A Hybrid Compact Neural Architecture for Visual Place Recognition

Marvin Chancán[1,3], Luis Hernandez-Nunez[2,3], Ajay Narendra[4], Andrew B. Barron[4], and Michael Milford[1]

***Abstract*—State-of-the-art algorithms for visual place recognition, and related visual navigation systems, can be broadly split into two categories: computer-science-oriented models including deep learning or image retrieval based techniques with minimal biological plausibility, and neuroscience-oriented dynamical networks that model temporal properties found in neural cells underlying spatial navigation in the brain. In this paper, we propose a new compact and high-performing place recognition hybrid model that bridges this divide for the first time. Our approach comprises two key components that incorporate neural models of these two categories: (1) FlyNet, a compact, sparse two-layer neural network inspired by brain architectures of fruit flies, *Drosophila melanogaster*, and (2) a one-dimensional continuous attractor neural network (CANN). The resulting FlyNet+CANN network combines the compact pattern recognition capabilities of our FlyNet model with the powerful temporal filtering capabilities of an equally compact CANN, replicating entirely in a hybrid neural implementation the functionality that yields high performance in algorithmic localization approaches like SeqSLAM. We evaluate our approach, and compare it to three state-of-the-art place recognition methods, on two benchmark real-world datasets with small viewpoint variations and extreme environmental changes; including day/night cycles where it achieves an AUC performance of 87% compared to 60% for Multi-Process Fusion, 46% for LoST-X and 1% for SeqSLAM, while being 6.5, 310, and 1.5 times faster respectively.**

***Index Terms*—Biomimetics, Localization, Visual-Based Navigation**

I. Introduction

PERFORMING visual place recognition (VPR) reliably is a challenge for any robotic system or autonomous vehicle operating over long time periods in real-world environments; mainly due to the wide variety of viewpoint changes, perceptual aliasing (multiple places may look similar), and visual appearance variations over time (e.g. day/night or weather/seasonal cycles) [1]. Convolutional neural networks (CNN), heavily used in a range of computer vision tasks [2], have been applied to the field of VPR with great success over the past five years [3], [4]; typically only used in real-time with dedicated hardware (GPU) [5]–[7]. However, as vanilla CNN models, trained on benckmark datasets such as ImageNet [8] or Places365 [9], generally neglect any temporal information between images. Conversely, sequence-based algorithms such as SeqSLAM [10] are often applied on top of those models to achieve state-of-the-art results on VPR tasks required to match two or more sequences of images.

Related research in visual navigation have recently used computer-science-oriented recurrent neural networks (RNN) [11] in an attempt to model the multi-scale spatial representation network dynamics found in the entorhinal cortex of mammalian brains [12], [13]. While the results are promising, these systems are tested only in small synthetic environments, and the integration of neuroscience-oriented recurrent models such as continuous attractor neural networks (CANN) [14], [15] is not well explored. Only recently, analytic theories to unify both types of recurrent networks, trained on navigational tasks, have been proposed [16].

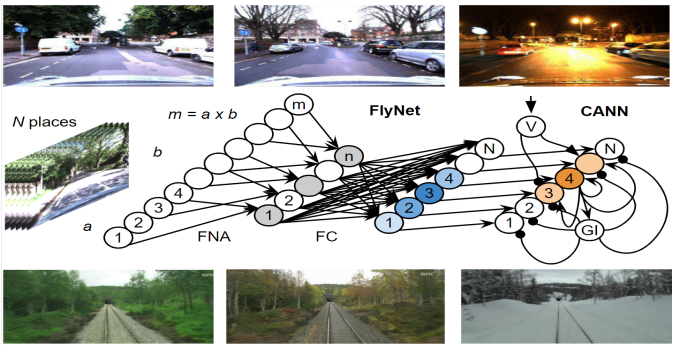

Fig. 1. **FlyNet+CANN hybrid neural architecture**. The FlyNet network comprises a hidden-layer inspired by the fruit fly olfactory neural circuit, FlyNet algorithm (FNA), and a fully-connected (FC) output layer. We integrate FlyNet with a continuous attractor neural network (CANN) to perform appearance-invariant visual place recognition. Experiments on two real-world datasets, Oxford RobotCar (top) and Nordland (bottom), show that our hybrid model achieves competitive results compared to conventional approaches, but with a fraction of computational footprint (see Fig. 2).

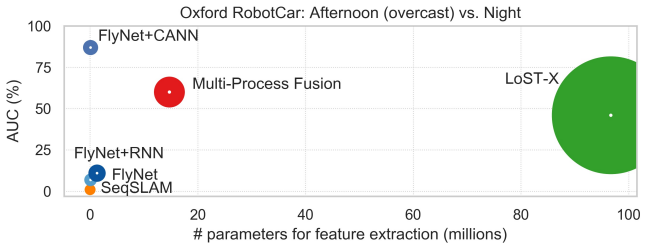

Fig. 2. **Oxford RobotCar AUC performance vs. Network Size**. Footprint comparison for the most challenging appearance change (day vs. night).

Manuscript received: September 5, 2019; Revised December 1, 2019; Accepted December 27, 2019.

This paper was recommended for publication by Editor Xinyu Liu upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the Peruvian Ministry of Education to M. Chancán and by an ARC Future Fellow FT140101229 to M. Milford.

[1]School of Electrical Engineering and Computer Science, Queensland University of Technology, Brisbane, QLD 4000, Australia

[2]Center for Brain Science & Department of Physics, Harvard University, Cambridge, MA 02138, USA

[3]School of Mechatronics Engineering, Universidad Nacional de Ingeniería, Lima, Rímac 15333, Peru. mchancanl@uni.pe

[4]Department of Biological Sciences, Macquarie University, Sydney, NSW 2109, Australia

Digital Object Identifier (DOI): see top of this page.

