPR localization particularly attractive for hardware restricted robotic platforms, which are common in mobile robotics [2], [3]. Nevertheless, VPR is a complicated task and proposed solutions must deal with several visual challenges. The same place can appear vastly different when visited under different illumination [4], seasonal weather conditions [5], viewpoints [6] and elements entering or leaving the scene [7]. As previously mentioned, mobile robotic platforms often operate with low-end hardware, making computational cost yet another important consideration when designing VPR techniques.

The importance of VPR and its variety of challenges has resulted in a growing number of approaches being proposed in the literature [8]. Some of the first successful solutions [9] relied on handcrafted local feature descriptors, such as Scale-Invariant Feature Transform (SIFT) [10] and Speeded-up Robust Features (SURF) [11], to build a viewpoint-robust map of the environment. Despite their strong resilience to viewpoint changes, local feature based approaches are susceptible to strong appearance changes. Whole-image descriptors, like Histogram Oriented Gradients (HOG) [12] have also been employed in VPR [13]. Recently, state-of-the-art VPR performance has been achieved by using Convolutional Neural Network (CNN) based approaches [14], as features from the inner layers of trained CNNs have been shown to significantly improve VPR performance [15]. Several CNN-based techniques [16], [17], [18] have thus been successfully employed for performing VPR.

The impressive VPR performance offered by CNN-based approaches comes with the significant downside of a high-computational cost, often demanding powerful graphical processing units (GPUs) to be ran in real time [3]. This shortcoming makes these top-performing techniques unusable for hardware-restricted platforms and several lightweight VPR algorithms have hence emerged to address it. HOG has been shown to be a fast VPR descriptor if used with suitable image sizes. CoHOG [19] makes use of the efficient HOG to encode high-entropy regions of an image, improving resilience to viewpoint changes but significantly increasing place matching computation times. [20] is also a region-based method, adapting the high-performing and costly NetVLAD [17] descriptor. CALC [21] presents itself as a train-free, lightweight CNN model, capable of competitive real-time VPR performance. Bio-inspired VPR approaches attempt to replicate the efficient localization abilities of small animals, resulting in algorithms such as [22], [23]. Recently, a lightweight VPR voting system [24] based on multiple units, each dubbed DrosoNet, inspired by the odour processing abilities of the fruit fly [25] has been proposed. The system capitalizes on the inherit randomness of the initialization and training process of individual units, where different models might specialize better or worse on different visual features. The combination of multiple units via the voting mechanism attempts to eliminate the weak spots of some units with the strengths of others. However, since each DrosoNet is trained on the entirety of each reference image, the sources of feature specialization are minimal.

In this work, we propose a novel, region-based approach to train individual DrosoNets coupled with a convolution-like prediction mechanism. By training different DrosoNets on different sections of the reference images, we introduce another source of variation between different units, rather than just relying on random initialization and training. Fur-

[1]B. Arcanjo, B. Ferrarini, K. D. McDonald-Maier and S. Ehsan are with the School of Computer Science and Electronic Engineering, University of Essex, United Kingdom (email: bq17319@essex.ac.uk; bferra@essex.ac.uk; kdm@essex.ac.uk; sehsan@essex.ac.uk)

[2]M. Milford is with the School of Electrical Engineering and Computer Science, Queensland University of Technology, Brisbane, QLD 4000, Australia (email: michael.milford@qut.edu.au)

[3]S. Ehsan is also with the school of Electronics and Computer Science, University of Southampton, United Kingdom (email: s.ehsan@soton.ac.uk)

thermore, the matching mechanism makes each unit place a prediction for each region in the query image, providing more information during voting. Our approach significantly improves VPR performance when compared to previous work, while retaining the lightweight capabilities of the technique.

II. METHOD

In this section, we provide implementation details of our proposed Patch-DrosoNet algorithm. Firstly, a quick overview of the basic functionality of DrosoNet is given. Then, we explain the processes of splitting the images, training the DrosoNets on different image patches and place-matching at runtime.

