Robotics: Science and Systems 2018
Pittsburgh, PA, USA, June 26-30, 2018

Lightweight Unsupervised Deep Loop Closure

Nate Merrill and Guoquan Huang

***Abstract*—Robust efficient loop closure detection is essential for large-scale real-time SLAM. In this paper, we propose a novel *unsupervised* deep neural network architecture of a feature embedding for visual loop closure that is both reliable and compact. Our model is built upon the autoencoder architecture, tailored specifically to the problem at hand. To train our network, we inflict random noise on our input data as the denoising autoencoder does, but, instead of applying random dropout, we warp images with randomized projective transformations to emulate natural viewpoint changes due to robot motion. Moreover, we utilize the geometric information and illumination invariance provided by histogram of oriented gradients (HOG), forcing the encoder to reconstruct a HOG descriptor instead of the original image. As a result, our trained model extracts features robust to extreme variations in appearance directly from raw images, *without* the need for labeled training data or environment-specific training. We perform extensive experiments on various challenging datasets, showing that the proposed deep loop-closure model consistently outperforms the state-of-the-art methods in terms of effectiveness and efficiency. Our model is fast and reliable enough to close loops in real time with no dimensionality reduction, and capable of replacing generic off-the-shelf networks in state-of-the-art ConvNet-based loop closure systems.**

I. Introduction

It is critical to perform low-latency, high-fidelity, online loop closure detection (or place recognition) for real-time visual SLAM in order to enable bounded localization errors. This is a challenging problem, because the visual appearance of one location at different times can change dramatically due to varying viewpoints, illumination, weather, and dynamic objects (see Fig. 1). Numerous algorithms have recently been developed to address these issues [1]. Although these methods can perform well, in particular, by incorporating temporal information [2, 3, 4, 5, 6, 7], they may not be fast or robust enough for real-time performance in challenging environments.

Convolutional neural networks (ConvNets) [8] have recently become the state of the art for many vision-based classification tasks [9]. While off-the-shelf ConvNets are proven as useful feature embeddings for place recognition [10, 11, 12, 13, 14], specialized networks have also been constructed and trained to further improve performance [15, 16, 17, 18]. However, most of these ConvNet-based approaches suffer from either slow feature extraction [18, 14], slow querying [11, 12], or the need for a large amount of labeled data for training [15, 16, 17].

To address the aforementioned issues, in this paper, we construct a novel autoencoder-based ConvNet for loop closure that requires very few parameters, and train it using public data in an *unsupervised* manner. In particular, when building our autoencoder network, we exploit the advantages of classical geometric vision techniques – the histogram of oriented gradients (HOG) [19] that offers a convenient way to compress images while preserving salient features, and the projective transformation (homography) [20] that relates images with differing viewpoints. In contrast, we also incorporate the modern stacked convolutional autoencoder into the network to be data-driven. Consequently, the features extracted from our network are not only robust to extreme variations in appearance, but also lightweight and compact enough for real-time loop closing – even for resource-constrained systems.

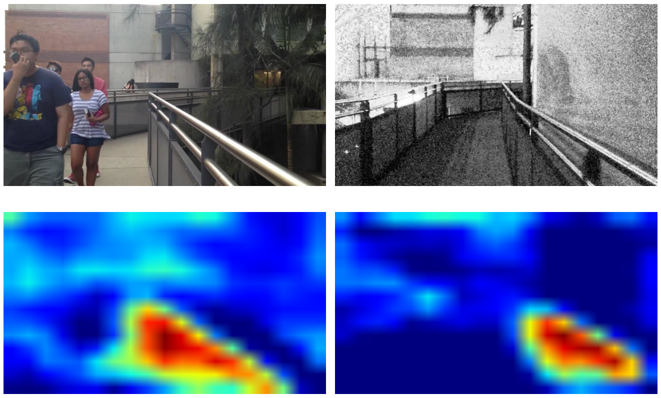

Fig. 1: An example image match from the *Gardens Point* dataset, which demonstrates large differences in viewpoint, dynamic objects, and illumination, as well as occlusions. Nevertheless, with the right image as the query, our proposed method correctly retrieves the left during our experiments (see Section IV), while all of the tested state-of-the-art methods retrieve incorrect images. Below each image, the first face of the descriptor layer, before flattening, is shown. Evidently, these visually dissimilar images are transformed into very similar activation maps.

The main contributions of this paper are the following:

- We design an *unsupervised*, convolutional autoencoder network architecture, tailored for loop closure, and amenable for efficient, robust place recognition.
- We perform extensive comparison studies of the proposed deep loop-closure model against the state-of-the-art methods on different datasets. To benefit the research community, we *open source* our code and pre-trained model used in this work along with a new dataset[1]

This work was partially supported by the University of Delaware (UD) College of Engineering, UD Cybersecurity Initiative, the NSF (IIS-1566129), and the DTRA (HDTRA1-16-1-0039).

The authors are with the Dept. of Computer and Information Sciences and Dept. of Mechanical Engineering, University of Delaware, Newark, DE 19716. Email: {nmerrill,ghuang}@udel.edu

[1] The code, dataset, as well as the pre-trained model from this work are available online: https://github.com/rpng/calc.

that captures extreme variations in viewpoint, weather, illumination, and dynamic objects in a single dataset.

The rest of the paper is structured as follows: After reviewing the related work in the next section, in Section III we present in detail the proposed deep loop closure network, including the network architecture, training scheme, and online usage. The proposed approach is tested extensively in Section IV – both against state-of-the-art algorithms and in a real-time loop-closure setting. Finally, we conclude the paper in Section V.

II. Related Work

Due to its importance, loop closure, or place recognition, has attracted significant attention in recent years. Many different algorithms have been introduced (see [1, 21] and references therein), with variant performance characteristics in terms of complexity, robustness and efficiency.

The approaches based on bag of words (BoW), such as FAB-MAP [22] and DBoW2 [4], are among the most popular for real-time visual SLAM systems (e.g., [23, 24, 25]). These methods build vocabulary trees based on point features of different descriptors [26, 27, 28], typically amenable for fast querying of matches; yet, they may fail when there are large variations in appearance between images. For this reason, SeqSLAM [2] was introduced to utilize the information provided by image sequences to construct a better hypothesis of loop closure. However, this method directly compares pixel values of down-sampled images and can fail under large variations in viewpoint. In contrast, different hand-crafted features are used in [29, 21, 30], where the loop closure is formulated as sparse optimization problems.