In this work, we propose a hybrid neural network that incorporates both computer-science- and neuroscience-oriented models, as in recent work [17], [18], but for VPR tasks for the first time[1]. Our approach comprises two key components (see Fig. 1): FlyNet, a compact neural network inspired by the *Drosophila* olfactory neural circuit, and a 1-d CANN as our temporal model to encode sequences of images and perform appearance-invariant VPR using real data.

Our resulting FlyNet+CANN model achieves competitive AUC results on two benchmark robotic datasets, but with far less parameters, minimal training time and smaller computational footprint than conventional deep learning and algorithmic based approaches. In Fig. 2, for instance, the area of the circle is proportional to the number of layers per deep neural network model, being 213 for LoST-X [19], 13 for Multi-Process Fusion (MPF) [20], and 2 for FlyNet+CANN.

The paper is structured as follows. Section II provides a brief overview of VPR research and the biological inspiration for our hybrid neural architecture; Section III describes the FlyNet model in detail; Sections IV and V present the experiments and evaluations respectively, where we compare our approach to three state-of-the-art VPR methods on two real-world datasets; and Section VI provides discussion around our neurally-inspired network as well as future work.

II. Related Work

This section outlines some key biological background for navigation in insect and mammalian brains, reviews the usage of deep learning based approaches for VPR, and discusses recent developments in temporal filtering techniques for sequential data to further improve performance.

A. Navigation in Biological Brains and Robots

Our understanding of how animals navigate using vision has been used as inspiration for designing effective localization, mapping and navigation algorithms. RatSLAM [21] is one example of this, using a model based on the rodent brain to perform visual SLAM over large environments for the first time [22]. Likewise, researchers have developed a range of robotic navigation models based on other animals including insects [23]–[25].

Insects such as ants, bees and flies exhibit great capabilities to navigate [26]–[30]. In fact, their brains share the same general structure [26], [31], with the central complex being closely associated with navigation, orientation and spatial learning [32], [33]. Visual place recognition is, however, most likely mediated by processing within the *mushroom bodies* (MB), a separate pair of structures within their brains that are known to be involved in classification, learning, and recognition of both olfactory and visual information in bees and ants [32]. They receive densely coded and highly processed input from the sensory lobes, which then connects *sparsely* to a large number of intrinsic neurons within the MB. Their structure has been likened to a multi-layer perceptron (MLP), and is considered optimal for learning and correctly classifying complex input [34].

[1]Project page: `mchancan.github.io/projects/FlyNet`

These impressive capabilities, achieved with relatively small brains, make them attractive models for roboticists. For FlyNet, we take inspiration from the olfactory neural circuit found in the *Drosophila melanogaster* fruit fly to design our network. Our focus here is primarily on taking high-level inspiration from the size and structure of the fly brain and investigating the extent to which it can be integrated with recurrent-based networks for VPR, much as in the early RatSLAM work and related development [35].

B. Deep Neural Networks for Visual Place Recognition

Over recent years, CNN have been applied to a range of recognition problems with great success, including VPR. These models can handle many challenging real-world environments with both visual appearance and viewpoint changes [36], [37], as well as large scale problems [19], [38]–[40]. Despite their success, these approaches typically rely on the usage of CNN models which are pretrained on various computer vision datasets [41] using millions of images [5], [6], [38]. Training CNN models in an end-to-end fashion specifically for VPR have also recently been proposed [4], [38], [42]. However, they are still using common network architectures, i.e., `AlexNet` [43], `VGG` [44], `ResNet` [45], with slight changes to perform VPR. All these systems share common undesirable characteristics with respect to their widespread deployability on real robots including large network sizes and extensive compute, and training requirements. In contrast, we propose the usage of compact architectures such as FlyNet to alleviate these requirements, while leveraging the temporal information found in most VPR datasets.

C. Modeling Temporal Relationships

To access and exploit the power of temporal information in many applications, researchers have developed a range of approaches including RNN such as long short-term memory (LSTM) [11]. These temporal-based approaches have been applied specifically to visual navigation [12] and spatial localization [13] in artificial agents. In a nice closure back to the inspiring biology, these approaches led to the arise of grid-like representations, among other cell types found in mammalian brains [46], when training RNN cells to perform path integration [14] and navigation [16]. RatSLAM [21], one of the older approaches to filtering temporal information in a neural network, incorporated multi-dimensional continuous attractor neural networks (CANN) with pre-assigned weights and structure set up to model the neural activity dynamics of place and grid cells found in the rat mammalian brain. Other non-neural techniques have been developed including SeqSLAM [10], which matches sequences of pre-processed frames to provide an estimate of place, with a range of subsequent works [47]–[49].

The work to date has captured many key aspects of the VPR problem, investigating complex but powerful deep learning-based approaches, biologically-inspired models that work in simulation or in small laboratory mazes, or larger mammalian-brain based models with competitive real-world robotics performance. In this paper, we attempt to merge the

desirable properties of several of these computer-science- and neuroscience-oriented models by developing a new hybrid bio-inspired neural network for VPR based on insect brain architectures such as FlyNet which is extremely compact and can uses the filtering capabilities of a 1-d CANN to achieve competitive localization results. We also show how our compact FlyNet model can easily be adapted to other filtering techniques including SeqSLAM and vanilla RNN.

III. Methods

We briefly describe the *fly algorithm* found in fruit fly brains that assign similar neural activity patterns to similar odors. We then present our FlyNet algorithm (FNA), inspired by the *fly algorithm* and describe our proposed single-frame, multi-frame, and hybrid models for visual place recognition.

A. Fly Algorithm

The fruit fly *Drosophila* olfactory neural circuit solves a similarity search problem by assigning similar neural activity patterns to similar odors [50], [51]. The *fly algorithm* performs a three-step procedure as the input odor goes through a three-layer neural circuit [50]. First, the firing rates across the first layer are centered to the same mean for all odors, removing the odor concentration dependence. Second, a binary, sparse random matrix connects the second and third layers, where each neuron in the third layer receives and sums about 10% of the firing rates from the second layer. Third, only the highest-firing 5% neurons across the third layer are used to generate a specific binary tag to the input odor using a winner-take-all (WTA) circuit. In summary, the *fly algorithm* mimics the pattern recognition capability found in the compact fly olfactory neural circuitry at a broad level and from a functional computer science perspective.