A. DrosoNet Usage

As explained in [24], DrosoNet is a compact and fast neural network image classifier inspired by Drosophila Melanogaster, where each of the total N places is a different class. DrosoNet works as a classification function:

$$D(i) = S_i \tag{1}$$

where i is the 32×64 input image and S_i is the output score vector of N elements, where each score corresponds to one of the reference places. The class obtaining the highest score in S is output as the place prediction.

While DrosoNet is a fast algorithm, its standalone VPR performance is lower than more computationally intensive techniques. Moreover, due to the randomness of its initialization and training, different DrosoNets exhibit high variance in their performance with different visual conditions. Combining multiple DrosoNets was hence proposed as a measure to improve overall VPR performance, relying only the native stochastic behaviour of the models for differentiation [24].

Our novel approach is to train multiple groups of DrosoNets on different patches of the same image, specializing each group on the features of each image region.

B. Image Splitting

The image splitting takes an input image i and returns a grid of patches of dimensions $r \times c$, where r is the number of rows and c the number of columns in the grid. Since DrosoNet operates with 32×64 images, we first resize the image to $32r \times 64c$. Within the same dataset, all images are split in the same fashion, and a group of DrosoNets is assigned to each patch.

C. Training

One DrosoNet group is assigned to each patch of the split reference images and it is trained only on those particular regions. The goal is for each DrosoNet group to specialize on the visual features of a region of the reference images. The total number of DrosoNets T in the algorithm is hence riven by

$$T = rcz \tag{2}$$

where z is an hyperparameter denoting the number of DrosoNets to use per patch.
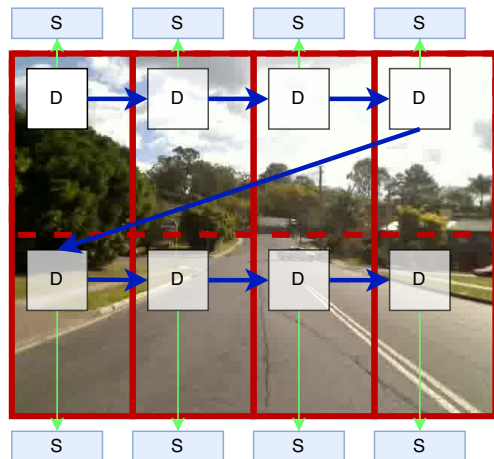
D. Matching

Fig. 1: Matching process on 2×4 grid.

At matching time, each DrosoNet evaluates all sections of the query image, regardless of what patch it was trained on. This process generates rc score vectors per query image per DrosoNet D. Fig. 1 provides an example of the matching process for a split grid of 2×4 regions, yielding a total of 8 score vectors per DrosoNet.

The total number of DrosoNet function calls per query image, C, and consequently also the total number of score vectors, can be computed by

$$C = Trc \tag{3}$$

Note how the number of regions factor rc is squared, and it is therefore important to use a low number of patches to preserve computational efficiency. Finally, all the C score vectors produced by the T different DrosoNets are merged using the same voting system as in [24], producing a final score vector from which the final prediction can be extracted.

The exhaustive approach to matching is required, as the visual features of a reference region might be partially present in a different patch of the query due to viewpoint changes.

III. EXPERIMENTS

In this section, we provide details on our experimental setup, such as model configurations and datasets. We then present and discuss our results in terms of VPR performance and computational efficiency.

A. Setup

We use the datasets in Table I, allowing for a margin of error of 1 frame for Nordland Winter and Fall [26] and of 2 frames for Day-Right and St. Lucia [27].