Recently, ConvNet-based approaches have risen in popularity. Chen et al. [10] first introduced the concept of using features produced by the off-the-shelf Overfeat network [31] as a holistic image descriptor – shown to outperform state-of-the-art place recognition systems. However, the descriptors extracted from such deep networks are too large to be used for real-time loop closure without approximating their similarity scores, which hinders their widespread deployment. Since then, many similar deep learning approaches have been introduced. For example, Sünderhauf et al. [12] employed ConvNet features to match subregions corresponding to landmarks, improving upon the performance of the holistic image descriptors, but the authors pointed out that it was *nowhere* near fast enough to be used in real time. Kenshimov et al. [32] proposed a method to omit parts of the activation maps from the neural networks in order to improve cross-seasonal place recognition. Hou et al. [14] combined ConvNet features with a bag of words scheme to speed up querying, while Bai et al. [33] combined ConvNet features with sequence searching to increase reliability. All of these methods rely on features extracted from generic neural networks that are not trained specifically for loop closure.

Others have trained their own networks for place recognition. For instance, Chen et al. [17] compiled a large place recognition-specific dataset to train classification networks for the sole purpose of feature embedding. NetVLAD [15] is an architecture that relies on geotags from Google Street View to label training images for a triplet loss scheme, where a triplet consists of two matching images and one non-matching image. Lopez-Antequera et al. [16] proposed a similar method using manually-labeled triplets, which reduces images into a single 128-dimensional vector. Their descriptor is shown to be useful for place recognition, and far more compact than that from any previous methods (e.g., [15]). However, all of these methods rely on *supervised* learning – requiring an immense amount of (human) effort to label images.

To address this issue, Gao and Zhang [18] recently introduced a stacked denoising autoencoder architecture [34] to solve the place recognition problem. Their method is shown to perform comparably to FAB-MAP 2.0 [35], but suffers from slow feature extraction. The model employed by Gao and Zhang [18] learns to reconstruct an image that has had random pixel values altered, but, if it is to be used for place recognition, it then has to be invariant to variations in viewpoint. Intuitively, it would be more useful to train an unsupervised model to reconstruct an image that has been altered to mimic the viewpoint variations that it will encounter in reality. With this observation, in this work, we build upon the autoencoder concept, utilizing the multi-view geometry of homographies and the invariance of HOG, to design a novel unsupervised architecture that is both more lightweight than the previously mentioned ConvNets, and trained to compensate for the specific types of visual appearance changes that are often encountered in loop closure scenarios.

III. Unsupervised Deep Loop Closure

In this section, we present in detail our method to construct, train, and utilize a novel autoencoder network for the loop closure task. Our model is designed to map high-dimensional raw images into a low-dimensional descriptor space, which is invariant to appearance differences. The proposed network and training scheme creates a compact robust feature embedding, while eliminating the need of image labeling.

A. Design Motivation

The standard denoising autoencoder network randomly drops input values during training to mitigate the effect of noise from actual signals during testing [34]. Clearly, such networks do not learn the variations in images that a loop closure system will encounter, such as changes in viewpoint, illumination and so on. Thus, the direct deployment of such autoencoders for place recognition may not be optimal.

Inspired by Sünderhauf et al. [12], where synthetic viewpoint alterations, in the form of simple translations, were used to *test* their place recognition system, we employ more generalized viewpoint alterations (projective transformations) to *train* our deep loop-closure model. Specifically, we inflict “noise” on the training inputs while modeling the natural variations due to robot motion, thus improving the performance of the autoencoder specifically for the place recognition task. However, these raw image pairs are not enough.

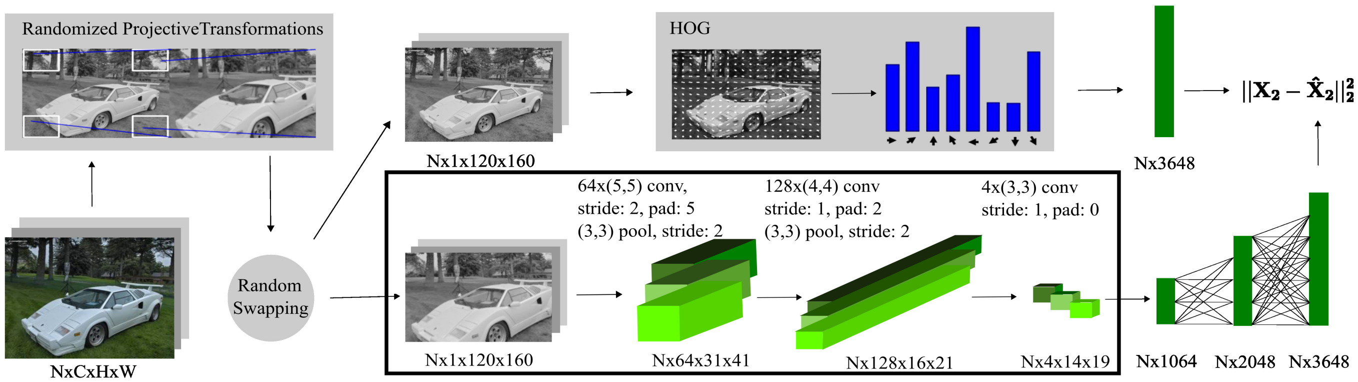

Fig. 2: The training pipeline for our deep model. In this architecture, the projective transformations and HOG descriptors are computed only once for the entire training dataset, and the results are then written to a database to use in training. Upon deployment, the batch size N is set to 1, and only the layers in the boxed area are used.

In experimentation with different network architectures, we constructed an autoencoder that shares the same encoding layers as the proposed model, but utilizes deconvolution and unpooling layers to attempt to reconstruct the other raw image from the pair. Without any extra optimization constraints, the network learned zero vectors, which suggests that the model needs more information to map one image from the pair to the other.