The *fly algorithm* is then formally defined in [50] as a binary locality-sensitive hash (LSH) function; a new class of LSH algorithms (see Eq. 1) but with relevant differences such as requiring significantly fewer computations as it uses sparse, binary random projections instead of dense, Gaussian random projections typical in LSH functions [52].

$$Pr[h(p) = h(q)] = sim(p,q) \tag{1}$$

where $sim(p,q)$ is the similarity function, and $h : \mathbb{R}^m \rightarrow \mathbb{Z}^n$ is the LSH function if for any $p, q \in \mathbb{R}^m$, Pr is $sim(p,q) \in [0,1]$.

B. Proposed FlyNet Algorithm

We leverage the *fly algorithm* from a computer vision perspective to propose the FlyNet Algorithm (FNA), see Algorithm 1. The FNA mapping uses 50% for the WTA circuit instead of 5% as in the *fly algorithm*, see Fig. 3. This number was found to provide good results across different experiments for VPR, with the desired compact structure. The FNA thus computes n random projections defined by $\mathbf{W}$ and the WTA circuit generates a binary output tag, which is a compact representation of the input image.

We perform an image preprocessing step, to obtain $\mathbf{x}$, before applying Algorithm 1. Details on this procedure are outlined in Section IV-A.

Algorithm 1 FlyNet Algorithm (FNA)

Input: $\mathbf{x} \in \mathbb{R}^m$

Output: $\mathbf{y} \in \mathbb{Z}^n$, $n < m$

1: Initialize $\mathbf{W} \in \mathbb{Z}^{n \times m}$: A binary, sparse random connection matrix between the input $\mathbf{x}$ and the output $\mathbf{y}$.
2: Compute the output $\mathbf{y} = \mathbf{W}\mathbf{x}$: Each output y_j receives and sums 10% randomly selected input values x_i.
3: WTA circuit: Set the top 50% output values y_i to 1, and the remaining to 0.

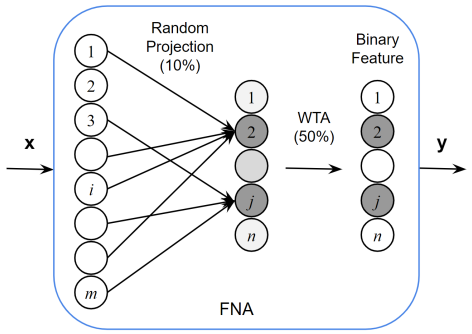

Fig. 3. **The FNA mapping**. The random projection shows only connections to y_2 and y_j, but all the units in that layer connect with 10% of the input.

C. Single-, Multi-frame, and Hybrid Models for VPR

We implement a range of VPR models that leverage the FNA compact representations, including one single-frame model, and three multi-frame models with temporal filtering capabilities, see Fig. 4.

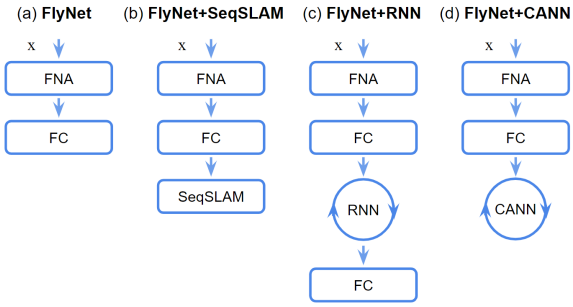

Fig. 4. **FlyNet baselines**. Our proposed (a) single- and (b, c) multi-frame models including the (d) hybrid FlyNet+CANN neural network for VPR.

1) ***FlyNet****:* The FlyNet model, shown in Fig. 4(a), is a two-layer neural network that comprises the FNA as a hidden-layer, and a fully-connected (FC) output layer. We configure FlyNet to have a grayscale input image dimension (m) of 32×64, and an output representation (n) of 64-d. The FNA output then feeds into a 1000-way *linear* MLP which compute a particular class score for each input image.

2) ***FlyNet+SeqSLAM****:* We incorporate the SeqSLAM algorithm [10] on top of our single-frame FlyNet network, as in previous research described in Sections I and II, see Fig. 4(b). The resulting model is a multi-frame system which we can compare along with our other temporal filtering based models FlyNet+RNN and FlyNet+CANN.

*3) **FlyNet+RNN**:* Is a purely neural model that incorporates a vanilla RNN on top of FlyNet for temporal information processing, see Fig. 4(c). We investigated the usage of other types of RNN such as gated recurrent units (GRU) and LSTM, however they showed no significant performance improvements despite having far more parameters.

*4) **FlyNet+CANN**:* Is our hybrid and also purely neural model for sequence-based VPR, see Fig. 4(d). We implemented a variation of the CANN architecture introduced in the RatSLAM work [22], but using a base implementation of a 1-d CANN proposed in [53], motivated by its suitability as a compact neural network-based way to implement the filtering capabilities of SeqSLAM [10]. As described in Section II-C, a CANN is a type of recurrent network that utilizes pre-assigned weights within its configuration. In Fig. 1 (middle) we show our detailed FlyNet+CANN implementation, where a unit in the CANN layer can excite or inhibit itself and units nearby using excitatory (arrows) or inhibitory (rounds) connections respectively, in contrast to an RNN. Also including a global inhibitor (GI) unit in its main structure. For this implementation, activity shifts in our 1-d CANN model, representing movement through the environment, were implemented with a direct shift and copy action. Although this could be implemented with more biologically faithful details such as velocity (V) units and asymmetric connections, as in prior CANN research [54].

