Noname manuscript No.
(will be inserted by the editor)

Appearance-invariant place recognition by adversarially learning disentangled representation

Cao Qin · Yunzhou Zhang* · Yan Liu · Sonya Coleman · Dermot Kerr · Guanghao Lv

Received: date / Accepted: date

Abstract Place recognition is an essential component to address the problem of visual navigation and SLAM. The long-term place recognition is challenging as the environment exhibits significant variations across different times of the days, months, and seasons. In this paper, we view appearance changes as multiple domains and propose a Feature Disentanglement Network (FD-Net) based on a convolutional auto-encoder and adversarial learning to extract two independent deep features – content and appearance. In our network, the content feature is learned which only retains the content information of images through the competition with the discriminators and content encoder. Besides, we utilize the triplets loss to make the appearance feature encode the appearance information. The generated content features are directly used to measure the similarity of images without dimensionality reduction operations. We use datasets that contain extreme appearance changes to carry out experiments, which show how meaningful recall at 100% precision can be achieved by our proposed method where existing state-of-art approaches often get worse performance.

Keywords Visual place recognition, Changing environment, Adversarial learning, Representation disentanglement

* Corresponding author.
C. Qin, Y. Zhang, Y. Liu, G. Lv
College of Information Science and Engineering, Northeastern University, Shenyang, China
E-mail: zhangyunzhou@mail.neu.edu.cn(✉)

S. Coleman, D Kerr
Intelligent Systems Research Centre, University of Ulster, Derry, UK

1 Introduction

Visual-based navigation systems have achieved impressive results in the past few years and are widely used in robotic applications. When mobile robots work in unstructured and dynamic environments, their positioning performance will be degraded due to the drift and error of state estimation. Therefore, the robot should not only have enough ability to locate itself but also be able to rectify the estimated odometry or recover the robot's position in localization failure scenarios. The traditional way to enhance robustness is to recognize places that the robot has visited before by place recognition or loop closure detection (in SLAM). Tracking is relatively easy if the change of appearance between frames is gradual and small. However, the appearance of a place will change dramatically when the robot explores a long-time trajectory. Visual place recognition (vPR) becomes a very challenging problem because of different day periods (days and nights) or weather conditions (winter or summer). In general, place recognition methods describe the visual content of a given image by using descriptors. The first method is to represent the image as a whole and build descriptors such as Gist [46], color histogram [4] and HOG [13]. However, the performance can be influenced by many factors such as viewpoint changes. Another kind of method is to extract local descriptors such as SIFT [30] or SURF [5]. In this context, images are represented as vectors that account for the number of occurrences of local image features taken from a dictionary. This method is called bag of visual words (BoW) [12], which can work quickly and effectively for many applications [47,39]. Nonetheless, the BoW-based method is highly sensitive to lighting and environmental differences.

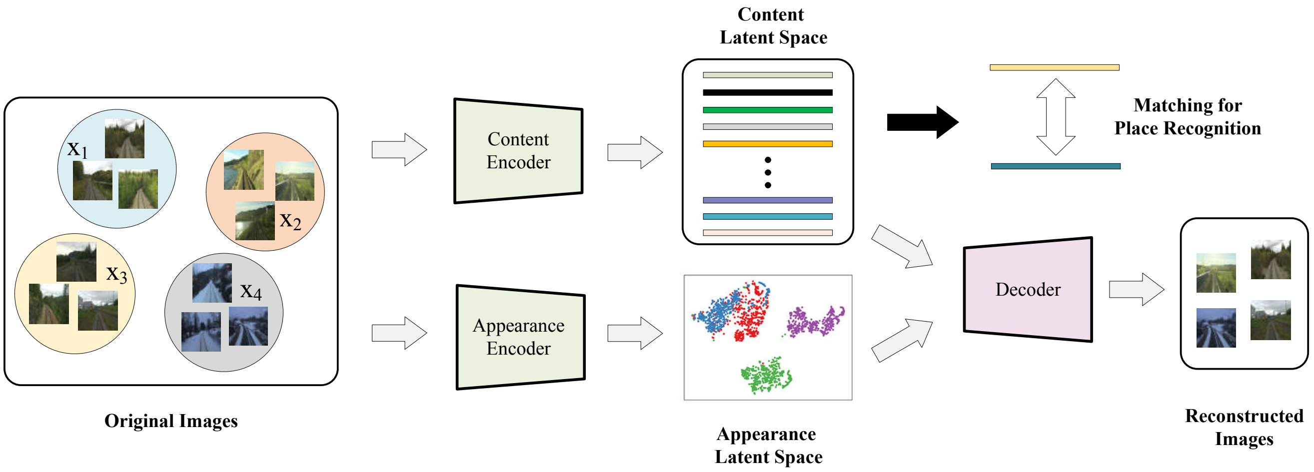


Fig. 1 The overall framework of our method. The original images are mapped to two independent feature spaces through the content encoder and appearance encoder. Besides, the features from two feature spaces should reconstruct the original image. The content feature will be used to measure the similarity of images.

While the convolution neural network (CNN) has shown a prominent effect in object classification and recognition [23], features extracted from CNN are used for judging whether images are similar or not. Some pre-trained models based on CNNs have proven to have better image recognition ability and robustness than traditional artificially designed image features. Even though they perform very well in the case of changes in appearance and perspective [50], satisfactory results cannot be achieved in a scene with severe appearance changes.

Extreme changes in appearance make it difficult to distinguish even images taken at the same location. One way to solve this problem is to learn how the appearance changes, and then generalize the learned factor to the original location to obtain new images. The query image is compared with the generated image to determine whether the reached position is similar. Recently, Generative Adversarial Networks (GANs) [18] has demonstrated its powerful ability to generate domain-specific images. Through the generative network, images can be converted from the source domain (spring) to the target domain (winter). In this way, even for scenes with extreme changes in appearance, images taken at the same location will become easier to be recognized after transformation. This idea of domain translation generally requires that images have a known one-to-one correspondence across domains, but labeling all correspondences is time-consuming and might not always be possible.

Changes in appearance can make images from the same place appear drastically different from each other, but they must have some commonalities such as the structure and layout of objects. Learning an interpretative representation of characteristics, with the ability to explore relationships between data across different domains has also attracted the attention of the researchers [27]. In order to understand and excavate the hidden common features among cross-domain data, cross-domain feature representation disentanglement aims to derive a latent feature space where the generated features can represent specific semantic information [6]. Once feature representation is successfully learned, the most distinguishing feature will be used to deal with various problems like visual classification or cross-domain image translation.

In view of the above consideration, we propose a Feature Disentanglement Network (FDNet) based on a convolutional auto-encoder and adversarial learning which can handle the place recognition problem of multi-domains within a unified framework. Based on the assumption that the image is composed of appearance and content factors, this approach removes the effect of appearance on image through adversarial learning. Thus the deep features that are not related to appearance changes for place recognition can be extracted. The latent feature space is learned from sets of images in each domain without requiring one-to-one image correspondences across the domains. Fig. 1 shows the overall framework of our method and the main contributions of our work are summarized as follows:

- We design a Feature Disentanglement Network (FDNet) based on a convolutional auto-encoder and adversarial training, which learns deep disentangled feature representation for place recognition.
- Our FDNet views appearance information and image content of interest as two latent factors to be disentangled, which handles the place recognition of multi-domains within a unified framework.

- We analyze the effect of length of the deep feature on the recognition performance, and achieve the measurement of similarity between images without any dimensionality reduction operations.
- A wide set of results comparing the proposed method against the main state-of-the-art algorithms in datasets with drastic appearance changes, while the disentangled feature representation is appearance-invariant and shows promising ability.

The remainder of this work is structured as follows. After reviewing the related work in the next section, we introduce in detail the proposed approach in Section 3. Our methods implementation and experimental results are presented in Section 4 and Section 5, while the last Section is devoted to the conclusion and future works.

2 Related Work

2.1 Appearance-changing Place Recognition

Visual place recognition has been a key part of the localization and mapping systems, and a lot of research works have been done in recent years [31]. There are two general methods to solve the appearance change in visual place recognition. One is to compute the visual characteristics that exhibit invariance properties to appearance; the other is to learn and predict appearance change.

Traditional visual features like SIFT and SURF are prone to be affected by the change appearance of the environment. Based on local keypoint features, Valgren et al. [53] used U-SURF features and achieved high recognition performance by comparing single-image pairs across different illumination conditions. A hybrid RatSLAM + FAB-MAP system was proposed in [17] for mapping in the difficult outdoor environment. This approach showed that it is practical to map in varying outdoor conditions visually. However, the authors also concluded that SURF features are sensitive to changes in illumination. Considering matching local sequences of images instead of matching a single location, SeqSLAM [38] was the first to achieve promising performance for localization across seasons and times of the day. Tayyab et al. [40] utilized the semi-dense image descriptors (HOG and AlexNet-based) and sequential information from network flows to improve the localization performance. Nevertheless, sequence-based methods only work with some assumptions such as similar velocity patterns and overlapping trajectories.

Since the potential of CNN over many computer vision tasks is excavated, a variety of methods have been proposed that address the vPR problem through CNN-derived description vectors. Carlevaris et al. [8] trained a convolutional multi-layer perceptron model to learn visual feature point descriptors that are robust to changes in scene lighting. In [50], feature maps were extracted from pre-trained models used for object recognition, which had proven to be effective in dealing with place recognition problems. Authors in [50] also concluded that the convolutional layer Conv3 performs better than all other layers under significant changes in appearance, and the higher fully-connected layer provides better viewpoint robust features. Roberto et al.[3] extracted information from different convolutional layers at different levels, and integrated them together to form CNN features. The feature compression techniques are applied to reduce the redundant data of CNN features to get the final representation. The research in [41] exploited the salient contents of the image and fused them with the convolutional features using feature aggregation to create a dense scene description. The learned discriminative image representation is able to improve the localization accuracy under challenging perceptual conditions. Our proposed approach is not limited to extracting image features from the middle layer of the network, but aims at providing feature representation with appearance-invariant characteristic through feature disentanglement.

The learning approaches use training data to find out how image features change with appearance, and to predict the image or its features after the appearance changes [42]. The authors in [34,36] transformed the images into illumination-invariant color space to significantly alleviate the negative effects of daily light and shadow. Nonetheless, it remains to prove that this transformation can be applied to other environmental changes, such as weather conditions. Lowry et al. [32] investigated how the overall appearance of the image changed with time and used linear regression to transform images from morning to afternoon. This transformation has been shown to improve the performance of visual localization compared with the matching between the original images. In [43], a superpixel vocabulary was built for each season and translates images across different seasons before matching. It demonstrates that SeqSLAM [38] and BRIEF-Gist [49] can benefit from this operation greatly. However, this method requires one-to-one correspondence of images in different seasons for training. Yasir et al. [25] took advantage of the popular GAN to generate the appearance of a place given the current environmental conditions. The features extracted from the first fully-connected layer are used for place recognition under the different weather conditions. Although it does not need to use paired correspondence across seasons, this system implements image conversion between only two different domains.