HOG, by design, provides geometric information about an image. Li et al. [36] showed that HOG description over segmented image patches can successfully be used to match images with vastly differing appearances in a place recognition setting. Furthermore, since HOG descriptors are fixed-length vectors for images of the same size, and can naturally be compared by the Euclidean distance, they are easily integrated into a neural network with ℓ_2 loss. However, since HOG relies on gradient orientation, it is not very robust to alterations in viewpoint, but, on the other hand, the image gradients are robust to illumination to some degree. Therefore, HOG provides the prior geometric knowledge needed by our network with the added benefit of learning illumination invariance, while the random projective transformations still create the added noise required to obtain a more useful feature embedding than just HOG can provide. Finally, it should be noted that we do not randomly place dynamic objects in the training pairs, even though our model is shown to be invariant to them (see Section IV). While doing so could potentially improve robustness to such occlusions, well-trained ConvNets are naturally invariant to such noise as Sünderhauf et al. [11] observed.

B. Network Architecture

Fig. 2 provides a visualization of the data flow from raw images to the loss layer. Before training begins, every image in the set of training images $\mathcal{I}$ is converted to grayscale, resized to 120×160, and used to create an image pair (see Fig. 3 and Algorithm 1). The HOG descriptor is computed for a randomly chosen image from each pair. We stack all the HOG descriptors from each batch of training images, denoted by $\mathbf{X}_2$ of dimension $N \times D$, where N is the batch size and D is the dimension of each HOG descriptor. The other image from the pair remains in raw form, and is stacked along with the other $N-1$ images in that training batch. The resulting tensor denoted by $\mathbf{X}_1$ has the dimension of $N \times 120 \times 160$.

The training network aims to reconstruct $\mathbf{X}_2$ given $\mathbf{X}_1$ using only two convolution and pooling paired layers, one pure convolution layer, and three fully-connected layers. Note that every layer has an activation after it. We use the rectified linear unit (ReLU) activation for the convolutional layers, while the sigmoid activation is chosen for the fully connected layers in order to better reconstruct the HOG descriptor (as it normalizes the data into $[0, 1]$). Additionally, since the Euclidean distance is naturally a good distance metric for HOG descriptors, we employ an ℓ_2 loss function to compare $\mathbf{X}_2$ with its reconstruction $\hat{\mathbf{X}}_2$. Upon deployment, we drop all layers but $\mathbf{X}_1$ and the three convolution layers. Our model is extremely lightweight compared to the state-of-the-art models for place recognition [16, 15, 12], taking up only 139 MB of GPU memory, allowing plenty of space for other processes – even on resource-constrained low-cost platforms.

C. Network Training

As previously mentioned, the proposed model does not require the training images to be labeled or contain any specific information – that is, any image from any scene can be used in the training set to improve our model. To illustrate this, we have trained our model on the *Places* dataset [37], which has over 8 million images originally designed for scene recognition. Figs. 2 and 3 contain a few examples of images from this dataset. While the majority of the images in the *Places* dataset are unrelated to any scene that a loop closure algorithm may encounter, the sheer number of images leads to improved performance over training on smaller datasets. Algorithm 1 outlines the main steps of utilizing such a dataset to create $\mathbf{T}_1$ and $\mathbf{T}_2$, the large tensors from which $\mathbf{X}_1$ and $\mathbf{X}_2$ are sampled during every iteration of stochastic gradient descent.

Algorithm 1 Generating Training Data

```
input: $\mathcal{I}$: A set of grayscale training images, resized to $H \times W$
output: $\mathbf{T}_1 \in \mathbb{R}^{M \times H \times W}$ and $\mathbf{T}_2 \in \mathbb{R}^{M \times D}$
define: $rand(\mathcal{A})$ as a map from set $\mathcal{A}$ to one of its elements, chosen at random
$W, H, D, M \leftarrow 160, 120, 3648, |\mathcal{I}|$
$\mathbf{T}_1 \leftarrow \mathbf{0}_{M \times H \times W}$
$\mathbf{T}_2 \leftarrow \mathbf{0}_{M \times D}$
$\mathbf{P}_c \leftarrow ((0,0),(0,H),(W,0),(W,H))$
for $i \in \mathbb{N} \cap [1, M]$ do
    $\mathbf{I} \leftarrow rand(\mathcal{I})$
    $\mathbf{P}_r \leftarrow randFourPts(W,H)$
    $\mathbf{H}_p \leftarrow estimateHomography(\mathbf{P}_r, \mathbf{P}_c)$
    $\mathbf{I}_w \leftarrow transform(\mathbf{I}, \mathbf{H}_p)$
    if $rand(\{0,1\})$ then
        $swap(\mathbf{I}, \mathbf{I}_w)$
    $\mathbf{T}_1^{(i)} \leftarrow \mathbf{I}$
    $\mathbf{T}_2^{(i)} \leftarrow calcHOG(\mathbf{I}_w)$
```

Given an image $\mathbf{I} \in \mathcal{I}$, we would like to automatically generate $\mathbf{I}_w$, which is of the same scene as $\mathbf{I}$ from a different viewpoint; this effect is achieved by applying a randomized 2D projective transformation matrix, $\mathbf{H}_p \in \mathbb{R}^{3 \times 3}$, to every pixel location in $\mathbf{I}$. To obtain this matrix, four points are randomly selected within the bounding boxes along the corners in the image $\mathbf{I}$ (see Fig. 3); $\mathbf{H}_p$ is then calculated to warp $\mathbf{I}$ such that those four points become the four corners of $\mathbf{I}_w$. We choose each bounding box of the point selection to be $H/4 \times W/4$ in order to avoid excessive distortion of $\mathbf{I}_w$, while still warping it enough to emulate a new perspective of the scene. Since every $\mathbf{I}_w$ appears zoomed in compared to $\mathbf{I}$, we randomly choose which of the images out of every pair to place into $\mathbf{T}_1$, avoiding unnecessary training biases.

We employ a HOG descriptor with large strides and a small window size to reduce the dimension of one of the images in each training pair – mapping an image $\mathbf{I} \in \mathbb{R}^{120 \times 160}$ of 19,200 pixels to $\mathbb{R}^{3,648}$. While this particular HOG descriptor may not be very informative for place recognition because of its aggressive data compression, it helps the autoencoder model to learn a good image encoding as mentioned in Section III-A. To construct and train our model, we utilize the Caffe Deep Learning Library [38] due to its efficiency. We train our model for roughly 42 epochs with a fixed learning rate of 9×10^{-4}. Based on Krizhevsky et al. [39], we choose a momentum of 0.9, and weight decay of 5×10^{-4}.