We compare our proposed approach against other lightweight techniques such as CoHOG, CALC and the established Voting system. For the first two techniques, we use the implementations provided in [28], while for Voting we use the settings in [24]. For Patch-DrosoNet, we use the

TABLE I: Dataset Details

Dataset	Condition	Image-Grid	DrosoNets Per Patch	Number of Images
Nordland Winter	Extreme seasonal	3x1	16	1000
Nordland Fall	Moderate seasonal	3x1	4	1000
Day-Right	Outdoor Lateral Shift	1x3	8	200
St. Lucia	Daylight; Dynamic Elements	4x2	8	1100

same DrosoNet configuration as in [24], with image-grids and DrosoNets per patch as detailed in Table I.

B. Results

Assessing VPR performance, we show how the different lightweight techniques compare in the precision-recall curves in Fig. 2, alongside the respective area under these curves (AUC). In the challenging appearance changes of Nordland Winter, our proposed Patch-DrosoNet is by far the best performing method, with more than double the AUC of all other techniques. In the more moderate appearance changes presented in Fall, our method is tied with the previously established Voting system, both being the top performers. In the viewpoint lateral shift assessment of Day-Right, Patch-DrosoNet is once again tied with the legacy Voting, with CoHOG achieving the best performance. Finally, our proposed approach is again the top performer in the St. Lucia dataset, displaying improved resilience against illumination changes and dynamic elements.

In Fig. 3 we show the prediction times of the different techniques on a dataset with 1000 reference images. Patch-DrosoNet is the second fastest algorithm, even in the most expensive configuration tested ($C = 512$ for the St. Lucia dataset). Only Voting remains faster due to a lower number of DrosoNet runs, but also present overall worse VPR performance, especially in the Winter dataset. Even with $C = 2000$, Patch-DrosoNet remains faster than CALC and CoHOG, leaving open the possibility for more complex DrosoNet grouping schemes.

IV. CONCLUSIONS AND FUTURE WORK

In this work, we propose a novel approach to train and utilize the established DrosoNet algorithm, resulting in a lightweight VPR technique with advantages over previous work. The core of the method consists on dividing images into patches and training different DrosoNet groups to specialize on different image regions, increasing differentiation between DrosoNets. At match time, each DrosoNet outputs its scores for each region of the query image and all score vectors are then combined using the voting mechanism.

However, the system is not without limitations. The optimal image grid and number of DrosoNets varies across different datasets, making it harder to produce a more general solution. For improving this work, we suggest focusing on how to identify key regions within an image, or using a segmentation algorithm to extract different classes of patches that can then be used for specialization.

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, 2015.

[2] F. Maffra, Z. Chen, and M. Chli, "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.

[3] B. Ferrarini, M. Waheed, S. Waheed, S. Ehsan, M. Milford, and K. D. McDonald-Maier, "Visual place recognition for aerial robotics: Exploring accuracy-computation trade-off for local image descriptors," in *2019 NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*. IEEE, 2019, pp. 103–108.

[4] A. Ranganathan, S. Matsumoto, and D. Ilstrup, "Towards illumination invariance for visual localization," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 3791–3798.

[5] T. Naseer, L. Spinello, W. Burgard, and C. Stachniss, "Robust visual robot localization across seasons using network flows," in *Twenty-eighth AAAI conference on artificial intelligence*, 2014.

[6] A. Pronobis, B. Caputo, P. Jensfelt, and H. I. Christensen, "A discriminative approach to robust visual place recognition," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 3829–3836.

[7] C.-C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte, "Simultaneous localization, mapping and moving object tracking," *The International Journal of Robotics Research*, vol. 26, no. 9, pp. 889–916, 2007.

[8] C. Masone and B. Caputo, "A Survey on Deep Visual Place Recognition," *IEEE Access*, vol. 9, pp. 19 516–19 547, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9336674/

[9] M. Cummins and P. Newman, "Appearance-only SLAM at large scale with FAB-MAP 2.0," *The International Journal of Robotics Research*, vol. 30, no. 9, pp. 1100–1123, 2011.

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

[11] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1. Ieee, 2005, pp. 886–893.