2.2 Adversarial Learning

Recent work [18] has shown that adversarial training contributes to improving the performance of many computer vision tasks such as image generation [52], image super–resolution [26] and style transfer [56]. A typical GAN network is composed of generator G and discriminator D. G captures the mathematical distribution model of real data and generates new samples from the learned distribution model. The generator tries to make the generated image unable to be distinguished between true or false in the discriminator. D is a classifier used to determine whether the input is real data or generated samples. They compete to outperform each other constantly to improve their generating and discriminating abilities and achieve a balance. With this adversarial training, the generator can learn a mapping method to project the hidden space to the image domain we want.

WGAN [2] improved GAN from the point of view of the loss function. It used Wassertein distance to measure the distance between generating data distribution and real data distribution instead of Jensen-Shannon divergence, thus alleviating the training instability of GAN. Subsequently, WGAN-GP [19] proposed a method to replace weight clipping in the WGAN discriminator, which used a gradient penalty to solve the problem of gradient disappearance or explosion in training. This method has a faster convergence rate than standard WGAN and can be widely used in a variety of GAN frameworks.

There are a lot of works on GAN and different applications, but we are more concerned about using this type of network for domain adaptation or domain transfer which is closely related to feature disentanglement. Ganin et al. [16] obtained domain invariant features by optimizing two discriminative classifiers at the same time, where the gradient reversal algorithm is used to realize adversarial losses. The Bidirectional Generative Adversarial Networks (BiGAN) [14] is an extension of the GAN which learns the inverse mapping from the image data back into the latent space in an unsupervised way. It was indicated that the learned feature representation is useful for image classification tasks. CoGAN [28] applied GAN to domain adaptation and image transformation by training a tuple of GANs for each image domain. The weight-sharing constraint in the high-level layer was used to generate a domain-invariant feature space. Markus et al. [54] improved the performance of free-space segmentation under varying appearance by applying adversarial domain adaptation techniques. They also proposed an approach IADA [55] to solve the domain adaptation problem of lifelong, continuously changing appearance.

Inspired by the adversarial learning method to solve the problem of domain adaptation and domain transfer, we consider transforming the place recognition problem in the case of extreme changes in appearance into multi-domain adaptation problem, and use adversarial training to map the images into the generated common space, so as to extract features that are insensitive to changes in appearance.

2.3 Representation Disentanglement

Disentangling hidden factors from images has enabled a deeper understanding of images [35,44]. The work in [20] is among the first to utilize an encoder-decoder structure for representation learning, whereas it is not explicitly disentangled. Kenshimov et al. [22] considered individual feature maps as the smallest indivisible units of analysis, and evaluated the performance to omit the activation maps that are significantly varied as the environment changes. Although this method can improve cross-seasonal place recognition, the feature extracted from a mid leveled CNN layer has low efficiency in matching.

With the recent development of generative models like generative adversarial networks (GANs) and variational autoencoders (VAEs), some researches on feature disentangling attempt to learn an interpretable representation from large amounts of data through deep neural networks (DNN). Odena et al. [45] realized feature disentangling based on the auxiliary classifier GAN (AC-GAN) proposed by them. Given attribute information in the training process, the model can automatically generate images to be conditioned on the desirable latent factors. In [9] InfoGAN was proposed to learn disentangled representations through unsupervised learning. The mutual information between pre-specified latent factors and the synthesized images are maximized. However, the semantic meaning of the feature in the latent space cannot be explicitly explained. Fader Networks [24] proposed a new method to learn attribute-invariant latent representations and generate variations of images by sliding attributes. The values of attributes and the salient information of the image are disentangled through an encoder-decoder architecture. A framework of Cross-Domain Representation Disentangler (C-DRD) was proposed in [29] to solve the problem of ground truth annotation of training data in the feature disentangling process. It was demonstrated that the domain adaptation and cross-domain feature disentanglement can be simultaneously executed for solving classification tasks of unsupervised domain adaptation. A Multimodal Unsupervised Image-to-image Translation (MUNIT) framework [21] was presented to solve the problem of unsupervised Image-to-Image transformations. The author assumed that image representation can be decomposed into a domain-invariant content code and a style code that can characterize domain-specific properties. The final image translation is generated by reorganizing the content code of the original image with a style code randomly extracted from the target domain.

The above methods have promising performance in feature disentangling and image generation. Motivated by them, we consider that the image is decomposed into two different feature spaces, content space and appearance space by an encoder-decoder architecture at extreme changing scenes. Instead of generating or predicting the changed image, we directly use features in latent content as image features for place recognition. In our setting, we have several domains that share the same content distribution but have different appearance distribution.

3 Proposed Approach

In this section, we will introduce the architecture of the proposed method in detail, which integrates convolutional auto-encoder and adversarial training to generate common feature space. The model maps a high-dimensional original image to a low-dimensional feature space with the propriety of high compression and invariance to appearances. The network structure is trained by unsupervised learning which does not need too many labels, so the method is efficient and feasible.

3.1 Motivation and Pipeline

A widely used method to deal with the problem of visual place recognition is to find an appropriate feature space for images. In this feature space, feature vectors have characteristics: they are not affected by changes in appearance and viewpoints, and the distance between feature vectors can measure the similarity between images. In other words, the greater the distance between feature vectors, the less similar structure or context the original images have. Once such a feature space is found, the place recognition problem can be transformed into the problem of measuring the difference between feature vectors. In this paper, we focus on how to deal with extreme changes in environmental appearance. The images captured at the same place at different times or under different weather conditions are quite different. As a result, we treat the appearance changes as multiple domains and map images from different domains to the pre-defined feature space by means of feature disentangling. These appearance changes can also be viewed as being modeled into discrete classes and classified by a discriminator. Based on the above considerations, we try to find such a feature representation through adversarial learning and propose a unified network architecture which can derive appearance-invariant feature from images across multiple domains (appearances). To be mentioned, our architecture is limited to coping with scenes with discrete changes in appearance such as spring to winter and day to night.

We first assume that the latent space of images can be decomposed into an appearance space and a content space. The content vector encodes the information that should be preserved during the appearance change, which is what we desire for place recognition. Given image sets $\{X_c\}_{c=1}^{N}$ across N domains (such as different seasons), the proposed method learns a domain-invariant representation z for the input image $x_c \in X_c$ (in each domain c). Fig. 2 shows an overview of the model and its learning process. The network consists of a content encoder E_c, an appearance encoder E_a, a decoder D_e and an appearance discriminator D_a. Take domain X as an example, the content encoder E_c maps images onto a shared, domain-invariant content space ($E_c : X \to C$) and the appearance encoder E_a maps images onto a domain-specific attribute space ($E_a : X \to A$). The decoder D_e restores images by accepting the feature vector from the two encoders ($D_e : \{C, A\} \to X$). It is worth mentioning that we impose constraints on the appearance encoder to ensure that the appearance features do not contain additional content information. Triplet loss is used so that the appearance features generated by images belonging to the same domain are closer to each other, while the appearance features of different domains are far from each other. The appearance discriminator D_a aims to distinguish whether the extracted content representations are from the same domain or not.

3.2 Description of the Loss Function

3.2.1 Auto-encoder loss

As shown in the middle of Fig . 2, image x_c is entered into the two encoders E_c and E_a to obtain a content vector v_c and an appearance vector v_a:

$$v_c = E_c(x_c), v_a = E_a(x_c) \quad (1)$$

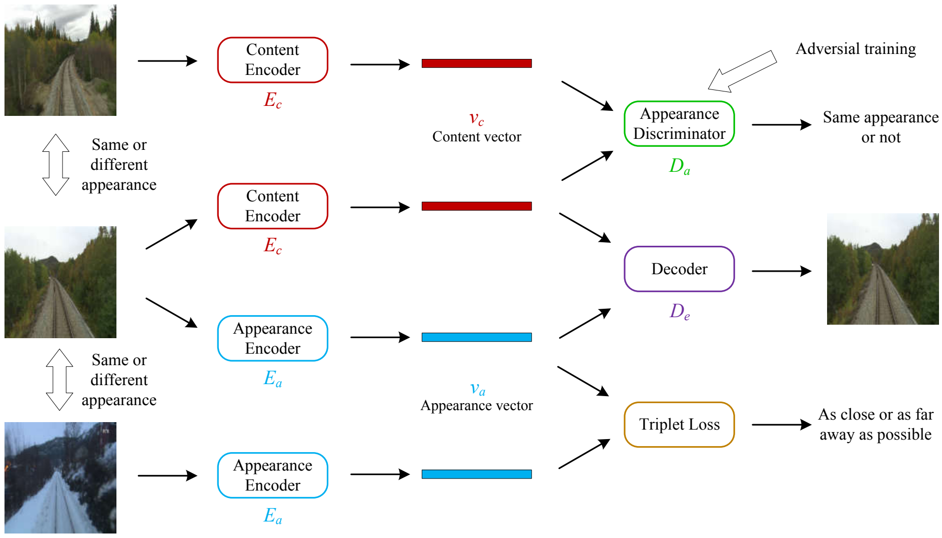


Fig. 2 Overview of our model and the learning process. D_a tries to tell if two content vectors come from the same domain. The purpose of the encoder E_c is to trick the appearance discriminator D_a so that it can not classify appearance features correctly. Triplets loss is used to make the appearance feature to encode the appearance information.

Then v_c and v_a are fed to the decoder D_e to reconstruct the original image x_c. Thus we get the reconstructed output:

$$\tilde{x_c} = D_e(v_c, v_a) \tag{2}$$

The mean squared error (MSE) is minimized in the training procedure. So the reconstruction loss L_r is given as:

$$L_r(\theta_c, \theta_a, \theta_{dec}) = \sum_{x_c \in X_c} \|x_c - \tilde{x_c}\|_2^2 \tag{3}$$

where $\theta_c, \theta_a, \theta_{dec}$ are the parameters of the encoders and the decoder respectively.

3.2.2 Appearance encoder loss

The proposed method embeds input images onto a shared content space C, and domain-specific space A. Intuitively, the content encoders should encode the common information that is shared between domains onto C, while the appearance encoder should map the remaining appearance information onto A.

Let's take two domains for example. Let $x_1 \in X_1$ and $x_2 \in X_2$ be images from two different image domains. x_1 and x_2 obtain the feature vectors in the feature space C and A respectively through the same encoder. However, sharing the same mapping functions cannot guarantee the representations in the latent space encode the same information for both domains. So we impose additional constraints on the encoder during the training process to obtain two disjoint feature spaces. First, we want the appearance encoder to be able to capture the appearance information in the image. For instance, when season changes we want the feature in this space to contain only seasonal information but not the structure or content information in the image. Therefore for appearance encoding of the same domain the distance between them should be closer, while for appearance encoding of different domains, the distance between them should be further and thus greater than a certain threshold. As shown in the lower part of Fig. 2, we train the network through a triplet embedding scheme, where the appearance encoder is used to produce three vectors v_{ai}^a,v_{ai}^p,v_{ai}^n. They are from three input images and form the positive pair $\{v_{ai}^a, v_{ai}^p\}$ and the negative pair $\{v_{ai}^a, v_{ai}^n\}$. Thus we want:

$$\|v_{ai}^a - v_{ai}^p\|_2^2 + \alpha < \|v_{ai}^a - v_{ai}^n\|_2^2 \tag{4}$$

where α is a margin that is enforced between positive and negative pairs. E_a is learned to minimize the following triplet loss function [48]:

$$L_a(\theta_a) = \sum_i^K max(\|v_{ai}^a - v_{ai}^p\|_2^2 - \|v_{ai}^a - v_{ai}^n\|_2^2 + \alpha, 0) \tag{5}$$

which is zero when the distance of the negative pair is larger than the distance of the positive pair by at least a margin α. Triplets not satisfying this condition will produce non-zero costs that the training process will attempt to reduce by updating the weights of the CNN accordingly through stochastic gradient descent. θ_a is the parameter of the appearance encoder. K is the number of all triplets in the training set.