D. Online Use

Once our model is trained, upon its deployment for online use, we create a database of the descriptors extracted by our model and later query it to find loop closure candidates. While K-D trees [40] are a popular means to create such databases for nearest-neighbors searches, there is no speed up over a linear search for 1,064-dimensional vectors – even when the search is approximated [41]. For this reason, we use the simple linear search method. Additionally, as the descriptors are compact enough, their similarity can be calculated directly with no dimensionality reduction.

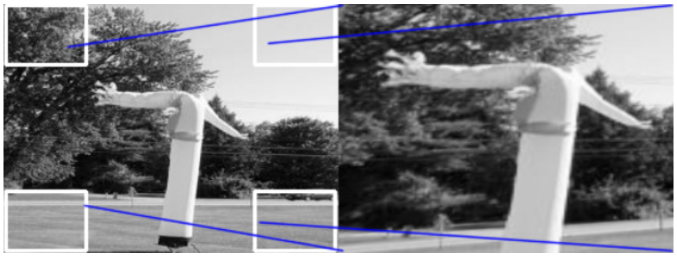

Fig. 3: An example of a possible training image pair. The four bounding boxes shown on the raw image (left) highlight the possible locations of each randomly selected point. Once the point correspondences are generated, a 2D projective transformation is calculated such that each one of those points becomes a corner of the warped image (right) after applying it. The randomly selected points are shown on the left, connected to their corresponding locations in the warped image shown on the right.

We seek to emphasize that our method of creating and querying the database with the descriptors extracted from our model is simple but effective; albeit, we are able to achieve faster-than-real-time querying speed with minimal memory usage (see Section IV-E). Furthermore, since many new ConvNet-based place recognition systems [12, 32, 14, 33] rely on features from bulky off-the-shelf networks, our light-weight model can potentially be utilized in many of these systems to achieve speedups with competative accuracy (see Section IV-G).

IV. Experimental Results

To validate the proposed unsupervised deep loop closure model, we have performed extensive comparison studies on various datasets with the state-of-the-art approaches as well as other benchmarks where applicable. While runtime is used as the criterion for evaluating efficiency, we utilize the precision-recall curve, a standard method to evaluate binary classification, to quantify effectiveness. While there are many ways to interpret a precision-recall curve, we primarily use: (i) the area under curve (AUC), where a higher AUC is desired; and (ii) the maximum recall rate with 100% precision, denoted by r, where again a higher value is desired. This can be observed visually in any precision-recall curve, as it will be the recall rate where the precision first dips down from 1.0. By observing both of these values, we obtain a comprehensive picture about how well the considered algorithms can generalize; however, the r value is slightly more desirable in practice, since one binary classifier can have non-perfect precision for all recall rates despite a high AUC.

For the results presented below, we compare the proposed approach with the following: (i) **Autoencoder**: A traditional denoising convolutional autoencoder. This model has the same encoding layers as our proposed model upon deployment, but,

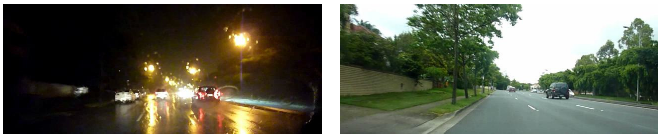

Fig. 4: An example image pair from the Alderley dataset. Note that these frames are extremely difficult to match, even for a human.

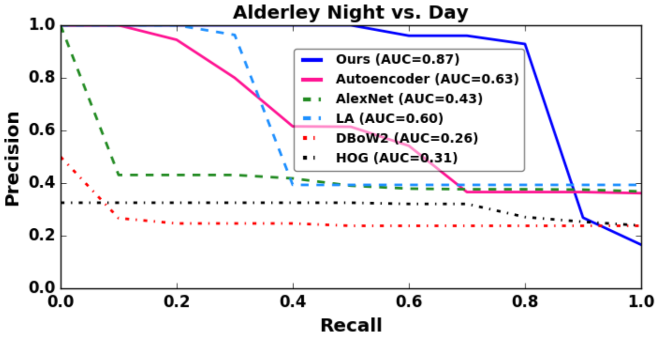

Fig. 5: Our method outperforms the state-of-the-art algorithms on the Alderley dataset, with the highest AUC and r value.

instead of reconstructing HOG descriptors of warped images, it utilizes deconvolution and unpooling layers to reconstruct the original image and is subject to random dropout during training. (ii) **LA**: The model from Lopez-Antequera et al. [16], which has comparable efficiency as our *unsupervised* model while requiring labeled data for training (i.e. *supervised*), making any retraining difficult. (iii) **DBoW2**: We use the DBoW2 vocabulary tree from the state-of-the-art ORB-SLAM [24, 23]. (iv) **AlexNet**: Sünderhauf et al. [11] found AlexNet `conv3` to be the most robust layer for place recognition; however, it was also noted that the 64,896-dimensional vector produced was too large to perform real-time database queries. Therefore, we apply Gaussian random projection (GRP) [42, 43] as in [12] to compress the `conv3` layer to the same size as the descriptors from the proposed model. In our tests, we use the AlexNet trained by BVLC. (v) **HOG**: Although the 3,648-dimensional HOG descriptor is used to train our model, we include this comparison merely to show that our model is able to learn a better feature than the original reduced HOG, rather than to show the ability of HOG as descriptors for place recognition. Note that for all of these methods, we use a single nearest-neighbor linear search in order to purely compare the ability of each descriptor to match places. At last, it should be pointed out that in all the following experiments, our approach uses the *same* model trained on a completely *different* dataset from the testing datasets, showing that the proposed deep loop closure network does not require environment-specific training.