IV. Experiments

To evaluate the capabilities of our proposed FlyNet based models, we conduct extensive experiments on two of the most widespread benchmarks used in VPR, the Nordland [55] and Oxford RobotCar [56] datasets (see Table I). We compare FlyNet (alone) with other related single-frame VPR methods and neural networks. Furthermore, we also compare our hybrid, multi-frame neural network with three state-of-the-art multi-frame VPR techniques such as SeqSLAM [10], LoST-X [19], and Multi-Process Fusion (MPF) [20]. In this section we describe our network configurations, dataset preparation, and existing state-of-the-art methods.

A. Real-World Datasets

*1) **Nordland**:* The Nordland dataset, introduced in [55] for VPR, comprises four single traverses of a train journey, in northern Norway, with extreme seasonal changes such as spring, summer, fall, and winter. The dataset is primarily used to evaluate visual appearance change, as instantiated through its four season coverage. In our experiments we use three traverses to perform VPR at 1fps as in [55]. We particularly use the summer subset for training our models, and the remaining to evaluate generalization capabilities.

*2) **Oxford RobotCar**:* The Oxford RobotCar dataset [56] provides over 100 traverses with different lighting (e.g. day, night) and weather (e.g. direct sun, overcast) conditions through a car ride in Oxford city; which implicitly contains various challenges of pose and occlusions such as pedestrians, vehicles, and bicycles for instance. In our evaluations we use the same subsets as in [19] including overcast (autumn) for training and both day and night for testing.

TABLE I
Sequence-Based Datasets for VPR (reference/query)

Dataset	Appearance Changes	Viewpoint Changes
Nordland	Small (summer/fall) Extreme (summer/winter)	Small
Oxford RobotCar	Small (overcast/day) Extreme (day/night)	Moderate

Data Preprocessing. In all our experiments, we use a subset of 1000 images per dataset and provide full resolution images to all the models, being 1920×1080 for Nordland and 1280×960 for Oxford RobotCar. For our FlyNet baselines, we convert the images into single channel (grayscale) frames normalized between [0, 1], and then resize them to 32×64. While the state-of-the-art methods apply their default image preprocessing before feeding their models.

B. Evaluation Metrics

We evaluate the VPR performance of the models using precision-recall (PR) curves and area under the curve (AUC) metrics. The tolerance used to consider a query place as a correct match is being within 20 frames around the ground truth location for the Nordland dataset, and up to 50 meters (10 frames) away from the ground truth for the Oxford RobotCar dataset, as per previous research [20], [19], [57].

C. Comparison of FlyNet with other Neural Networks

We compare FlyNet (alone) with a range of related single-frame networks including FC models that use dropout techniques [58], a vanilla CNN model often used in visual navigation research [59], [60], and the well-known NetVLAD method [38]. We trained all these models end-to-end using a 1000-way *linear* MLP classifier (FC) — except for the off-the-shelf NetVLAD backbone and the FNA layer which sparse matrix $\mathbf{W}$ stays unchanged. The average classification accuracy results of ten experiments, using different seed numbers, are shown in Fig. 5. For FlyNet, we used its FC output layer as the *linear* classifier, see Fig. 4(a). For the FC network, we used a three-layer MLP with 64, 64 and 1000 units, as per the FlyNet architecture. We then obtained the FC+Dropout network by including dropout rates of 90% and 50% before the first and second layer of the FC model, to approximate the FlyNet sparsity and for fair comparison purposes. For the CNN model, we used 2 *convolutional* layers but with grayscale input images of 32×64 as in FlyNet. For NetVLAD, we used RGB images of 244×244, as required by its off-the-shelf VGG-16 [44] model, but we reduced their output feature from 4096-d to 64-d to be comparable in size with the FlyNet representation. It is worth noticing that we do not reduce the CNN and NetVLAD model sizes down to the same size as FlyNet as they use predefined (rigid) architectures inherent to their approaches. We used the Adam optimizer [61] for training, and a learning rate set to 0.001 for all our experiments.

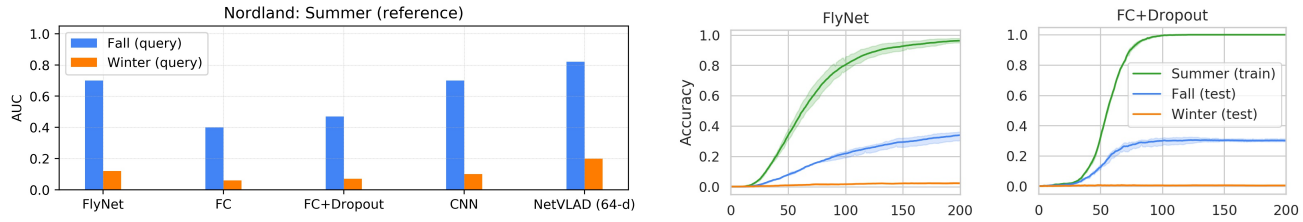

Fig. 5. **Comparison of FlyNet to other neural networks**. AUC performance comparison across different models on the Nordland dataset (left). Average accuracy over 10 training experiments vs. number of epochs for FlyNet and a fully-connected (FC) network with dropout (center, right).

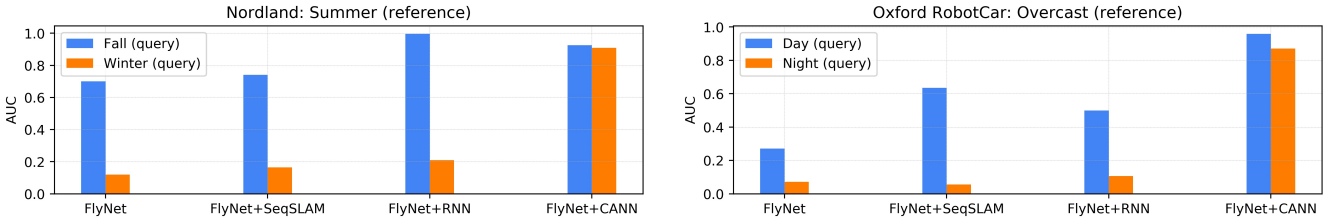

Fig. 6. AUC performance comparison of our single- and multi-frame FlyNet baselines on the Nordland (left) and Oxford RobotCar (right) datasets.