3.2.3 Adversarial loss

The auto-encoder itself with equation (3) cannot make the latent representation E_c appearance-independent. The appearance information of the original image x_c existing in v_c inevitably degrades the final performance. This is why we train an extra appearance discriminator in order to regularize the encoder E_c to make v_c appearance-independent. As shown in the upper right corner of Fig. 2, appearance discriminator D_a takes two content vectors v_{ci} and v_{cj} as inputs and tries to determine if the two vectors come from the same domain. The purpose of the encoder E_c is to trick the appearance discriminator D_a so that it does not classify appearance features correctly.

D_a is treated as a binary classifier. For each training pair $\{x_i, x_j\}$ with its ground truth label y, when $y = 1$ x_i and x_j are from the same domain and when $y = 0$ they are from different domains. The classification loss can be defined as the cross-entropy between predicted class distribution $D_a(v_{ci}, v_{cj})$ and the label y:

$$
\begin{aligned}
L_d^{adv}(\theta_d) = -\sum_{(x_i,x_j)\in D} & y * log(D_a(E_c(x_i), E_c(x_j))) \\
&+ (1-y) * log(1 - D_a(E_c(x_i), E_c(x_j))) \quad (6)
\end{aligned}
$$

where $y \in \{0, 1\}$, and θ_d is the parameter of the discriminator, which can also be represented as:

$$L_d^{adv}(\theta_d) = \mathbb{E}_v[logD_a(v)] \quad (7)$$

where v is the concatenation of content vectors $E_c(x_i)$ and $E_c(x_j)$. The discriminator D_a is trained to minimize $L_d^{adv}(\theta_d)$ in equation (3). In contrast, the encoder E_c is trained to maximize $L_d^{adv}(\theta_d)$ in order to remove the information of appearance in v_c. As a result, the objective of the encoder E_c is derived as follows:

$$L_e^{adv}(\theta_c) = -L_d^{adv}(\theta_d) = -\mathbb{E}_v[logD_a(v)] \quad (8)$$

In this way, only the content information is learned in v_c, while only the appearance characteristics are encoded in the appearance vector v_a. However, as mentioned in WGAN [2], cross-entropy is not a stable loss function during adversarial training if there is a large gap between the predicted distribution and the real distribution. With the loss in equation (7), optimization becomes even more unstable due to the volatile gradient. To stabilize the training process, we replace equation (7) with Wasserstein GAN objective with gradient penalty [19] defined as:

$$L_d^{adv}(\theta_d) = \mathbb{E}_v[logD_a(v)] + \lambda_{gp}\mathbb{E}_{\hat{v}}[(\|\nabla_{\hat{v}} D_a(\hat{v})\|_2 - 1)^2] \quad (9)$$

$\hat{v}$ is sampled uniformly along the straight lines connecting pairs of training data (v_i, v_j), where v_i and v_j have different labels. λ_{gp} is a weighting parameter.

To train the whole network, we alternatively update the encoder, decoder, and discriminator with the following gradients:

$$
\begin{aligned}
\theta_c, \theta_a, \theta_{dec} &\xleftarrow{+} -\Delta_{\theta_c,\theta_a,\theta_{dec}}(L_r + L_a + L_e^{adv}) \\
\theta_d &\xleftarrow{+} -\Delta_{\theta_d}(L_d^{adv})
\end{aligned} \quad (10)
$$

It is worth noting that θ_c, θ_a and θ_{dec} are jointly updated in each iteration. θ_d is updated separately. Finally, the pseudo-code for training the method is summarized in Algorithm 1. Implementation details of our network architectures will be presented in Section 4.

Algorithm 1 Learning of FDNet

Input: batch_size B , domain_num N_d , A set of training images X
Output: parameters: $\theta_c, \theta_a, \theta_e, \theta_d$

```
1: θc, θa, θe, θd ← initialize;
2: for Iters. of whole model do
3:     X_b ← Sample mini-batch from X_s
4:     T ← generate triplets according to Algorithm 2
5:     P ← generate pairs with its label by sampling from X_b
6:     for Iters. of updating auto-decoder do
7:         θc, θa, θdec ←+ −Δ_{θc,θa,θdec}(L_r + L_a + L_e^adv)
8:     end for
9:     for Iters. of updating discriminator do
10:        θd ←+ −Δ_{θd}(L_d^adv)
11:    end for
12: end for
13: return θc, θa, θe, θd
```

4 Implementation

4.1 Network Architecture

Fig. 3 displays the network architecture of the encoder, decoder, and the discriminator. Before training begins, every image in the set of training images is resized to 224×224 and used to create image pairs (see Algorithm 2). The salmon-colored blocks represent input and output images. The numbers below the block represent the shape of feature maps output by the block. The content encoder and appearance have the same structure as shown on the left side of Fig. 3. Each encoder contains several encoding blocks and a fully-connected layer. Each encoding block consists of a convolution layer (filter size 5, stride 2), followed by batch normalization and a Leaky Rectified Linear Unit (slope 0.2). L is the length of the output vector from the encoder. It's worth mentioning that the parameters of the two encoders are not shared, in order to ensure that the

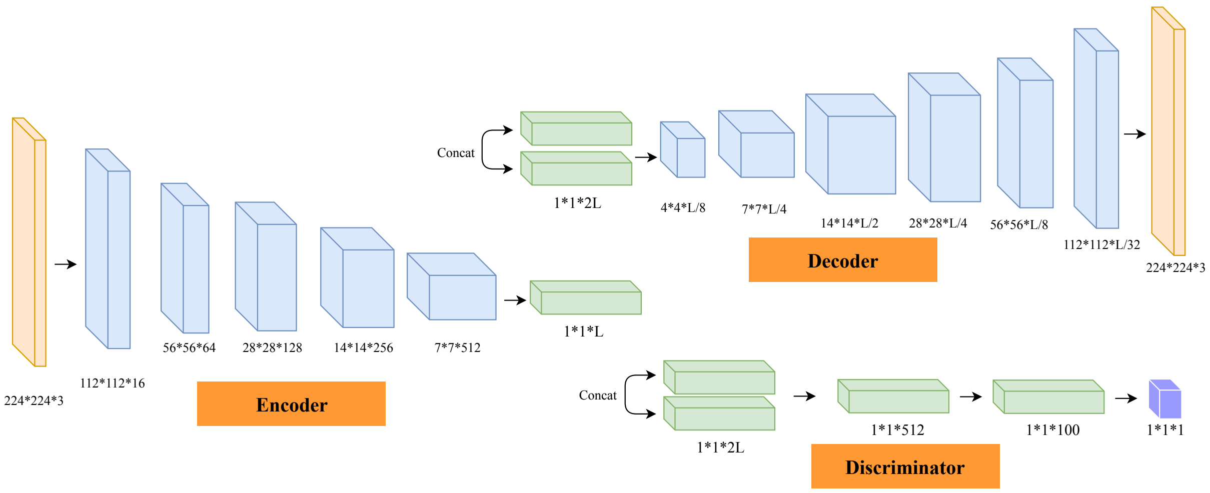


Fig. 3 Network architectures of of the encoder, decoder and the discriminator.

appearance and content features have different distributions in the feature space. The decoder accepts vectors from two encoders and concatenates them to reconstruct the original image. The decoder contains several upsample blocks. Each upsample block consists of a deconvolution layer, batch normalization and a Leaky Rectified Linear Unit. The discriminator accepts two vectors from the content encoder. It has three fully-connected layers which are mapped to a single output for classification. In the experimental section, we show that the content encoder learns appearance-independent features that can be used for place recognition.

Dropout: It is useful to add dropout to improve the robustness of the model. The dropout rate is set as 0.5 in the encoder, and 0.25 in the classifier.

Hyper-parameters: The batch size is 4, and all weights are initialized from the zero-centered normal distribution with a standard deviation of 0.02. An Adam optimizer is used with a learning rate of 0.0001 and momentum 0.5. λ_{gp} is set to 10 and the margin α in triplet loss is set to 0.1.

Training details: We first pretrained the encoder and decoder for 5000 mini-batches, then pretrained the discriminator for 8000 mini-batches. Finally, we trained the encoder/decoder for 1 iteration and 2 iterations for the discriminator. The joint stage was trained for 60000 mini-batches in total.

4.2 Feature Embedding and Matching

The disentanglement of image features is completed when the whole network is trained. The output of E_c is a vector that provides a representation of the original image which is useful to accurately discriminate images under changing conditions. The evaluation is performed by single-image nearest neighbor search based on the cosine distance of the extracted feature vectors. However, computing the cosine distance between high-dimension vectors is an expensive operation. For example, the convolutional feature in Conv3 [50] used in the matching process will lead to high computational load. Although Locality Sensitive Hashing (LSH) is used to reduce the dimension of the feature vectors to improve the efficiency, such dimensionality reduction depresses the performance of place recognition. In our method, since we directly output the required feature vectors through the fully-connected layer of the encoder, we can obtain the vector of different lengths by modifying the structure of the network when considering the feature dimension. Thus, the problem of feature dimension is ignored during the process of network construction. In view of this, we make performance comparison of vectors with different lengths in Section 5.2.1.

In this way, the final feature vector $\hat{F}$ can be obtained. The query feature $\hat{F}_q$ of the query location l_q and the database feature vector $\hat{F}_{db}$ are compared using the cosine distance as in equation (11)

$$s(\hat{F}_q, \hat{F}_{db}) = \frac{\hat{F}_q \cdot \hat{F}_{db}}{\|\hat{F}_q\| \|\hat{F}_{db}\|} \tag{11}$$

The location l_s with the minimum distance to the query location l_q is regarded as a true positive match if it is from the same location as l_q (within dataset tolerances– see Table 1 for a summary of tolerances).

4.3 Hard Triplets Selection

To learn the desired feature vector produced by the appearance encoder, triplets must be chosen to provide

Table 1 Tolerances for true positives matches

Dataset	Location Tolerances
Nordland	5 frames
Alderley	2 frames
Oxford RobotCar	30 meters
St Lucia	30 meters
FAS	3 frames

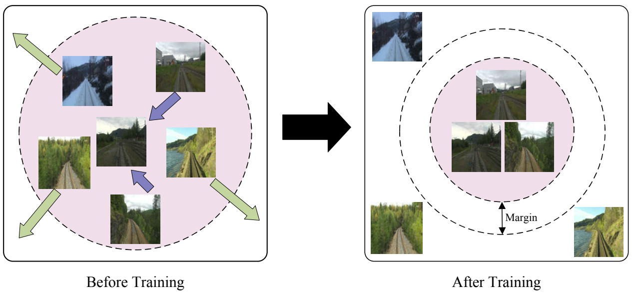


Fig. 4 Schematic illustration of samples before training (left) versus after training (right) by minimizing the triplet loss.

relevant visual cues. As seen in Fig. 4, the distance between an anchor and a positive is minimized so that samples that have the same identity will be aggregated. The distance between the anchor and a negative will be maximized to maintain at least the distance between dissimilar samples.