A. The Alderley Dataset

The Alderley dataset was first introduced in SeqSLAM [2] and is composed of two image sequences, extracted from videos taken during a rainy night and a sunny day. Fig. 4 shows an example image match from this dataset; it is very difficult even for human to realize these images are of the same place. Frame correspondences are included in the dataset, providing ground truth for place recognition, with an added tolerance for multiple sequential frames of the same location. We test on the last 200 frames of each sequence. The comparison results are shown in Fig. 5. Clearly, our method is the most robust in this case, taking the highest AUC and r value by large margins. Interestingly, the regular autoencoder performs well here. Note that the model from [16] was trained on a different subset of the Alderley dataset than used here, giving their model an advantage over the others that have not been trained on any of the Alderley dataset. Nevertheless, our model is still more robust in this experiment, while the other methods are failing due to the significant differences in appearance in this dataset.

B. The Gardens Point Dataset

The Gardens Point dataset consists of three traversals through the QUT campus in Brisbane, Australia. In this dataset, there are two day-time traversals – one tends to contain images of the left side of the walking path, and the other contains the right side. Additionally, there is one night-time traversal, which tends to the right side of the path as well. Unlike in the Alderley dataset, image i from one sequence matches image i from any of the other two. We utilize this, as well as an added tolerance for multiple sequential images of the same location, to define the ground truth for this experiment in addition to the remaining precision-recall experiments in this work, since the rest of the datasets follow the same format.

An example of this dataset is shown in Fig. 1, while Fig. 6 shows the comparative results. Fig. 6 (top) is for the *day left and day right sequences*. Our method, AlexNet, and LA perform comparably in this dataset, and even the autoencoder is not far behind; however, we will see that this trend of comparable performance does not carry throughout the experiments. DBoW2 is competitive in this experiment, but falls short of our method, AlexNet, and LA. The HOG descriptor we used to train our model is clearly not nearly as robust as our final descriptor, even in this daytime dataset – one of the easier datasets used in experimentation. To further challenge these methods, we use the *night-time sequence* from the Gardens Point dataset – the results of which are shown in Fig. 6 (bottom). In this case, our method takes the highest r value, and the second-highest AUC. DBoW2, HOG, and the autoencoder completely fail in this test. Although the performance of DBoW2 could most likely could be improved by training the ORB vocabulary tree on the night-time images, we want to test all of these methods void of any environment-specific training for the purpose of generalization.

C. The Nordland Dataset

The Nordland dataset, one of the most challenging place recognition datasets to date [32], consists of four time-synchronized videos of train journeys through Norway. Each of the four 9-hour long sequences corresponds to a different

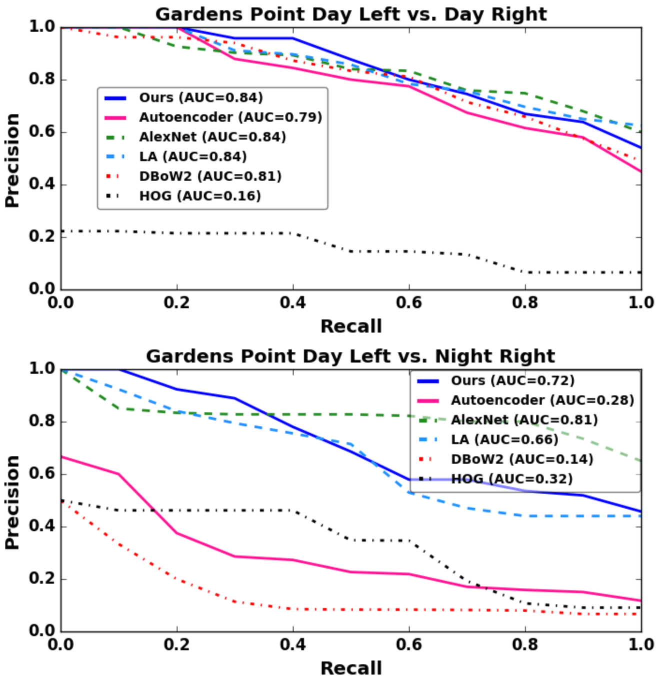

Fig. 6: The comparison results on the Gardens Point dataset. (top) Our method performs comparably with [16] (which, however, is a supervised learning approach) in the *day-time sequence*, while (bottom) our method outperforms its competitors in the *night-time sequence*.

season, creating a difficult challenge for cross-seasonal place recognition. In addition to seasonal variation, the images also contain extreme blurring from the fast speed of the train. Fig. 7 shows an example of an image pair from this dataset. We test our method on one of the most difficult sequence pairs, *Winter versus Spring*. Specifically, we extracted 5,357 frames from these two videos. This experiment was performed using frames 29 to 200, as this was the first sequence we found where the train was constantly in motion and outside of tunnels. Note that images from inside the tunnels are completely black, and therefore useless for experimentation; additionally, if the train was stopped at a station, there were too many sequential images of the same location, causing large biases in the precision-recall curves.

The results of this experiment are shown in Fig. 8. It should be noted that Lopez-Antequera et al. [16] used all but the last hour of each Nordland sequence in training their model; this implies that their model has seen this testing data in the training phase. However, even with this incredible disadvantage, our model outperforms theirs, along with other methods in this experiment.

D. Our Campus Loop Dataset

The Nordland dataset provides extreme weather variations, the Gardens Point dataset provides extreme brightness and viewpoint variations, as well as many dynamic objects, while the Alderley dataset provides all but large viewpoint variations.

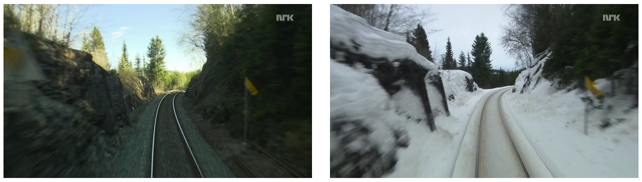

Fig. 7: An example image pair from the Nordland dataset. The left image is from the spring sequence while the right one is from the winter.

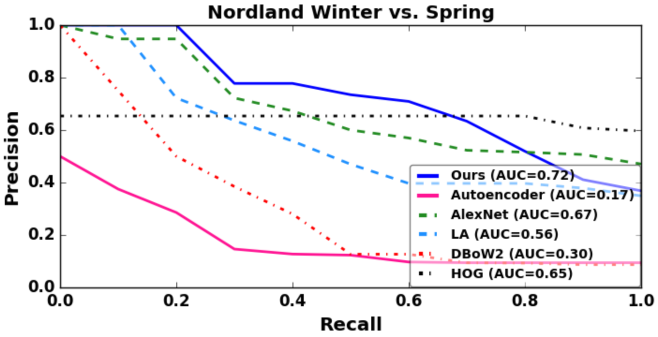

Fig. 8: Comparison results on the Nordland dataset. Our method is observed to be more robust to the seasonal changes provided by this subset of the winter and spring sequences.