TABLE II
FLYNET BASELINES FOOTPRINT

Architecture	# layers	# params	# neurons
FlyNet	2	64k	1064
FlyNet+RNN	4	1.3m	2576
FlyNet+CANN	3	72k	2066

D. FlyNet Baselines Experiments

We trained and tested our four FlyNet baselines, described in Section III-C, in order to obtain our best performing model and compare it against existing VPR methods. In Table II, we show the number of layers, weights, and units for each FlyNet baseline. In FlyNet and FlyNet+RNN, the FNA hidden-layer used 64 units, and their FC layers used 1000 units; see Fig. 4(a, c). The number of recurrent units in the RNN-based model was 512. In FlyNet+CANN, the CANN layer used 1002 units. We show the AUC performance of our four FlyNet baselines on the Nordland and Oxford RobotCar datasets in Fig. 6 to further analyze them in Section V-A.

E. Comparison to existing State-of-the-Art VPR Methods

We compare our best performing FlyNet based multi-frame model with three state-of-the-art multi-frame VPR systems: the algorithmic technique SeqSLAM (without FlyNet attached), and two deep learning based methods such as LoST-X and the recent work Multi-Process Fusion (MPF).

1) ***SeqSLAM****:* SeqSLAM [10] shows state-of-the-art VPR results under challenging visual appearance changes. We use the MATLAB implementation in [55], with a sequence length of 20 frames, threshold of 1, and the remaining SeqSLAM parameters maintained its default values.

2) ***LoST-X****:* The multi-frame LoST-X pipeline [19] uses visual semantics to perform VPR with opposite viewpoints across day and night cycles. This method uses the RefineNet model [62] (a `ResNet-101` [45] based model) as semantic feature encoder, which is pre-trained on the Cityscapes dataset [63] for high-resolution semantic segmentation.

3) ***Multi-Process Fusion (MPF)****:* MPF [20] is also a multi-frame VPR technique. We use the `VGG-16` network [44] trained on Places365 [9] to encode the images and feed the MPF sequence-based algorithm.

V. RESULTS

In this section we analyze the experiments shown in Section IV, along with Figs. 5–6, and show the PR curves including related AUC results that compare our best performing model with existing state-of-the-art VPR methods.

A. FlyNet (single-frame) vs. other Networks and VPR models

From Fig. 5 (left), we can see that FlyNet is directly competitive with both FC networks, despite FlyNet having over 3 times fewer parameters (64k vs 199k) and also using 32 times less memory, as the FNA layer uses only 1-bit per binary weight, as per previous research [64], compared to the corresponding layer with 32-bit floating point weights in the FC models. On the other hand, for the CNN and NetVLAD models (6 and 234 times larger than FlyNet), the larger the model the better the results we obtained. Under small environmental changes (summer to fall) both networks achieved over 70% AUC, similar to FlyNet. However, under extreme visual changes (summer to winter) all these models show relatively similar results, below 12% AUC as for FlyNet, except for NetVLAD with 20%. In Fig. 5 (right), we show in detail the average training results of FlyNet against the FC model with dropout across 200 epochs.

B. FlyNet Baselines Evaluations

Although there are significant performance differences at a single-frame matching level, in Fig. 6 we can see that when using sequence-based filtering techniques these differences reduce significantly, meaning that using the more compact networks is viable in a range of applications where temporal filtering is practically feasible. It is possible then to leverage our compact FlyNet network and integrate it with a range of sequence-based methods such as SeqSLAM, RNNs and

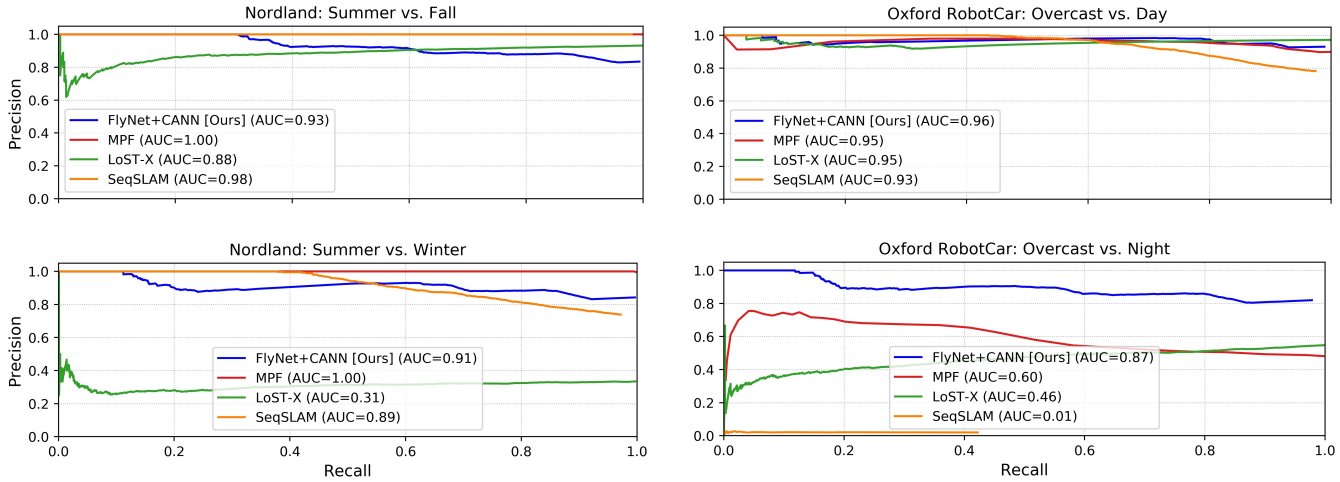

Fig. 7. PR curves of FlyNet+CANN vs. SeqSLAM, LoST-X and MPF on 1000-places of the Nordland (left) and Oxford RobotoCar dataset (right).