[13] C. McManus, B. Upcroft, and P. Newmann, "Scene signatures: Localised and point-less features for localisation," in *Proceedings of Robotics: Science and Systems*, Berkeley, USA, July 2014.

[14] Y. Hou, H. Zhang, and S. Zhou, "Convolutional neural network-based image representation for visual loop closure detection," in *2015 IEEE International Conference on Information and Automation*, 2015, pp. 2238–2245.

[15] M. Zaffar, A. Khaliq, S. Ehsan, M. Milford, and K. McDonald-Maier, "Levelling the playing field: A comprehensive comparison of visual place recognition approaches under changing conditions," *arXiv preprint arXiv:1903.09107*, 2019.

[16] Z. Chen, A. Jacobson, N. Sünderhauf, B. Upcroft, L. Liu, C. Shen, I. Reid, and M. Milford, "Deep learning features at scale for visual place recognition," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 3223–3230.

[17] 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.

[18] Y. Hou, H. Zhang, and S. Zhou, "Convolutional neural network-based image representation for visual loop closure detection," in *2015 IEEE international conference on information and automation*. IEEE, 2015, pp. 2238–2245.
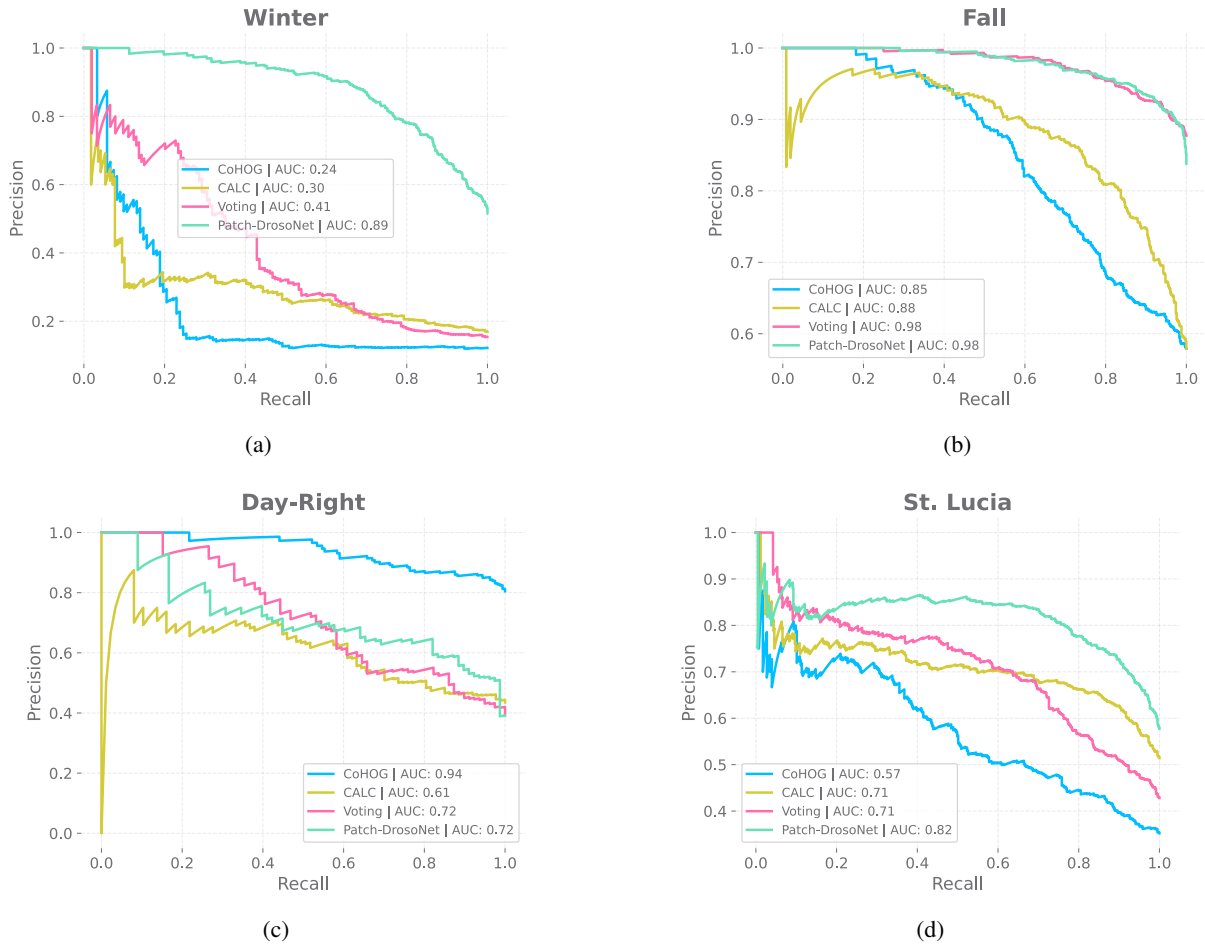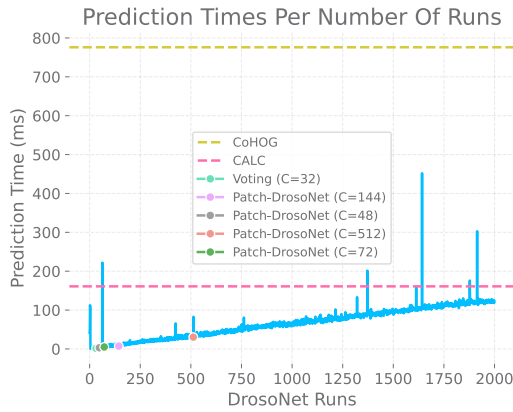
Fig. 2: Precision-Recall Curves