We adopt the method of hard triplets selection. Assuming that the training set contains images of N_d different domains. For each mini-batch with the shape (B, N_d), the features corresponding to all data are obtained by the appearance encoders and the distance between features are calculated and stored in the matrix. Then we need to find the positive sample with the maximum distance and the negative sample with the minimum distance for each anchor. In this way, the hardest triplet for every anchor is obtained. Finally, a total of $B * N_d$ triplets for a mini-batch can be generated. The pseudo of the calculation is listed in Algorithm 2.

5 Experiments

In this section, we conduct several experiments to demonstrate the performance of the proposed method. We firstly introduce the setup of the experiment, including the datasets, the sequences, and the evaluation methodology. Then, we provide details of experiments compared with other approaches and give quantitative and qualitative results in terms of the place recognition accuracy.

Algorithm 2 Generating Triplets

Input: batch_size B , domain_num N_d , A set of training images X
Output: triplets T

```
T ← initialize;
X_b ← Sample mini-batch from X in shape (B, N_d)
V_b ← get embeddings from X_b
M_b ← calculate pair_distance for each embedding of V_b
(A_v, P_v)(A_v, N_v) ← get all valid positive pairs and negative pairs
for a in X_b do
    (a, p) ← find elements with the maximum distance in (A_v, P_v) according to M_b
    (a, n) ← find elements with the maximum distance in (A_v, N_v) according to M_b
    T ← T.append(a, p, n)
end for
return T
```

5.1 Experimental setup

5.1.1 Datasets

In order to evaluate our approach, datasets are required to traverse the path in different environments but without too much view-point change. Moreover, ground truth information, such as the corresponding scenes should be contained in the datasets.

Nordland: The Nordland dataset is one of the most challenging place recognition datasets due to the changing landscape and weather, as Fig.5 (a) illustrates. It includes four simultaneous video streams of different seasons. Each 9-hour video corresponds to a season, and they were manually aligned so that frames with the same numeral are from the same location. In addition to the extreme changes in appearance produced by the season, these images also include extreme blurring because of the train's excessive speed. We extract the image from video at a rate of a frame per second removing all frames where the train was in a tunnel or stationary. Then the sequence is divided into two parts, one for training including 27000 images, and the other for testing including 1000 images.

Alderley: The Alderley dataset was first introduced in SeqSLAM [38]. It consists of two videos, one on rainy nights and the other on sunny days. Fig.5(b) shows an example of images that contains severe changes in illumination and weather conditions in a given location. These two pictures are difficult to identify the same place even for humans. Frame correspondences are provided in the dataset for place recognition as ground-truth. We used the first 1000 frames of the sequence for the test set, and the rest for the training of the network.

Oxford RobotCar: The Oxford RobotCar Dataset [33] consists of over 100 repetitions of car traverses through Oxford, UK, recorded over a year across different times of day. We extract images at 5 frames per second from the route, which corresponds to approximately three kilometers through Oxford. The videos were recorded on a sunny day (2014-12-16-09-14-09) and a night day (2014-12-10-18-10-50). The training set includes 1758 images and the remaining 754 images are used for testing. We use a ground truth tolerance of 30 meters.

St Lucia: The St Lucia dataset [17] contains several car traverses through the suburb of St Lucia, Queensland. The videos are captured with a forward-facing camera placed on the roof of a car across five different times of day. We train the network and test on the early morning sequence (time:190809_0845) and the late afternoon sequence (time:180809_1545) which contains significant appearance changing. We use the provided GPS information and set ground-truth tolerance to 30 meters. The images are extracted from the 15 FPS videos. The first 3500 images are used for training and the next 500 are for evaluation.

FAS: The Freiburg Across Seasons dataset (FAS) [40] was recorded by a camera-equipped car in Freiburg city, Germany, across different seasons including summer and winter. The ground truth was provided for all the localization sequences with reference to the Mapping sequence. We use the Localization-2 sequence and the Mapping sequence for training and testing, which contain 3130 image pairs and 1347 image pairs, respectively. The ground truth tolerance is set to 3 frames.

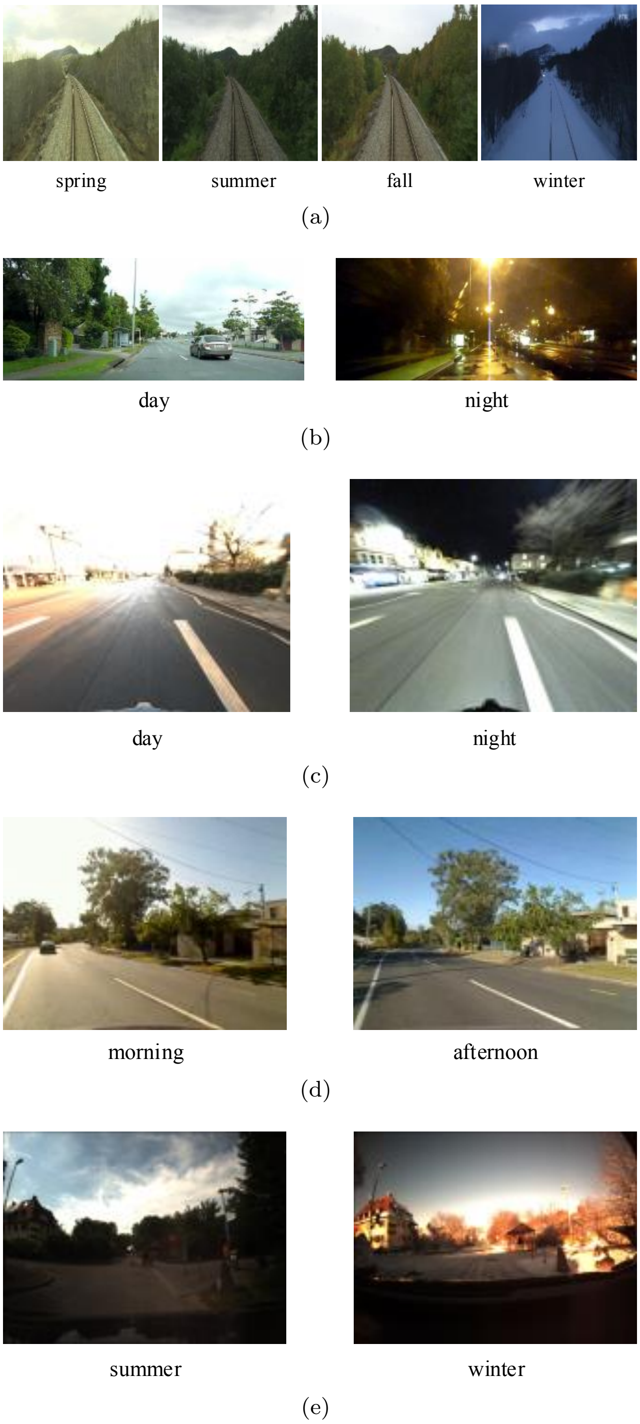


Fig. 5 Randomly selected sample images from the dataset. (a) Images in Nordland Dataset. (b) Images in Alderley Dataset. (c) Images in Oxford RobotCar Dataset. (d) Images in St Lucia Dataset. (e) Images in FAS Dataset.

5.1.2 *Evaluation Methodology*

To evaluate the performance of the proposed method, we compared it with several different state-of-the-art approaches such as:

(a) **Gist**: A holistic representation of images which can retain the context information.
(b) **DBoW**: We use the DBoW [15] vocabulary tree applied in ORB-SLAM [39].
(c) **Conv3**: The conv3 feature discussed in [50] is used in this paper to carry out the experiment. The original conv3 feature from AlexNet is a vector of 64896 dimensions, which makes the matching inefficient. We use the Gaussian random projection (GRP) [7] to compress the conv3 feature to the same dimension as our method, because GRP is more efficient in dimensionality reduction than LSH in the practical test.
(d) **Landmarks**: The method proposed by Zetao et al.[11] extracts several different salient regions to express the global features of images while requiring no labeled data for training.
(e) **Conv4_fine-tuned**: The conv4 features extracted from the HybridNet [10] which is fine-tuned and trained specifically for place recognition.
(f) **NetVLAD**: It achieved weakly supervised training for place recognition using a CNN architecture

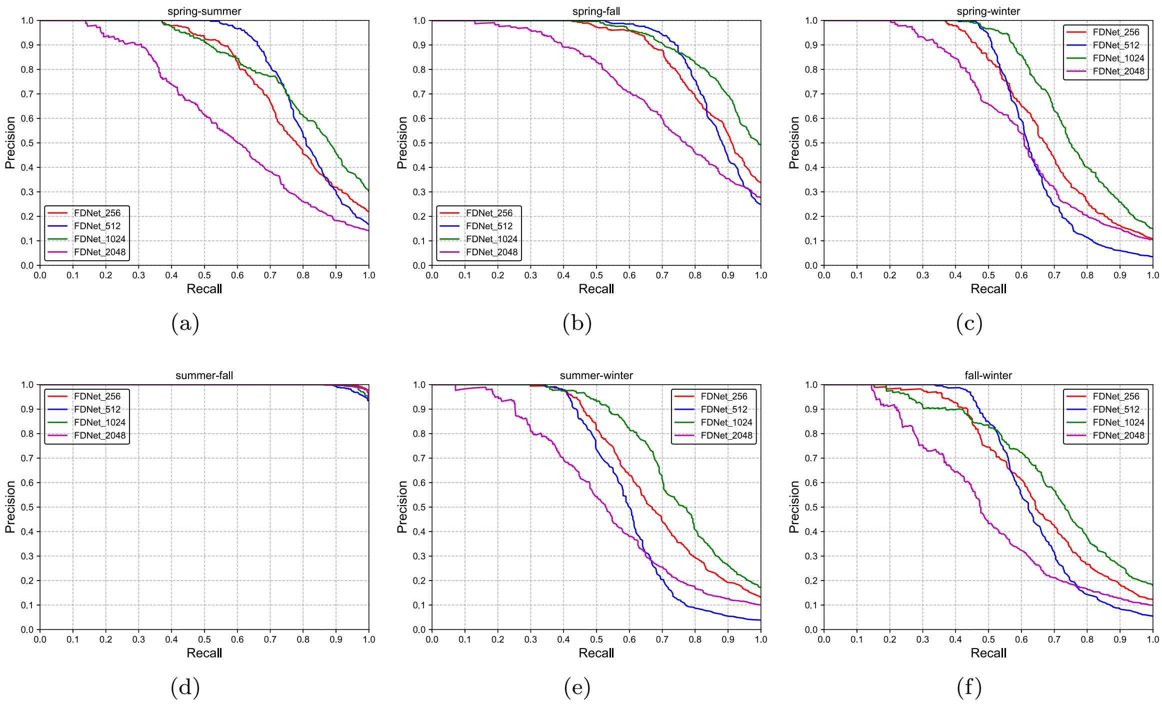


Fig. 6 Precision-recall curves comparing different lengths of the vector with our method on Nordland Dataset. (a) spring versus summer. (b) spring versus fall. (c) spring versus winter. (d) summer versus fall. (e) summer versus winter. (f) fall versus winter

that embeds a traditional VLAD layer. We have employed the Pytorch implementation of NetVLAD [1] with the hardest triplet loss.