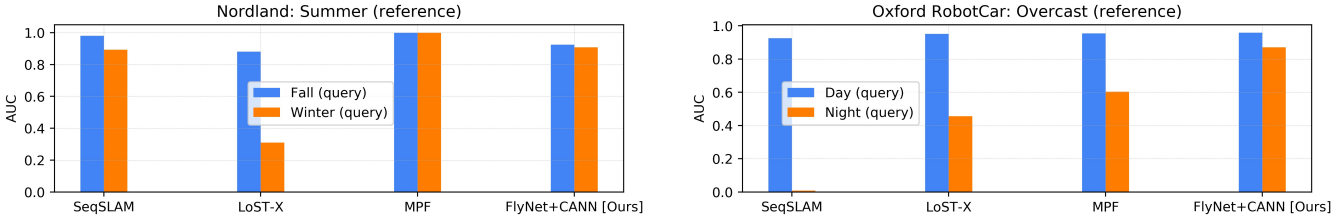

Fig. 8. AUC performance of FlyNet+CANN vs. SeqSLAM, LoST-X, and MPF on the Nordland (left) and Oxford RobotCar (right) dataset.

a 1-d CANN for VPR and achieve competitive results using a hybrid purely neural models such as FlyNet+CANN. For FlyNet+SeqSLAM, the performance of FlyNet (alone) was improved. Similarly, the RNN layer on top of the FlyNet model improved even further the results. However, by integrating FlyNet with a 1-d CANN we were able to outperform both models, even under extreme environmental changes (day/night, summer/winter), so we choose this hybrid approach to compare with the existing state-of-the-art methods in the following Section.

C. State-of-the-Art Analysis

Figs. 7–8 show the quantitative results of our FlyNet+CANN network and the other three state-of-the-art, multi-frame VPR methods: SeqSLAM, LoST-X, and MPF. Fig. 7 (left) shows the PR curves on the Nordland dataset, where it can be seen that MPF is performing better while being able to recall almost all places at 100% precision on both fall and winter testing traverses. Achieving also the highest AUC results, see Fig. 8 (left). On the other hand, the semantic-based system LoST-X, is able to recall a few matches at 100% precision on both testing traverses (fall, winter). In contrast, FlyNet+CANN achieves state-of-the-art results comparable with SeqSLAM and MPF in all the tested traverses, as can be seen in Fig. 8 (left).

Similarly, PR performance curves on Oxford RobotCar are shown in Fig. 7 (right). Also notable in this case is that FlyNet+CANN again achieves state-of-the-art results that are now comparable with SeqSLAM, LoST-X, and MPF approaches. The FlyNet+CANN model consistently maintains its AUC performance even under extreme environmental changes

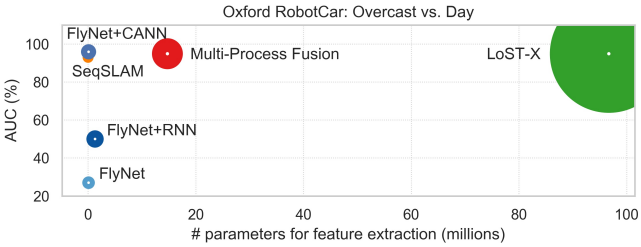

Fig. 9. **Oxford RobotCar AUC performance vs. Model Size**. Similar to Fig. 1, it compares small appearance changes (overcast vs. day).

(overcast to night cycle), as shown in Fig. 7 (right-bottom). In Fig. 8 (right), we also show how FlyNet+CANN outperforms the remaining methods in terms of AUC results.

D. Computational performance

The processing time required to perform appearance-invariant VPR by our FlyNet+CANN model is compared to the three state-of-the-art methods in terms of running time for (1) feature extraction, (2) feature matching between the reference and query traverses, and (3) average matching time of a single query image to the 1000 database images. This average time is calculated as (Feat. Ext. + Matching)/1000. Table III shows that our hybrid approach is 6.5, 310, and 1.5 times faster than MPF, LoST-X, and MPF respectively.

Fig. 9 shows a similar comparison presented in Fig. 2 but with moderated appearance changes (overcast vs. day) on the Oxford RobotCar dataset. In both figures, again, the area of the circle is proportional to the number of layers per model, except for the SeqSLAM system which performs an algorithmic matching procedure. We can see that the state-of-the-art systems MPF, LoST-X and SeqSLAM achieve

Fig. 10. **Top**: Sample images (reference) of the Nordland summer (left) and Oxford RobotCar overcast traversal (right). **Bottom**: Corresponding frames retrieved using our FlyNet+CANN network from the winter (left) and night traversal (right).

better AUC results than in Fig. 2 with 95%, 95% and 93% respectively, while FlyNet+CANN also present competitive performance with 96%.

TABLE III
PROCESSING TIME COMPARISON ON THE NORDLAND DATASET

VPR system	Feat. Ext.	Matching	Average
FlyNet+CANN	35 sec	25 sec	0.06 sec (16.66 Hz)
MPF	1.9 min	4.6 min	0.39 sec (2.56 Hz)
LoST-X	110 min	200 min	18.6 sec (0.05 Hz)
SeqSLAM	50 sec	40 sec	0.09 sec (11.11 Hz)

E. Influence of bio-inspiration

From the results shown in Figs. 7–9 and Table III, we can see how our hybrid FlyNet+CANN neural network achieve competitive AUC results with existing deep-learning- and algorithmic-based VPR methods, but with significantly fewer parameters, a smaller footprint and reduced processing time. The influence of bio-inspiration in developing this model enabled the design and evaluation of an extremely compact and high-performing neural architecture compared to deep-learning-based approaches. For FlyNet+CANN, the compact pattern recognition capabilities of the FlyNet network required only 64k parameters to efficiently encode our images, compared to 14.7m for MPF, and 96.68m for LoST-X (see Fig. 9). On the other hand, the integration of a neuroscience-oriented 1-d CANN model, on top of FlyNet, has enabled us to use a lower performance but fast network by temporally filtering the output to get better results for the whole place recognition model, which allowed our hybrid approach to be up to three orders of magnitude faster than existing methods.