However, we found that no dataset can provide all of these challenges. Therefore, we collect our own dataset, termed as the *Campus Loop dataset*. The dataset consists of two sequences of 100 images each. The sequences are a mix of indoor and outdoor images in a campus environment. The first sequence was taken on a snowy day, when it was very cloudy, while the second was taken nine days later, when most of the snow had melted and the sun was out. The indoor images obviously do not vary as much with this weather change. Additionally, each image match contains varying perspectives and many dynamic objects, making this one of the most challenging publicly-available place recognition datasets. Fig. 9 shows an example of an image pair from this dataset.

The results of experimentation with this dataset are shown in Fig. 10. As expected, across the board the performance is worse than any other dataset thus far. Nevertheless, comparatively, our method is the most robust to the challenges presented in this new dataset. The model from Lopez-Antequera et al. [16] falls flat in this experiment, performing significantly worse than the other three deep-learning methods, and falling short of even DBoW2 in AUC.

E. Runtime Evaluation

To validate the efficiency of our approach, we perform runtime evaluations of both descriptor computing time, and database querying time for a single-nearest neighbor search. These tests are conducted on affordable hardware to allow for better reproducibility – specifically, an i7-6700HQ CPU, and a GeForce GTX 960M GPU. Note that in this test we only

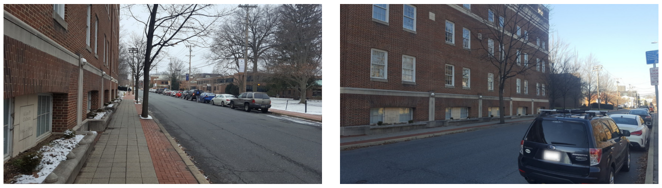

Fig. 9: An image pair example from our Campus Loop dataset, which has extreme variations in viewpoint, weather, and dynamic objects.

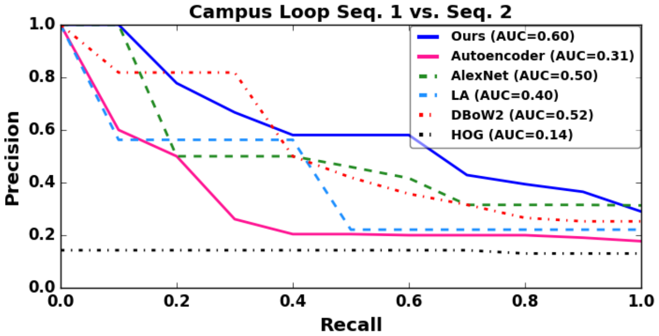

Fig. 10: Our approach outperforms the other benchmark methods on our own Campus Loop dataset, with the highest r value while tying with AlexNet conv3 for the highest AUC.

TABLE I: Times (in milliseconds) to extract features and query a database of 4,541 images on the KITTI dataset.

Method	Extract (GPU)		Extract (CPU)		Query	
	μ	σ	μ	σ	μ	σ
Ours	**0.862**	0.025	44.0	2.98	**1.47**	0.031
DBoW2	N/A	N/A	15.8	3.08	4.25	0.547
AlexNet (no GRP)	2.13	0.038	405.0	17.4	80.8	0.708
AlexNet	16.6	0.658	418.0	17.8	N/A	N/A

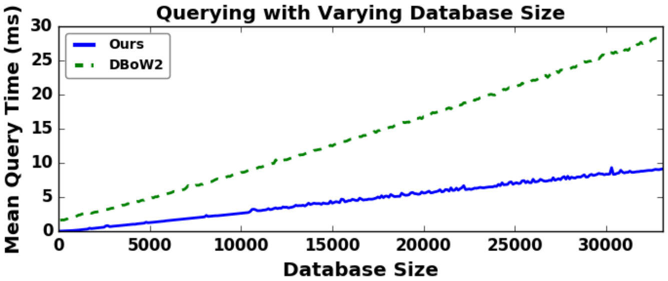

Fig. 11: The proposed method performs queries faster than DBoW2 with varying database size.

compare against DBoW2 and AlexNet due to the following: (i) Lopez-Antequera et al. [16] do not provide an open-sourced library for performing image matches, so it would not necessarily be fair to time their models in our code. They report 1.8 millisecond descriptor computing time using a GPU, which is slower than ours. However, their descriptor is smaller than ours, so it should be cheaper to query, while ours is shown to be competative (if not better) in accuracy, and more convenient to fine-tune or retrain. (ii) The reduced HOG is presented in the preceding tests only to show that our model learns a better version of it, rendering its runtime irrelevant.

We choose DBoW2 with ORB features as one benchmark, since it is one of the fastest place recognition libraries used in many state-of-the-art SLAM systems (e.g., [23, 24]). Additionally, we choose AlexNet conv3 features both with GRP, compressed to 1,064 dimensions, and in original form, since it is a popular choice for ConvNet-based place recognition (i.e. in [12, 14, 33, 32]). Note that AlexNet has been modified to only contain up to the conv3 layer here for fair testing. The *KITTI Visual Odometry* dataset sequence 00 [44] is used as testing data for the first experiment. We utilize the 4,541 376 × 1241 stereo pairs from this sequence to construct two subsets, placing all of the left images in the database, and using the right images for querying. Table I shows the results of this experiment, where feature extraction time refers to the time between starting with a raw image and ending with having that image's representation inserted into the respective database. The query times do not include any descriptor calculation times. Note that DBoW2 has no GPU implementation, and AlexNet with GRP produces features of the same size as ours, so the query times will be the same. From Table I, it is clear that our method is faster than the others for feature extraction when a GPU is used to make forward passes through the net, and is still reasonably fast when using a CPU. Additionally, our method for querying, though it is simple, outperforms DBoW2 in terms of speed in this experiment.