(g) **CALC**: An unsupervised deep neural network [37] for fast and robust loop closure. The authors utilized the auto-encoder to reconstruct the HOG descriptor of original input images. The open-source implementation is utilized in our experiments.

The designed methodology for testing performance is principally based on precision-recall curves, which are calculated from the similarity matrix obtained in each test set. A threshold is set and used in the matching process between the similarity matrix and ground-truth matrix. In this way, the occurrence times of TP (True Positive), TN (True Negative), FP (False Positive) and FN (False Negative) on the dataset are obtained. The values of precision (P) and recall (R) are calculated as follows:

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN} \tag{12}$$

The final precision-recall curve is obtained by varying the threshold value θ in a uniform distribution between 0 and 1. In our tests, 500 values of θ are taken in order to obtain well-defined curves.

Maximum recall at 100% precision: The proportion of correct matches that can be achieved with no false positives. 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 and a higher value is desired.

5.2 Results

5.2.1 Vector Length

As mentioned in the previous section, vector length makes a difference in the performance of place recognition and the efficiency of matching. In our experiment, the length of feature vectors extracted by the content encoder can be adjusted by constructing different fully-connected layers. On the premise of high efficiency, we need to find the most appropriate length of the feature vector. We tested the performance of different vector lengths as shown in Fig. 6. It can be seen that the length of 2048 performs worse than other lengths in six test experiments except summer-fall comparison and the other three lengths have similar results. However, according to the principle of selecting a higher value on maximum recall at 100% precision, it is not difficult to see that the feature vector of 512 performs better.

Table 2 summarizes the required time for the feature extraction and feature matching between reference

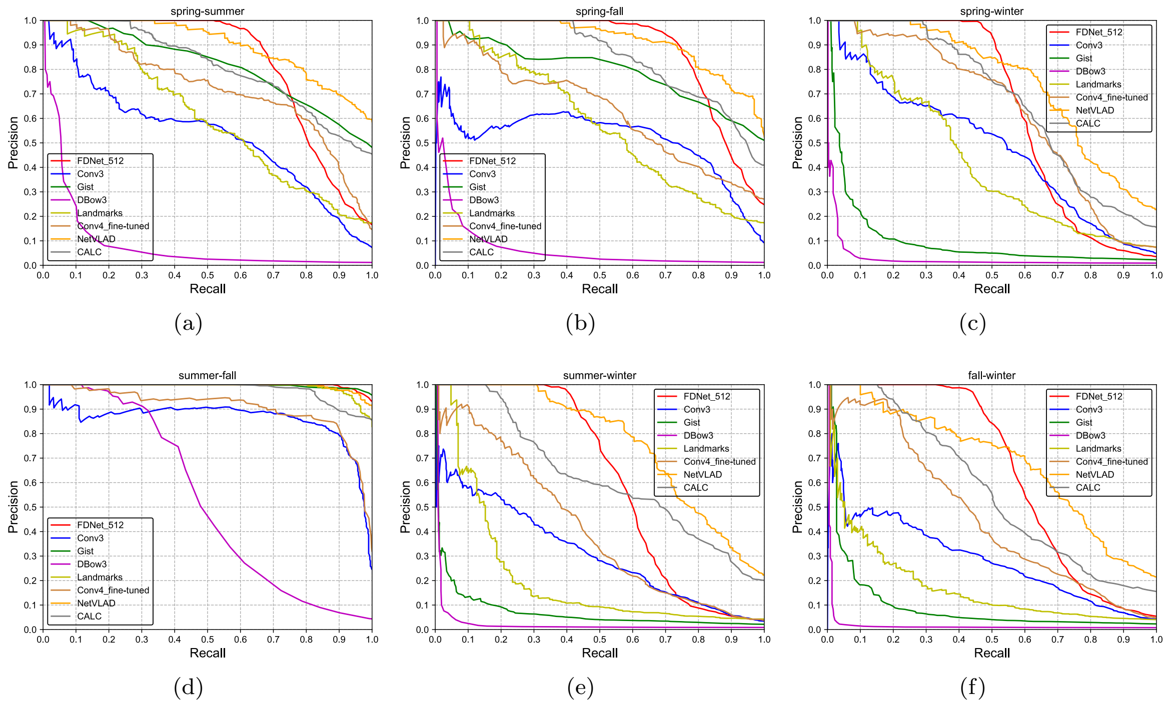


Fig. 7 Precision-recall curves comparing the different approaches with our novel method on the Nordland dataset. (a) spring versus summer. (b) spring versus fall. (c) spring versus winter. (d) summer versus fall. (e) summer versus winter. (f) fall versus winter

Table 2 Runtime comparison between different lengths of the vector with our method on the Nordland Dataset.

	Feature Extraction (ms)	Feature Matching (ms)
FDNet_256	20.9	0.32
FDNet_512	21.2	0.35
FDNet_1024	22.6	0.36
FDNet_2048	28.7	0.40

and a single query image. We tested 2000 images from the Nordland Dataset and obtained the average value. There was no significant difference in time consumption of feature matching under different feature lengths. The time of feature extraction are all less than 30ms, and the time of feature matching are within 0.4ms. Considering the performance and time consumption of different feature lengths, we finally choose the vector of length 512 for the subsequent experiments.

5.2.2 Results on Nordland Dataset

Firstly, we show the precision–recall curves on the Nordland dataset as displayed in Fig. 7. In order to show the robustness of the method to appearance changes, we cross-compare the data of four different seasons and generate a Precision-Recall(PR) curve. Table 3 shows the precision and recall values obtained at maximum recall and precision respectively. It is observed that our method has a significantly higher performance in the majority of cases. Even when the weather changes from spring, summer or fall to winter, FDNet can maintain a higher value on maximum recall at 100% precision. The main reason for the improvement is that the content features contain little appearance information and are therefore able to cope with changes in appearance. A good example is the second column of images in Fig. 9. Since there are obvious seasonal changes between the query image and the dataset image, the appearance characteristics are no longer preserved. Except for FDNet and CALC, all other methods match the wrong image for this query.

However, we can also see that the accuracy of FDNet declines more rapidly in the high recall area. This is because the highly compressed feature vectors inevitably lose part of the image information, resulting in the difference among most image features is not so obvious. Generally speaking, the proposed method tends to localize more precisely than other state-of-the-art approaches, providing better resistance to the changing of appearance.

On this dataset, NetVLAD is comparable to FDNet for that it gets the closest recall value to our method. CALC shows moderate performance and the fine-tuned conv4 feature improved greatly compared with the orig-

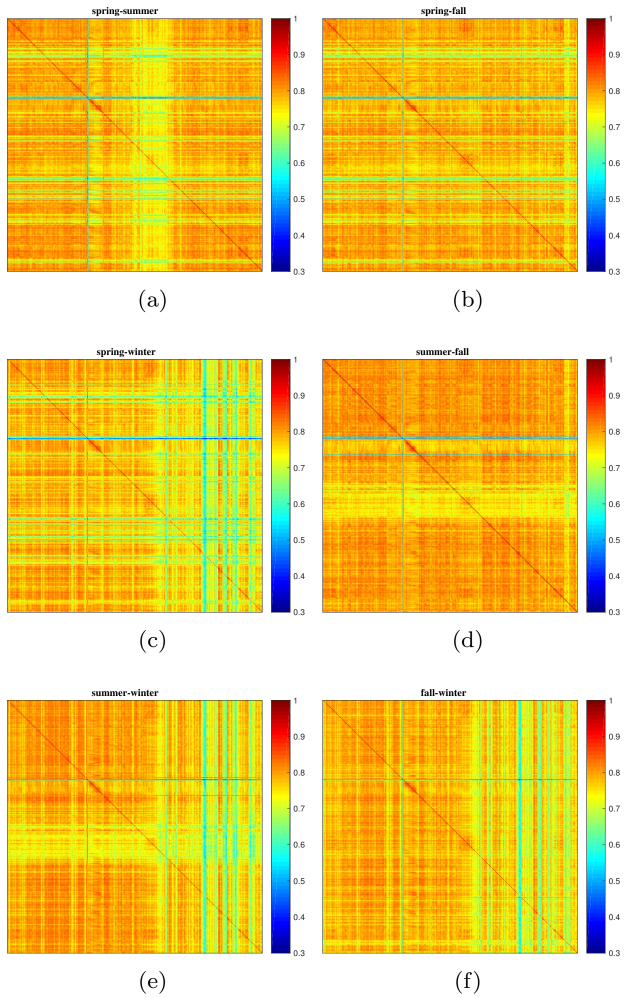


Fig. 8 The similarity matrices belonging to our approach in the test sequence of Nordland dataset.

inal conv3 feature. Furthermore, DBow3 performs the worst in most cases because of the limitations of hand-crafted features.

Since we know that image sequences on different seasons are synchronized, the ground truth similarity matrix is a diagonal matrix. Fig. 8 depicts the similarity matrices obtained with the test sequence on the Nordland dataset. We also cross-compare the data of four different seasons. It can be seen that there is a significant difference among the elements on the diagonal line and those on the non-diagonal line. However, the difference between the non-diagonal elements is not so great as to be difficult to distinguish. This is because the Nordland dataset captures the railway scene, and most of the images are very similar in content.

In order to evaluate the discriminant ability of the content vector from a quantitative perspective, we presented the changes in the content vector under different appearances of the same scene in the form of a histogram. Fig.11 (a) displays the absolute difference of content vectors extracted from location $T1$ in Fig. 10 across different seasons. It can be seen that the value of the absolute difference is small and below 0.05 even if the appearance changes. Fig. 11(b) is the absolute difference generated by the location $T1$ and $T2$. When location changes, the absolute difference increases significantly and is higher than the result in (a). This demonstrates that the content vectors generated by feature disentanglement have the ability to perceive image content when appearances change.

To quantitatively analyze the invariance of the appearance vector, we display the response of appearance vectors belonging to images from Fig. 10. As shown in Fig.12, it is obvious that the appearance vectors extracted from images under the same season only change slightly even if location changes, which indicates that the appearance vectors extracted can accurately encode the appearance information of images. Additionally, we find that there is a significant difference between appearance features from winter and appearance features extracted from other seasons, while the appearance features extracted from spring, summer, and fall show a smaller difference. This is caused by the obvious disparities between winter images and other seasonal images.

We visualize the distribution of appearance features mapped to two-dimensional space subsequently. As shown in Fig. 13(a), points belonging to the same class are easier to gather together, and the distribution of points under winter has obvious distance from the other seasons. Although the feature points of spring, summer and fall are close to each other, it is not difficult to distinguish them.