Fig. 3: Prediction Computation Time Comparison.

[19] M. Zaffar, S. Ehsan, M. Milford, and K. McDonald-Maier, "Co-HOG: A light-weight, compute-efficient, and training-free visual place recognition technique for changing environments," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1835–1842, 2020.

[20] S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer, "Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 14 136–14 147.

[21] N. Merrill and G. Huang, "Lightweight unsupervised deep loop closure," *arXiv preprint arXiv:1805.07703*, 2018.

[22] M. J. Milford, G. F. Wyeth, and D. Prasser, "RatSLAM: a hippocampal model for simultaneous localization and mapping," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, vol. 1.   IEEE, 2004, pp. 403–408.

[23] M. Chancán, L. Hernandez-Nunez, A. Narendra, A. B. Barron, and M. Milford, "A hybrid compact neural architecture for visual place recognition," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 993–1000, 2020.

[24] B. Arcanjo, B. Ferrarini, M. Milford, K. D. McDonald-Maier, and S. Ehsan, "An efficient and scalable collection of fly-inspired voting units for visual place recognition in changing environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2527–2534, 2022.

[25] T. A. Ofstad, C. S. Zuker, and M. B. Reiser, "Visual place learning in drosophila melanogaster," *Nature*, vol. 474, no. 7350, pp. 204–207, 2011.

[26] N. Sünderhauf, P. Neubert, and P. Protzel, "Are we there yet? challenging SeqSLAM on a 3000 km journey across all four seasons," in *Proc. of Workshop on Long-Term Autonomy, IEEE International Conference on Robotics and Automation (ICRA)*.   Citeseer, 2013, p. 2013.

[27] A. J. Glover, W. P. Maddern, M. J. Milford, and G. F. Wyeth, "Fab-map+ ratslam: Appearance-based slam for multiple times of day," in *2010 IEEE international conference on robotics and automation*.   IEEE, 2010, pp. 3507–3512.

[28] M. Zaffar, S. Garg, M. Milford, J. Kooij, D. Flynn, K. McDonald-Maier, and S. Ehsan, "Vpr-bench: An open-source visual place recognition evaluation framework with quantifiable viewpoint and appearance change," *International Journal of Computer Vision*, pp. 1–39, 2021.