F. Generalization Results

Fig. 10 shows the qualitative results for our best performing baseline FlyNet+CANN on both benchmark datasets. The proposed model is able to correctly match places under significant environmental changes such as summer to winter for the Nordland dataset, and day to night for the Oxford RobotCar dataset.

VI. CONCLUSION

In this paper, we presented a novel bio-inspired visual place recognition hybrid model based by the part on the fruit fly brain and integrated with a compact continuous attractor neural network. Our proposed model was able to achieve competitive performance compared to benchmark systems that have much larger network storage and compute footprints. It was also, to the best of our knowledge, the furthest in capability an insect-based place recognition system has been pushed with respect to demonstrating real-world appearance-invariant without resorting to full deep learning architectures.

Future research bridging the divide between well-characterized insect neural circuits [65], [66] as well as recent deep neural network architectures and computational models of network dynamics related to spatial memory and navigation [67] are likely to yield further performance and capability improvements, and may also shed new light of the functional purposes of these neural systems.

ACKNOWLEDGMENT

The authors would like to thanks Jake Bruce currently at Google DeepMind for insightful discussions about the potential ways to implement the FlyNet+RNN model, and also thanks to Stephen Hausler and Ming Xu for helpful discussions to perform our state-of-the-art comparison.

REFERENCES

[1] S. Lowry et al. "Visual Place Recognition: A Survey," in *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 1–19, Feb. 2016.
[2] Y. LeCun, Y. Bengio, and G. Hinton. "Deep Learning." *Nature*, vol. 521, pp. 436–444, Mar. 2015.
[3] Z. Chen, O. Lam, A. Jacobson, and M. Milford, "Convolutional Neural Network-based Place Recognition," in *Australasian Conference on Robotics and Automation (ACRA)*, 2014.
[4] Z. Chen et al. "Deep learning features at scale for visual place recognition," *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3223–3230, 2017.
[5] N. Sünderhauf et al. "Place Recognition with ConvNet Landmarks: Viewpoint-Robust, Condition-Robust, Training-Free," *Proceedings of Robotics: Science and Systems XIV*, 2015.
[6] N. Sünderhauf et al. "On the Performance of ConvNet Features for Place Recognition," *2015 IEEE International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, 2015, pp. 4297–4304.
[7] Z. Xin, X. Cui, J. Zhang, Y. Yang, and Y. Wang, "Real-time visual place recognition based on analyzing distribution of multi-scale CNN landmarks," *Journal of Intelligent & Robotic Systems*, 2018.
[8] J. Deng, W. Dong, R. Socher, L. Li, Kai Li and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, 2009, pp. 248–255.
[9] Zhou, Bolei et al. "Places: A 10 Million Image Database for Scene Recognition." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40 (2018): 1452–1464.
[10] M. Milford and G. F. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights," in *2012 IEEE International Conference on Robotics and Automation (ICRA)*, 2012.
[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[12] A. Banino et al. "Vector-based navigation using grid-like representations in artificial agents". *Nature*, 557(7705):429–433, 2018.
[13] C. J. Cueva, and X.-X. Wei. "Emergence of grid-like representations by training recurrent neural networks to perform spatial localization." ArXiv abs/1803.07770 (2018).
[14] McNaughton, B.L., Battaglia, F.P., Jensen, O., Moser, E.I., and Moser, M., "Path integration and the neural basis of the 'cognitive map'," *Nature Reviews Neuroscience*, 7, 663-678, 2006.

[15] Giocomo, L.M., Moser, M., and Moser, E.I., "Computational Models of Grid Cells," *Neuron*, 71, 589-603, 2011.

[16] B. Sorscher et al. "A unified theory for the origin of grid cells through the lens of pattern formation," in *Advances in Neural Information Processing Systems* 32 (NeurIPS), pages 10003–10013, 2019.

[17] Pei, Jing et al, "Towards artificial general intelligence with hybrid Tianjic chip architecture," Nature 572 (2019): 106-111.

[18] Yang, Z., Wu, Y., Wang, G., Yang, Y., Li, G., Deng, L., Zhu, J., and Shi, L., "DashNet: A Hybrid Artificial and Spiking Neural Network for High-speed Object Tracking," ArXiv, abs/1909.12942, 2019.

[19] S. Garg, N. Süenderhauf, and M. Milford, "LoST? Appearance-Invariant Place Recognition for Opposite Viewpoints using Visual Semantics," *Proceedings of Robotics: Science and Systems XIV*, 2018.

[20] S. Hausler, A. Jacobson and M. Milford, "Multi-Process Fusion: Visual Place Recognition Using Multiple Image Processing Methods," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1924–1931, 2019.

[21] 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 (ICRA)*, vol. 1, pp. 403–408, 2004.

[22] M. J. Milford and G. F. Wyeth, "Mapping a Suburb With a Single Camera Using a Biologically Inspired SLAM System," in *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1038–1053, Oct. 2008.

[23] A. Cope et al. "The green brain project – Developing a neuromimetic robotic honeybee," in *Biomimetic and Biohybrid Systems*, Berlin: Springer Berlin Heidelberg, 2013, pp. 362–363.

[24] B. Webb, "Using robots to model animals: a cricket test," *Robotics and Autonomous Systems*, vol. 16, no. 2, pp. 117–134, 1995.

[25] J. Dupeyroux, J. R. Serres, and S. Viollet, "Antbot: A six-legged walking robot able to home like desert ants in outdoor environments," *Science Robotics*, vol. 4, Feb. 2019.

[26] A. B. Barron and J. A. Plath, "The evolution of honey bee dance communication: a mechanistic perspective," *Journal of Experimental Biology*, vol. 220, no. 23, pp. 4339–4346, 2017.

[27] A. Narendra et al. "Mapping the navigational knowledge of individually foraging ants, *myrmecia croslandi*," *Proceedings. Biological sciences/The Royal Society*, vol. 280, no. 1765, 2013.