5.2.3 Results on Alderley Dataset

Apart from the typical seasonal changes previously studied, we also perform evaluations under extremely variable illumination conditions. We conduct experiments on Alderley Dataset. This dataset contains image sequences in both day and night scenarios, and the changes between images at the same location are more significant. Table 4 shows the precision and recall values obtained at maximum recall and precision respectively. On the whole, all the methods performed poorly on this dataset. The third query (column) in Fig. 9 is an example that all the methods fail to find the correct match. The PR curves plotted in Fig. 14 show an acceptable accuracy for our method in this challenging case, and we can see that the proposed method (10.82%) performs second only to NetVLAD (11.54%) and maintains higher accuracy in the region of low recall. In addition, we also draw the point distribution mapped by appear-

Fig. 9 Samples of matched/mismatched images by different methods. Each column represents a query and matched images of various methods. Images with green frames are correct matches, while the ones with red frames are incorrect matches.

Table 3 Recall and precision values at maximum precision and recall respectively comparing different methods on the Nordland dataset.

	spring-summer		spring-fall		spring-winter		summer-fall		summer-winter		fall-winter	
	Recall at 100% precision	Precision at max recall	Recall at 100% precision	Precision at max recall	Recall at 100% precision	Precision at max recall	Recall at 100% precision	Precision at max recall	Recall at 100% precision	Precision at max recall	Recall at 100% precision	Precision at max recall
FDNet_512	**51.83%**	16.42%	**52.69%**	24.83%	**40.52%**	3.50%	**88.52%**	93.20%	**33.74%**	3.82%	**33.39%**	5.40%
Conv3	1.91%	7.24%	1.47%	9.28%	3.48%	4.89%	1.91%	24.50%	0.35%	3.44%	1.39%	4.22%
Gist	14.61%	47.08%	4.87%	51.47%	1.04%	2.27%	63.65%	**95.67%**	0.35%	2.14%	1.22%	2.28%
DBow3	0.52%	1.11%	0.52%	1.19%	0.17%	0.91%	12.0%	3.99%	0.70%	0.84%	0.52%	0.76%
Landmarks	7.39%	17.43%	10.43%	17.57%	9.13%	7.49%	80.43%	85.82%	4.78%	4.28%	0.87%	4.03%
Conv4_fine-tuned	8.34%	14.6%	2.43%	27.11%	7.34%	8.17%	8.52%	31.88%	1.04%	3.72%	1.04%	4.70%
NetVLAD	33.91%	**59.43%**	37.83%	**52.87%**	25.22%	**22.63%**	83.04%	91.63%	30.87%	**22.12%**	10.00%	**21.39%**
CALC	16.33%	45.41%	41.8%	40.95%	28.92%	15.62%	57.90%	85.47%	15.27%	20.13%	15.10%	15.58%

Table 4 Recall and precision values at maximum precision and recall respectively comparing different methods on the Alderley, Oxford RobotCar, St Lucia and FAS Dataset.

	Alderley day-night		Oxford RobotCar night-day		St Lucia morning-afternoon		FAS summer-winter	
	Recall at 100% precision	Precision at max recall	Recall at 100% precision	Precision at max recall	Recall at 100% precision	Precision at max recall	Recall at 100% precision	Precision at max recall
FDNet_512	10.82%	10.28%	7.13%	9.73%	**14.43%**	11.13%	**24.80%**	15.69%
Conv3	0.37%	9.41%	2.57%	12.22%	0.17%	9.06%	3.80%	10.11%
Gist	2.24%	20.15%	5.40%	11.85%	2.43%	8.42%	4.77%	7.99%
DBow3	1.12%	3.85%	1.74%	7.00%	0.87%	5.37%	0.42%	2.96%
Landmarks	1.12%	22.75%	2.77%	16.60%	3.91%	20.54%	1.42%	14.41%
Conv4_fine-tuned	4.48%	6.10%	7.73%	10.91%	5.40%	12.31%	8.28%	12.26%
NetVLAD	**11.54%**	**33.40%**	**8.97%**	**31.27%**	13.04%	**24.23%**	20.11%	**22.35%**
CALC	6.40%	19.30%	6.97%	20.30%	4.83%	16.97%	10.46%	15.69%

ance vectors to low-dimensional space in Fig. 13 (b). It is observed that there is a gap between the feature distribution under the day and that under the night. However, the distribution of points drawn is not very concentrated, and it seems to be able to describe the direction information of the original images. Perhaps because appearance vectors perceive that there are obvious trajectory changes of the images on the test set.

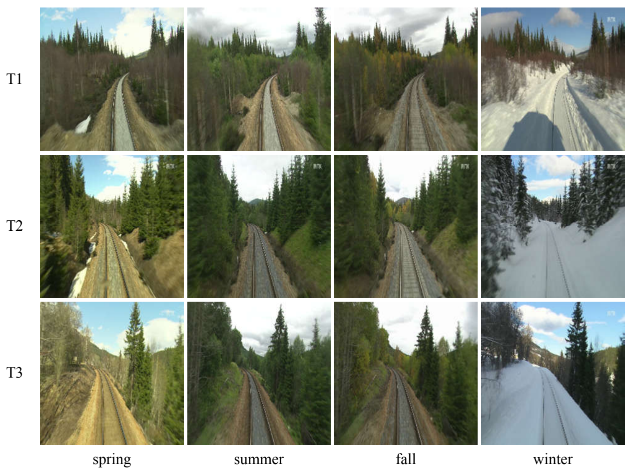


Fig. 10 Examples of different location in Nordland dataset.

5.2.4 Results on Oxford RobotCar Dataset

The PR performance on the Oxford RobotCar dataset is shown in Fig. 15. It is notable that NetVLAD has achieved far better results (8.97% recall at max precision) than all other methods, while CLAC follow-ups with relatively poor performance. Gist and DBoW3, which are based on hand-crafted features, still perform poorly. Disappointingly, our method doesn't show any advantages in this dataset (only 7.13% recall at max precision). The main reason could be the significant loss of visual information at night-time and dynamic objects such as pedestrians and cars. As displayed in the fourth query (column) of Fig. 9, the FDNet can not obtain the right match because a moving car appears in the scene.

5.2.5 Results on St Lucia Dataset

The PR curves plotted in Fig. 16 show competitive accuracy for the proposed method in this challenging case. As expected, our method achieves the best performance in terms of the recall values at max precision (14.43%) followed by NetVLAD (13.04%). CALC and Conv4_fine-tuned have shown similar performance on this dataset. Landmarks obtain slightly better results than Conv3 and DBow3, thanks to the fact that the scene contains some visible road signs. The fifth query (column) in Fig. 9 is an example. In the case of moder-

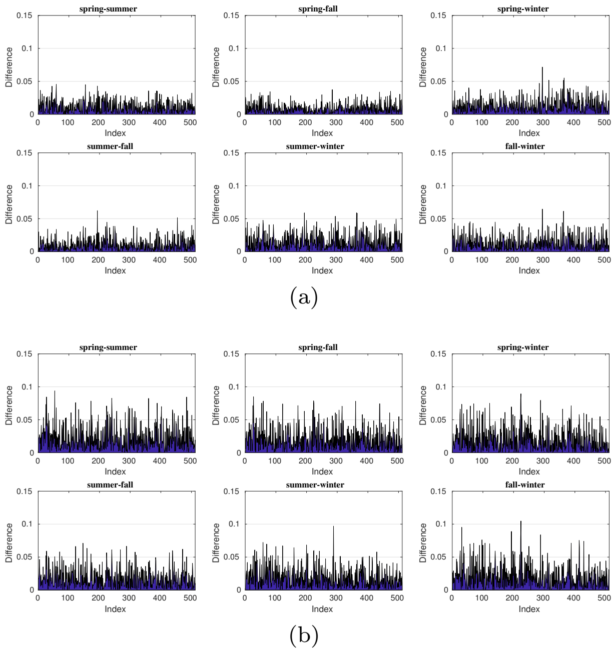


Fig. 11 The response of the absolute difference of content vectors. (a) The absolute difference of content vectors in location $T1$ across different seasons. (b)The absolute difference of content vectors in location $T1$ and $T2$ across different seasons. Where 'spring-summer' represents the first location is under the spring and the second location is under the summer.

ate changes in appearance, most methods can find the right matches.

5.2.6 Results on FAS Dataset

Similar to the results on the Nordland dataset (summer-winter), our method achieves effective place recognition accuracy on the FAS dataset (as shown in Fig. 17). In terms of recall values at max precision, our method (24.80%) outperforms all others significantly. CALC and Conv4_fine-tuned suffer noticeable performance degradation, with respect to our method and NetVLAD. This experiment shows that under the condition of seasonal variation, our approach can always maintain relatively better performance.

5.3 Robustness to viewpoint changes

Viewpoint change is also a major challenge for visual place recognition systems. The previous sections have examined the performance of the proposed method in the case of significant changes in appearance. In this section, we conduct experiments on the Nordland dataset and simulate viewpoint changes by using shifted image crops with reference to [51]. We use 2000 pairs of images in the summer and winter season which are cropped to half of their original width. Viewpoint changes are simulated by shifting the queried images to the right. Consequently, the performance of various overlaps between images in 100%, 90%, 75% and 65% are compared. Fig. 18 demonstrate the results of this experiment. We found that our method can perform relatively stable in the case of slight viewpoint change (overlap in 90%), but once the viewpoint changes too much, the performance will be significantly reduced. As a result, we continue to explore which features the viewpoint changes are encoded into in our method. An image of the summer is selected as a reference, and its viewpoint changes are simulated as shown in Fig. 19(a). We can observe the changes of content vector and appearance vector of these images in Fig. 19(b) and Fig. 19(c). The phenomenon that the content feature changes greatly while the appearance feature does not change at all indicates that the viewpoint change is considered as 'content' in our algorithm.

Table 5 Runtime comparison between different methods on The Nordland Dataset

VPR System	Feature Extraction (ms)	Feature Matching (ms)
Conv3	136	0.34
Gist	223	0.38
DBoW3	2.3	0.013
Landmarks	737	19
Conv4_fine-tuned	158	0.36
NetVLAD	980	0.038
CALC	39	0.31
FDNet_512	21.2	0.35

5.4 Computational Performance

In this section, we evaluate the computational cost in terms of the running time for (1) feature extraction from the networks, (2) feature matching between reference and a single query image. Note that the reported times in this paper were tested on Intel Xeon CPU at 2.10GHz, and that feature extraction was performed on NVIDIA TITANX GPU with 12GB memory. Table 5 shows evaluation results on the Nordland Dataset. We run experiments on 2000 images and record the average runtime. As expected, CNNs-based approaches always take more time to encode an image. Among the competing approaches, the NetVLAD is slower than others. DBoW3 is the most efficient, with an average time of 2.3ms per image, followed by ours at 21.2ms.