We also test the query speed for a variable-sized database, comparing only to DBoW2 – the most competative candidate from Table I. We use the large *St. Lucia* dataset [45], which, similar to KITTI, is a sequence of stereo pairs. However, this sequence contains over 30,000 stereo pairs, making it very useful for testing a variable database size. The left images are used for the database, and a subset of the right images is used for querying. Fig. 11 shows the results of this experiment, from which it is evident that our querying method is inexpensive, even for very large databases – larger than that created by a typical SLAM system.

F. Online Loop Closure

Precision-recall curves are a good metric for binary classification, but they do not fully prove that our method is capable of accurately closing loops in practice. Therefore, we perform real-time loop closure using an extremely simple application of our model on KITTI [44] sequences 00 and 05. In this experiment, we simulate keyframe selection by using every seventh frame for loop detection. A loop closure hypothesis is proposed if a database query score is above an a-priori threshold τ, and a loop is determined closed if three consecutive queries retrieve descriptors within six frames of the first query. We exclude the most recent images from the search space, and do not start loop detection until the database is of sufficient size. We choose τ from the precisions and recalls shown in Fig. 6 (bottom) such that it maximizes the recall rate with

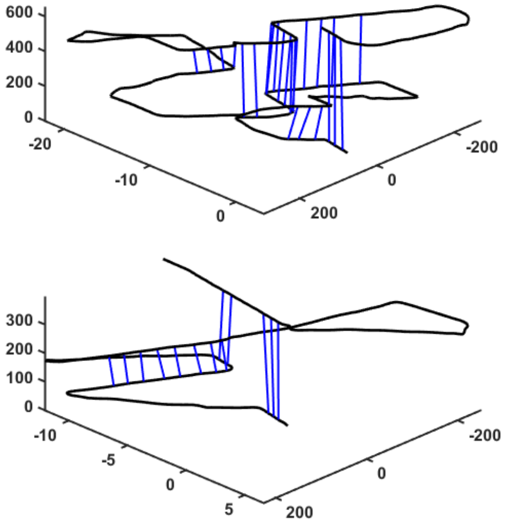

Fig. 12: The results of online loop closure using KITTI 00 and 05, respectively. The 2D location of the trajectory is represented on the x-y plane, and the z-axis is the current keyframe number.

perfect precision. Fig. 12 shows the results of this experiment. Clearly our method is able to consistently close loops on a practical SLAM dataset using a threshold from completely unrelated ground truth data, which shows that it is ready to use in a real-time SLAM system. Additionally, this application of our model for online loop closure is extremely simple, and can be improved upon easily by looking at the k-nearest neighbors instead of the single-nearest neighbor, adding extra false positive rejection methods (i.e. a geometric check), or utilizing any of the methods described in the next section.

G. Integration into ConvNet-Based Place Recognition Systems

As stated in Section III-D, our model can easily be integrated into off-the-shelf ConvNet-based place recognition systems [12, 32, 14, 33] for faster feature extraction. These methods build upon the use of holistic image descriptors, improving performance in different cases. They treat the ConvNet as a black box for image description – throwing out image classifications from classification networks. Many of these methods are forced to reduce the dimension of the ConvNet features to minimize runtime, while our model already produces a small enough descriptor for real-time use, and is smaller and faster than the typical classification network used by these methods.

To prove this, we reproduce the state-of-the-art landmark-based place recognition system presented in [12], replacing Edge Boxes [46] with BING [47] as Hou et al. [14] did to reduce runtime, and replacing the dimension-reduced AlexNet `conv3` landmark descriptor with that from our model. Sünderhauf et al. [12] proposed reducing the AlexNet `conv3` layer to 1,024 dimensions, while ours is naturally 1,064, so we do not need to reduce it further – avoiding the cost of the 1024×64896 by $64896 \times m$ matrix multiplication required to project m landmark descriptors (using AlexNet) into 1,024 dimensions, which must be done every time a new image is added to the database. The results of this experiment can be seen in Fig. 13. The landmark-based method offers an enormous improvement over the holistic image descriptor – approaching perfect performance on the Gardens Point daytime dataset. Our model is seamlessly integrated into this system, which suggests that it can easily replace bloated classification networks in other such ConvNet-based place recognition systems.

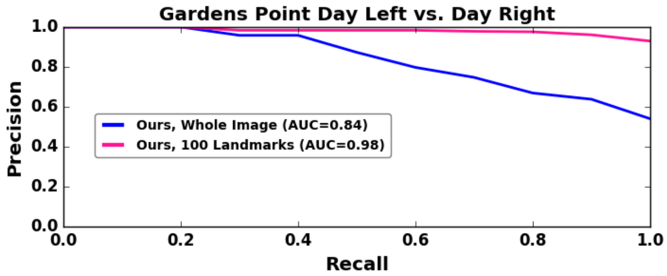

Fig. 13: The landmark-based method shows a vast improvement over the holistic approach – nearing perfect performance.

V. Conclusions

We have presented a novel unsupervised deep neural network for fast and robust loop closure, applicable in visual SLAM. Built upon the denoising autoencoder architecture, we apply randomized projective transformations to images in order to capture extreme variations in viewpoints due to robot motion, while employing the fixed-length HOG descriptor to help our network better learn the geometry of scenes. The proposed model allows for vast amounts of data to be used in training, since none of it needs to be labeled or contain any special information. Furthermore, although our pre-trained model generalizes well in its current state, it is easy to fine-tune or retrain due to our unsupervised design – increasing the likelihood of improvement as more data becomes available.

We have performed thorough comparison studies on different datasets against the state-of-the-art image description methods for place recognition, where the extensive experimental results have shown that the proposed deep loop closure method generally outperforms the benchmarks in terms of both effectiveness (precision-recall) and efficiency (runtime). Our model is compact, robust, and fast – making it a promising candidate to replace larger, slower classification networks in ConvNet-based place recognition systems, as we have shown by reproducing [12]. Due to its lightweight yet robust design, our model is suitable for use in real-time SLAM systems – in particular, direct algorithms [48, 25, 49, 50] where no intermediate image representation is needed. We aim to provide an out-of-the-box solution for loop closure, and, more generally, place recognition. We are currently working to integrate our model into various SLAM systems, applicable for autonomous navigation in challenging environments.