[28] J. Degen et al."Exploratory behaviour of honeybees during orientation flights," *Animal Behaviour*, vol. 102, pp. 45–57, 2015.

[29] T. Warren, Y. Giraldo, and M. Dickinson, "Celestial navigation in drosophila," *The Journal of experimental biology*, vol. 222, 2019.

[30] T. Ofstad, C. Zuker, and M. Reiser, "Visual place learning in drosophila melanogaster," *Nature*, vol. 474, pp. 204–209, 2011.

[31] J. Plath and A. Barron, "Current progress in understanding the functions of the insect central complex," *Current Opinion in Insect Science*, vol. 12, pp. 11–18, 2015.

[32] J. Plath et al. "Different roles for honey bee mushroom bodies and central complex in visual learning of colored lights in an aversive conditioning assay," *Frontiers in Behavioral Neuroscience*, vol. 11, 2017.

[33] K. Pfeiffer and U. Homberg, "Organization and functional roles of the central complex in the insect brain," *Annual Review of Entomology*, vol. 59, pp. 165–184, 2014. 2013.

[34] R. Huerta, "Learning pattern recognition and decision making in the insect brain," in *AIP Conference Proceedings*, vol. 1510, pp. 101–119, 2013.

[35] Yu, F., Shang, J., Hu, Y., and Milford, M. "NeuroSLAM: a brain-inspired SLAM system for 3D environments," Biological Cybernetics, 1-31, 2019.

[36] T. Sattler, et al, "Benchmarking 6DOF Outdoor Visual Localiza-tion in Changing Conditions, *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[37] M. A. Esfahani, K. Wu, S. Yuan, H. Wang, "DeepD-SAIR: Deep 6-DOF Camera Relocalization using Deblurred Semantic-Aware Image Representation for Large-Scale Outdoor Environments, *Image and Vision Computing*, 89: 120130, 2019.

[38] R. Arandjelović et al. "NetVLAD: CNN Architecture for Weakly Supervised Place Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 5297–5307.

[39] H. Noh et al. "Large-Scale Image Retrieval with Attentive Deep Local Features," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 3476–3485.

[40] A. Torii, R. Arandjelovix0107, J. Sivic, M. Okutomi, and T. Pajdla, "24/7 Place Recognition by View Synthesis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, pp. 257–271, 2015.

[41] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[42] Z. Chen et al. "Learning Context Flexible Attention Model for Long-Term Visual Place Recognition," *IEEE Robotics and Automation Letters (ICRA)*, vol. 3, no. 4, pp. 4015–4022, Oct 2018.

[43] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25*, pp. 1097–1105, 2012.

[44] K. Simonyan, and A. Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." *CoRR* abs/1409.1556, 2015.

[45] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, 2016, pp. 770–778.

[46] E. I. Moser, E. Kropff, and M.-B. Moser, "Place Cells, Grid Cells, and the Brains Spatial Representation System," *Annual Review of Neuroscience*, vol. 31, no. 1, pp. 69–89, 2008.

[47] T. Naseer et al. "Robust visual robot localization across seasons using network flows," in *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI Press, 2014, pp. 2564–2570.

[48] W. Churchill and P. Newman, "Experience-based navigation for long-term localisation," *The International Journal of Robotics Research*, vol. 32, no. 14, pp. 1645–1661, 2013.

[49] Y. Li et al. "Reliable patch trackers: Robust visual tracking by exploiting reliable patches," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 353–361, 2015.

[50] S. Dasgupta et al. "A neural algorithm for a fundamental computing problem," *Science*, vol. 358, no. 6364, pp. 793–796, 2017.

[51] C. Pehlevan, A. Genkin and D. B. Chklovskii, "A clustering neural network model of insect olfaction," *2017 51st Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, 2017, pp. 593-600.

[52] J. Wang, H. T. Shen, J. Song, J. Ji, "Hashing for similarity search: A survey," arXiv:1408.2927 [cs.DS]. 13 August 2014.

[53] Miller, P., "Dynamical systems, attractors, and neural circuits," F1000 Research, 5, F1000 Faculty Rev-992, 2016.

[54] P. Stratton, M. Milford, G. Wyeth, and J. Wiles, "Using strategic movement to calibrate a neural compass: A spiking network for tracking head direction in rats and robots," *PLOS ONE*, vol. 6, no. 10, pp. 1–15, 2011.

[55] N. Sünderhauf, P. Neubert, and P. Protzel, "Are we there yet? Challenging SeqSLAM on a 3000 km journey across all four seasons," in *IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, 2013.

[56] 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.

[57] J. Mao, X. Hu, X. He, L. Zhang, L. Wu and M. J. Milford, "Learning to Fuse Multiscale Features for Visual Place Recognition," in *IEEE Access*, vol. 7, pp. 5723-5735, 2019.

[58] N. Srivastava et al. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[59] Piotr Mirowski, Razvan Pascanu, Fabio Viola, Hubert Soyer, et al, "Learning to navigate in complex environments," ICLR, 2017.

[60] Lasse Espeholt, et al, "Impala: Scalable Distributed Deep-RL with IMPortance eighted Actor-Learner Architectures," ICML, 2018.

[61] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, abs/1412.6980, 2014.

[62] G. Lin, et al. "RefineNet: Multi-path Refinement Networks for High-Resolution Semantic Segmentation." *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[63] M. Cordts et al. "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[64] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, Y. Bengio, "Binarized Neural Networks", in *Advances in Neural Information Processing Systems 29*, pp. 4107-4115, 2016.

[65] L. Hernandez-Nunez et al. "Reverse-correlation analysis of navigation dynamics in Drosophila larva using optogenetics." *eLife*, 4 (2015).

[66] M. E. Berck et al. "The wiring diagram of a glomerular olfactory system." *eLife* 5 (2016), p. e14859.

[67] M. G. Campbell et al. "Principles governing the integration of landmark and self-motion cues in entorhinal cortical codes for navigation." *Nature Neuroscience* (2018) 21:10961106.