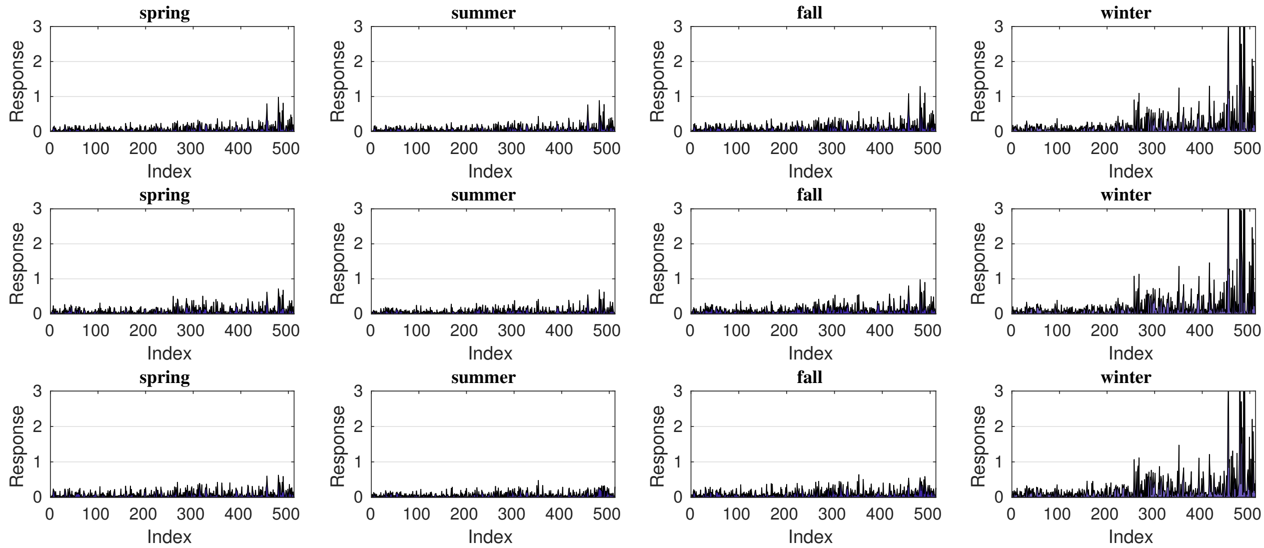
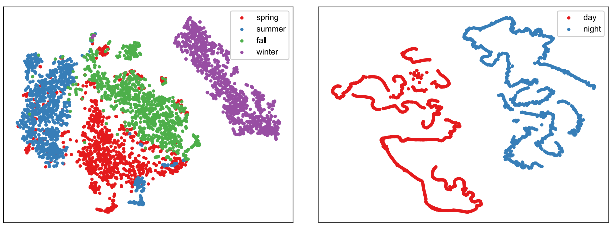


Fig. 12 The response of appearance vectors. From top to bottom, each row in turn belongs to T1, T2 and T3



(a) (b)

Fig. 13 The distribution of appearance features mapped into two-dimensional space. (a) Nordland dataset (b) Alderley dataset.

6 Conclusion and future works

In this work, we have proposed a method for visual place recognition which exploits the content information extracted by feature disentanglement. Employing the convolutional auto-encoder and adversarial learning, the original image is decoupled into content and appearance information. Through the competition with the discriminators and content encoder, the encoder learns to extract features good for content factor recognition but not useful for appearance factor recognition. Furthermore, the network is trained stably without perfectly aligned images and can handle multiple appearance changes in place recognition within a unified framework. The generated content features are directly used to compare the similarity of images without dimensionality reduction operations. Finally, we use the similarity matrix to check possible loops in the test datasets to evaluate the performance.

We have performed thorough comparison studies on different datasets against the state-of-the-art image description methods for place recognition, where the

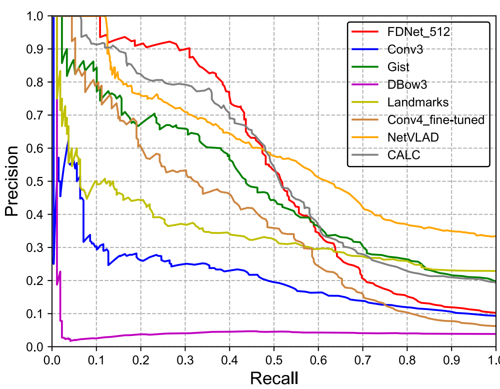


Fig. 14 Precision-recall curves comparing the different approaches with our novel method on the Alderley dataset.

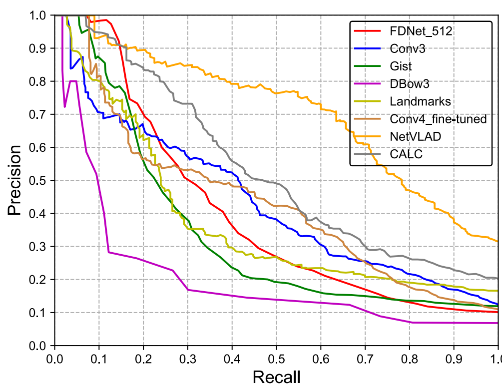


Fig. 15 Precision-recall curves comparing the different approaches with our novel method on the Oxford RobotCar dataset.

extensive experimental results have demonstrated that the proposed method achieves a satisfactory precision in changing conditions and generally outperforms the benchmarks in terms of the recall at perfect precision. Moreover, the two-dimensional distribution of appearance features was displayed, which demonstrated that the appearance feature accurately encodes the appearance information of images.

While the proposed method only considers discrete appearance changes, we will try to deal with the place recognition problem in the continuously changing environment [55] because most appearance changes such as weather and lighting always change with time. Besides, we will furthermore address the remaining challenge of viewpoint robustness.

Acknowledgements Supported by National Natural Science Foundation of China(61973066), Advanced Technology Project(No. 41412050202, 61403120111), Fundamental Research Funds for the Central Universities(N172608005, N1826-08004), Natural Science Foundation of Liaoning (No.2018052-0040) and National Natural Science Foundation of China (No. 61471110)

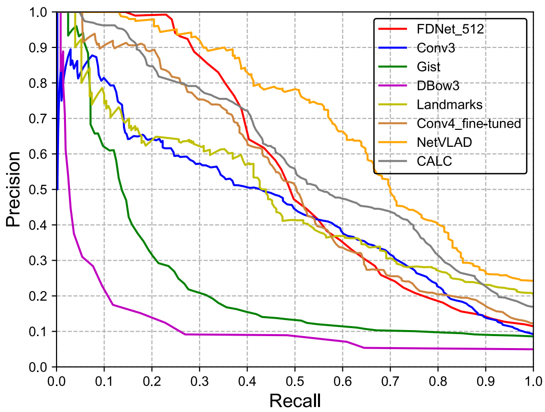


Fig. 16 Precision-recall curves comparing the different approaches with our novel method on the St Lucia dataset.

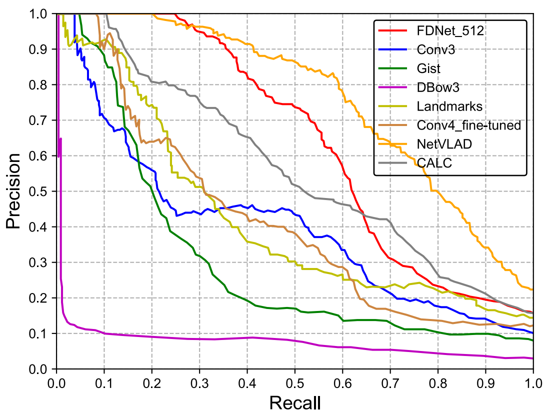


Fig. 17 Precision-recall curves comparing the different approaches with our novel method on the FAS dataset.

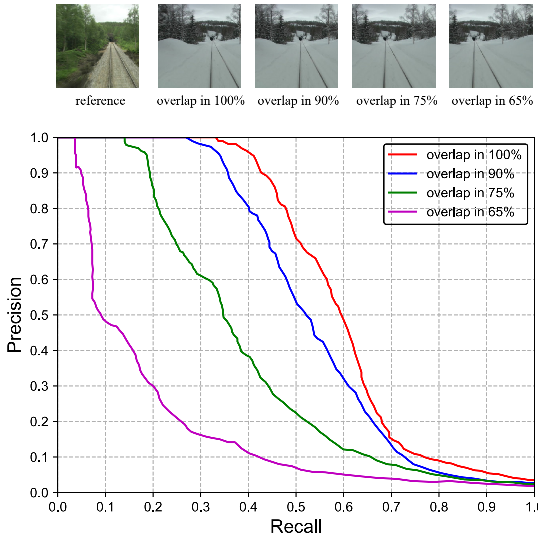


Fig. 18 Experiments under synthetic viewpoint change using cropped and shifted images of the Nordland summer and winter dataset. Top row: Examples for the simulated viewpoint variation. Bottom: Precision-recall curves for different overlap values.

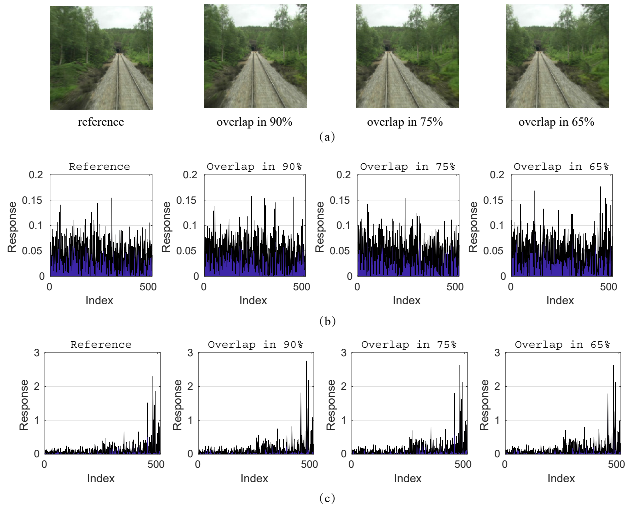


Fig. 19 (a) Examples for the simulated viewpoint variation at the same place. (b) The response of content vectors. (c) The response of appearance vectors.

References

1. Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: Netvlad: Cnn architecture for weakly supervised place recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5297–5307 (2016)
2. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein gan. arXiv preprint arXiv:1701.07875 (2017)
3. Arroyo, R., Alcantarilla, P.F., Bergasa, L.M., Romera, E.: Fusion and binarization of cnn features for robust topo-

logical localization across seasons. In: Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on, pp. 4656–4663. IEEE (2016)