References

[1] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual Place Recognition: A Survey," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 1–19, Feb 2016.

[2] M. J. 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*, May 2012, pp. 1643–1649.

[3] T. Naseer, M. Ruhnke, C. Stachniss, L. Spinello, and W. Burgard, "Robust visual slam across seasons," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2015, pp. 2529–2535.

[4] D. Gálvez-López and J. D. Tardós, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, October 2012.

[5] E. Pepperell, P. I. Corke, and M. J. Milford, "All-environment visual place recognition with smart," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 1612–1618.

[6] M. Milford, W. Scheirer, E. Vig, A. Glover, O. Baumann, J. Mattingley, and D. Cox, "Condition-invariant, top-down visual place recognition," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 5571–5577.

[7] L. Bampis, A. Amanatiadis, and A. Gasteratos, "Fast loop-closure detection using visual-word-vectors from image sequences," *The International Journal of Robotics Research*, vol. 37, pp. 62–82, December 2017.

[8] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, Dec 1989.

[9] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. The MIT Press, 2016.

[10] Z. Chen, O. Lam, A. Jacobson, and M. Milford, "Convolutional neural network-based place recognition," *CoRR*, vol. abs/1411.1509, 2014.

[11] N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the performance of convnet features for place recognition," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2015, pp. 4297–4304.

[12] N. Sünderhauf, S. Shirazi, A. Jacobson, F. Dayoub, E. Pepperell, B. Upcroft, and M. Milford, "Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free," in *Robotics: Science and Systems*, Auditorium Antonianum, Rome, July 2015.

[13] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: An astounding baseline for recognition," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, ser. CVPRW '14. Washington, DC, USA: IEEE Computer Society, 2014, pp. 512–519.

[14] Y. Hou, H. Zhang, and S. Zhou, "Bocnf: efficient image matching with bag of convnet features for scalable and robust visual place recognition," *Autonomous Robots*, Nov 2017.

[15] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2017.

[16] M. Lopez-Antequera, R. Gomez-Ojeda, N. Petkov, and J. Gonzalez-Jimenez, "Appearance-invariant place recognition by discriminatively training a convolutional neural network," *Pattern Recognition Letters*, vol. 92, no. Supplement C, pp. 89 – 95, 2017.

[17] 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)*, May 2017, pp. 3223–3230.

[18] X. Gao and T. Zhang, "Unsupervised learning to detect loops using deep neural networks for visual SLAM system," *Autonomous Robots*, vol. 41, no. 1, pp. 1–18, Jan. 2017.

[19] 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, June 2005, pp. 886–893 vol. 1.

[20] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[21] Y. Latif, G. Huang, J. Leonard, and J. Neira, "Sparse optimization for robust and efficient loop closing," *Robotics and Autonomous Systems*, vol. 93, pp. 13–26, Jul. 2017.

[22] M. Cummins and P. Newman, "Fab-map: Probabilistic localization and mapping in the space of appearance," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 647–665, 2008.

[23] Mur-Artal, Raúl, Tardós, and J. D., "ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[24] Mur-Artal, Raúl, J. M. M. Montiel, Tardós, and J. D., "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[25] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 834–849.

[26] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Comput. Vis. Image Underst.*, vol. 110, no. 3, pp. 346–359, Jun. 2008.

[27] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International Conference on Computer Vision*, Nov 2011, pp. 2564–2571.

[28] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, and P. Fua, "Brief: Computing a local binary descriptor very fast," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1281–1298, July 2012.

[29] Y. Latif, G. Huang, J. Leonard, and J. Neira, "An online sparsity-cognizant loop-closure algorithm for visual navigation," in *Proc. of Robotics: Science and Systems*, Berkeley, CA, Jul. 12-16 2014.

[30] H. Zhang, F. Han, and H. Wang, "Robust multimodal sequence-based loop closure detection via structured sparsity," in *Proc. of Robotics: Science and Systems*, AnnArbor, Michigan, June 2016.

[31] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. Lecun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," 12 2013.

[32] C. Kenshimov, L. Bampis, B. Amirgaliyev, M. Arslanov, and A. Gasteratos, "Deep learning features exception for cross-season visual place recognition," *Pattern Recognition Letters*, vol. 100, pp. 124 – 130, 2017.

[33] D. Bai, C. Wang, B. Zhang, X. Yi, and X. Yang, "Sequence searching with cnn features for robust and fast visual place recognition," *Computers and Graphics*, vol. 70, pp. 270 – 280, 2018, cAD/Graphics 2017.

[34] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning*, ser. ICML '08. New York, NY, USA: ACM, 2008, pp. 1096–1103.

[35] 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.

[36] J. Li, R. M. Eustice, and M. Johnson-Roberson, "High-level visual features for underwater place recognition," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 3652–3659.

[37] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[38] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[39] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, ser. NIPS'12. USA: Curran Associates Inc., 2012, pp. 1097–1105.

[40] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Commun. ACM*, vol. 18, no. 9, pp. 509–517, Sep. 1975.

[41] M. Muja and D. G. Lowe, "Scalable nearest neighbor algorithms for high dimensional data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, 2014.

[42] S. Dasgupta, "Experiments with random projection," in *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, ser. UAI '00. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000, pp. 143–151.

[43] E. Bingham and H. Mannila, "Random projection in dimensionality reduction: Applications to image and text data," in *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '01. New York, NY, USA: ACM, 2001, pp. 245–250.

[44] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[45] M. Warren, D. McKinnon, H. He, and B. Upcroft, "Unaided stereo vision based pose estimation," in *Australasian Conference on Robotics and Automation*, G. Wyeth and B. Upcroft, Eds. Brisbane: Australian Robotics and Automation Association, 2010.

[46] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *ECCV*, 2014.

[47] M. M. Cheng, Z. Zhang, W. Y. Lin, and P. Torr, "Bing: Binarized normed gradients for objectness estimation at 300fps," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 3286–3293.

[48] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2017.

[49] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "Dtam: Dense tracking and mapping in real-time," in *2011 International Conference on Computer Vision*, Nov 2011, pp. 2320–2327.

[50] A. Concha, G. Loianno, V. Kumar, and J. Civera, "Visual-inertial direct SLAM," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 1331–1338.