4. Ballard Michael J. Swain, D.H.: Color indexing. International Journal of Computer Vision **7**(1), 11–32 (1991)
5. Bay, H., Ess, A., Tuytelaars, T., Gool, L.V.: Speeded-up robust features (surf). Computer Vision & Image Understanding **110**(3), 346–359 (2008)
6. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence **35**(8), 1798–1828 (2013)
7. Bingham, E., Mannila, H.: 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, pp. 245–250. ACM (2001)
8. Carlevaris-Bianco, N., Eustice, R.M.: Learning visual feature descriptors for dynamic lighting conditions. In: Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on, pp. 2769–2776. IEEE (2014)
9. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: Advances in neural information processing systems, pp. 2172–2180 (2016)
10. Chen, Z., Jacobson, A., Sünderhauf, N., Upcroft, B., Liu, L., Shen, C., Reid, I., Milford, M.: Deep learning features at scale for visual place recognition. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 3223–3230. IEEE (2017)
11. Chen, Z., Maffra, F., Sa, I., Chli, M.: Only look once, mining distinctive landmarks from convnet for visual place recognition. In: IEEERSJ International Conference on Intelligent Robots & Systems (2017)
12. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: Workshop on statistical learning in computer vision, ECCV, vol. 1, pp. 1–2. Prague (2004)
13. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. Proc.int.conf.comp.vis.patt.recog **1**(12), 886–893 (2005)
14. Donahue, J., Krähenbühl, P., Darrell, T.: Adversarial feature learning. arXiv preprint arXiv:1605.09782 (2016)
15. Gálvez-López, D., Tardós, J.D.: Bags of binary words for fast place recognition in image sequences. IEEE Transactions on Robotics **28**(5), 1188–1197 (2012). DOI 10.1109/TRO.2012.2197158
16. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: International Conference on Machine Learning, pp. 1180–1189 (2015)
17. Glover, A.J., Maddern, W.P., Milford, M.J., Wyeth, G.F.: Fab-map+ ratslam: Appearance-based slam for multiple times of day. In: Robotics and Automation (ICRA), 2010 IEEE International Conference on, pp. 3507–3512. IEEE (2010)
18. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: International Conference on Neural Information Processing Systems, pp. 2672–2680 (2014)
19. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. In: Advances in Neural Information Processing Systems, pp. 5767–5777 (2017)
20. Huang, F.J., Boureau, Y.L., LeCun, Y., et al.: Unsupervised learning of invariant feature hierarchies with applications to object recognition. In: Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, pp. 1–8. IEEE (2007)
21. Huang, X., Liu, M.Y., Belongie, S., Kautz, J.: Multimodal unsupervised image-to-image translation. In: ECCV (2018)
22. Kenshimov, C., Bampis, L., Amirgaliyev, B., Arslanov, M., Gasteratos, A.: Deep learning features exception for cross-season visual place recognition. Pattern Recognition Letters **100**, 124–130 (2017)
23. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: International Conference on Neural Information Processing Systems, pp. 1097–1105 (2012)
24. Lample, G., Zeghidour, N., Usunier, N., Bordes, A., Denoyer, L., et al.: Fader networks: Manipulating images by sliding attributes. In: Advances in Neural Information Processing Systems, pp. 5967–5976 (2017)
25. Latif, Y., Garg, R., Milford, M., Reid, I.: Addressing challenging place recognition tasks using generative adversarial networks. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 2349–2355. IEEE (2018)
26. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A.P., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: CVPR, vol. 2, p. 4 (2017)
27. Liu, A.H., Liu, Y.C., Yeh, Y.Y., Wang, Y.C.F.: A unified feature disentangler for multi-domain image translation and manipulation. In: Advances in Neural Information Processing Systems, pp. 2591–2600 (2018)
28. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. In: Advances in neural information processing systems, pp. 469–477 (2016)
29. Liu, Y.C., Yeh, Y.Y., Fu, T.C., Wang, S.D., Chiu, W.C., Wang, Y.C.F.: Detach and adapt: Learning cross-domain disentangled deep representation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
30. Lowe, D.G.: Lowe, d.: Object recognition from local scale-invariant features. in: Proc. iccv. In: IEEE International Conference on Computer Vision, p. 1150 (1999)
31. Lowry, S., Sünderhauf, N., Newman, P., Leonard, J.J., Cox, D., Corke, P., Milford, M.J.: Visual place recognition: A survey. IEEE Transactions on Robotics **32**(1), 1–19 (2016)
32. Lowry, S.M., Milford, M.J., Wyeth, G.F.: Transforming morning to afternoon using linear regression techniques. In: IEEE International Conference on Robotics and Automation, pp. 3950–3955 (2014)
33. Maddern, W., Pascoe, G., Linegar, C., Newman, P.: 1 year, 1000 km: The oxford robotcar dataset. The International Journal of Robotics Research **36**(1), 3–15 (2017)
34. Maddern, W., Stewart, A., McManus, C., Upcroft, B., Churchill, W., Newman, P.: Illumination invariant imaging: Applications in robust vision-based localisation, mapping and classification for autonomous vehicles. In: Proceedings of the Visual Place Recognition in Changing Environments Workshop, IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, vol. 2, p. 3 (2014)
35. Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., Frey, B.: Adversarial autoencoders. arXiv preprint arXiv:1511.05644 (2015)

36. McManus, C., Churchill, W., Maddern, W., Stewart, A.D., Newman, P.: Shady dealings: Robust, long-term visual localisation using illumination invariance. In: Robotics and Automation (ICRA), 2014 IEEE International Conference on, pp. 901–906. IEEE (2014)
37. Merrill, N., Huang, G.: Lightweight unsupervised deep loop closure. In: Proc. of Robotics: Science and Systems (RSS). Pittsburgh, PA (2018)
38. Milford, M.J., Wyeth, G.F.: Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights. In: Robotics and Automation (ICRA), 2012 IEEE International Conference on, pp. 1643–1649. IEEE (2012)
39. Mur-Artal, R., Montiel, J.M.M., Tards, J.D.: Orb-slam: A versatile and accurate monocular slam system. IEEE Transactions on Robotics **31**(5), 1147–1163 (2017)
40. Naseer, T., Burgard, W., Stachniss, C.: Robust visual localization across seasons. IEEE Transactions on Robotics **34**(2), 289–302 (2018)
41. Naseer, T., Oliveira, G.L., Brox, T., Burgard, W.: Semantics-aware visual localization under challenging perceptual conditions. In: Robotics and Automation (ICRA), 2017 IEEE International Conference on, pp. 2614–2620. IEEE (2017)
42. Naseer, T., Spinello, L., Burgard, W., Stachniss, C.: Robust visual robot localization across seasons using network flows. In: AAAI, pp. 2564–2570 (2014)
43. Neubert, P., Sünderhauf, N., Protzel, P.: Superpixel-based appearance change prediction for long-term navigation across seasons. Robotics and Autonomous Systems **69**, 15–27 (2015)
44. Odena, A., Olah, C., Shlens, J.: Conditional image synthesis with auxiliary classifier gans. In: International Conference on Machine Learning, pp. 2642–2651 (2017)
45. Odena, A., Olah, C., Shlens, J.: Conditional image synthesis with auxiliary classifier GANs. In: Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 2642–2651. PMLR (2017)
46. Oliva, A., Torralba, A.: Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope. Kluwer Academic Publishers (2001)
47. Radenović, F., Tolias, G., Chum, O.: CNN image retrieval learns from BoW: Unsupervised fine-tuning with hard examples. In: ECCV (2016)
48. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 815–823 (2015)
49. Sünderhauf, N., Protzel, P.: Brief-gist-closing the loop by simple means. In: Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, pp. 1234–1241. IEEE (2011)
50. Sünderhauf, N., Shirazi, S., Dayoub, F., Upcroft, B., Milford, M.: On the performance of convnet features for place recognition. In: Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on, pp. 4297–4304. IEEE (2015)
51. Sünderhauf, N., Shirazi, S., Jacobson, A., Dayoub, F., Pepperell, E., Upcroft, B., Milford, M.: Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free. Proceedings of Robotics: Science and Systems XII (2015)
52. Taigman, Y., Polyak, A., Wolf, L.: Unsupervised cross-domain image generation. arXiv preprint arXiv:1611.02200 (2016)
53. Valgren, C., Lilienthal, A.J.: Sift, surf & seasons: Appearance-based long-term localization in outdoor environments. Robotics and Autonomous Systems **58**(2), 149–156 (2010)
54. Wulfmeier, M., Bewley, A., Posner, I.: Addressing appearance change in outdoor robotics with adversarial domain adaptation. In: Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on, pp. 1551–1558. IEEE (2017)
55. Wulfmeier, M., Bewley, A., Posner, I.: Incremental adversarial domain adaptation for continually changing environments. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 1–9. IEEE (2018)
56. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: IEEE International Conference on Computer Vision (2017)

Cao Qin received the B.S. degree in Automation from Northeastern University, Shenyang, China. He is currently studying toward the Ph.D. degree with the College of Information Science and Engineering, Northeastern University, Shenyang, China. He has published several English research papers and conference papers. His research interests include visual SLAM, place recognition and deep learning.

Yunzhou Zhang received B.S. and M.S. degree in Mechanical and Electronic engineering from National University of Defense Technology, Changsha, China in 1997 and 2000, respectively. He received Ph.D. degree in pattern recognition and intelligent system from Northeastern University, Shenyang, China, in 2009. He is currently a professor with the Faculty of Robot Science and Engineering, Northeastern University, China. Now he leads the Cloud Robotics and Visual Perception Research Group. His research has been supported by funding from various sources such as National Natural Science Foundation of China, Ministry of science and technology of China, Ministry of Education of China and some famous high-tech companies. He has published many journal papers and conference papers in intelligent robots, computer vision and wireless sensor networks. His research interests include intelligent robot, computer vision, and sensor networks.

Yan Liu received the B.S. degree in Mathematics and Applied Mathematics from Tonghua Normal University, Tonghua, China, in 2016, and the M.S. degree in System Theory from Northeastern University, Shenyang, China, in 2018. She currently is a Ph.D. student in Faculty of Robot Science and Engineering, Northeastern University of China. Her research interest is intelligent robot.

Sonya Coleman received the B.Sc. degree (Hons.) in mathematics, statistics, and computing, and the Ph.D. degree in mathematics from Ulster University, Londonderry, U.K., in 1999 and 2003, respectively. She is currently a Professor with the School of Computing and Intelligent System, Ulster University, and also a Cognitive Robotics Team Leader with the Intelligent Systems Research Centre. Her research has been supported by funding from various sources such as EPSRC, The Nuffield Foundation, The Leverhulme Trust, and the EU. She was involved in the EU FP7 funded projects RUBICON, VISUALISE, and SLANDAIL. She has authored or co-authored over 150 publications in robotics, image processing, and computational neuroscience. Dr. Coleman was awarded the Distinguished Research Fellowship by Ulster University in recognition of her contribution research in 2009.

Dermot Kerr received the B.Sc. degree (Hons.) in computing science and the Ph.D. degree in computing and engineering from Ulster University, Londonderry, U.K., in 2005 and 2008, respectively. He is currently a Lecturer with the School of Computing, Engineering and Intelligent System, Ulster University. He was involved in the EU FP7 funded projects VISUALISE and SLANDAIL. His current research interests include computational intelligence, biologically inspired image processing, mathematical image processing, feature detection, omnidirectional vision, and robotics. Dr. Kerr is an Officer and a member of the Irish Pattern Recognition and Classification Society.

Lv Guanghao received the B.E. degree from Northeastern University, shenyang, China. He is currently working toward the MA.Eng degree with the Faculty of Robot Science and Engineering,

Northeastern University, Shenyang, China. His research interests include visual SLAM and Loop Closure Detection.

Author Photo

	Cao Qin
	Yunzhou Zhang
	Yan Liu
	Sonya Coleman
	Dermot Kerr
	Guanghao Lv

Conflict of interest statement

We declare that there is no conflict of interest in the submission of this manuscript entitled "Appearance-invariant place recognition by adversarially learning disentangled representation" by Cao Qin, Yunzhou Zhang, Yan Liu, Sonya Coleman, Dermot Kerr and Guanghao Lv, and the manuscript is approved by all authors for publication. We would also like to declare that the work described was original research that has not been published previously, and not under consideration for publication elsewhere, in whole or in part. We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted. All authors listed have approved the manuscript that is enclosed.

Cao Qin, Yunzhou Zhang*, Yan Liu, Sonya Coleman, Dermot Kerr and Guanghao Lv

Corresponding Author: Prof. Yunzhou Zhang.
College of Information Science and Engineering, Northeastern University, Shenyang, China.
Phone: +86-24-83687761.
Email: zhangyunzhou@mail.neu.edu.cn