2019 International Conference on Robotics and Automation (ICRA)
Palais des congres de Montreal, Montreal, Canada, May 20-24, 2019

A Multi-Domain Feature Learning Method for Visual Place Recognition

Peng Yin[1,*], Lingyun Xu[1], Xueqian Li[3], Chen Yin[4], Yingli Li[1],
Rangaprasad Arun Srivatsan[3], Lu Li[3], Jianmin Ji[2,*], Yuqing He[1]

***Abstract*—Visual Place Recognition (VPR) is an important component in both computer vision and robotics applications, thanks to its ability to determine whether a place has been visited and where specifically. A major challenge in VPR is to handle changes of environmental conditions including weather, season and illumination. Most VPR methods try to improve the place recognition performance by ignoring the environmental factors, leading to decreased accuracy decreases when environmental conditions change significantly, such as day versus night. To this end, we propose an end-to-end conditional visual place recognition method. Specifically, we introduce the multi-domain feature learning method (MDFL) to capture multiple attribute-descriptions for a given place, and then use a feature detaching module to separate the environmental condition-related features from those that are not. The only label required within this feature learning pipeline is the environmental condition. Evaluation of the proposed method is conducted on the multi-season *NORDLAND* dataset, and the multi-weather *GTAV* dataset. Experimental results show that our method improves the feature robustness against variant environmental conditions.**

I. Introduction

In the last decade, the robotics community has achieved numerous breakthroughs in vision-based simultaneous localization and mapping (SLAM) [1] that have enhanced the navigation abilities of unmanned ground vehicles (UGV) and unmanned aerial vehicles (UAV) in complex environment. Visual place recognition (VPR) [2] or loop closure detection (LCD) helps robots to find loop closure in SLAM framework and is an essential element for accurate mapping and localization. Although many methods have been proposed in recent years, VPR is still a challenging problem under varying environmental conditions. Traditional VPR approaches that use handcrafted features to learn place descriptors for local scene description, often fail to extract valid features when encountering significant changes [3] in environmental conditions, such as changes in season, weather, illumination, as well as viewpoints.

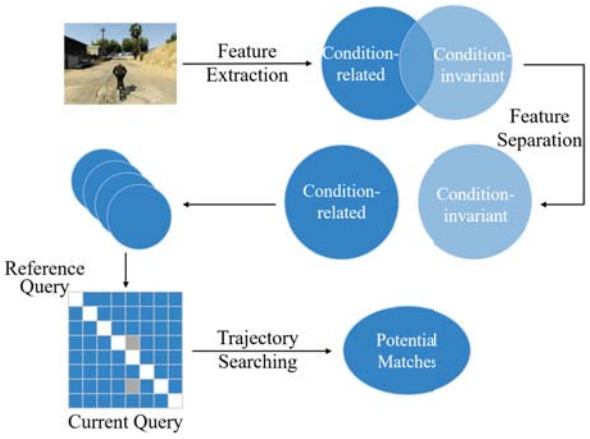


Fig. 1: The pipeline of our proposed conditional visual place recognition method. In summary, there exists three core modules: 1) a CapsuleNet [4] based feature extraction module that is responsible for extracting condition-related and condition-invariant features from the raw visual inputs; 2) a condition enhanced feature separation module to further separate condition-related ones in the joint feature-distribution; 3) a trajectory searching mechanism for finding best matches based on the feature differences of query trajectory features.

Ideally, the place recognition method should be able to capture condition-invariant features for robust loop closure detection, since the appearance of scene objects (e.g., roads, terrains and houses) is often highly related to environmental conditions, and that each object has its own appearance distribution under variant conditions. To the best of our knowledge, there are few VPR methods that have explored how to improve the place recognition performance against variant environmental conditions [5]. A major drawback of these methods is that the change in environmental conditions affects the local features, resulting in decreased accuracy of VPR. In this paper, we propose the condition-directed visual place recognition method to address this issue. Our work consists of two parts: feature extraction and feature separation.

Firstly, in the feature extraction step, we utilize a CapsuleNet-based network [4] to extract multi-domain place features, as shown in Fig. 1. Traditional convolutional neural network (CNN) is efficient in object detection , regression and segmentation, but as pointed out by Hinton, the inner

This paper was supported by the National Natural Science Foundation of China (No. 61573386, No. 91748130, U1608253) and Guangdong Province Science and Technology Plan projects (No. 2017B010110011).

P. Yin, L. Xu, Y. Li and Y. He are with the State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, University of Chinese Academy of Sciences, Beijing. (yinpeng, xulingyun, liyingli, heyuqing@sia.cn) J. Ji is with the School of Computer Science and Technology, University of Science and Technology of China, Hefei Anhui. (jianmin@ustc.edu.cn) X. Li, R.A. Srivatsan and L. Li are with the Biorobotics Lab, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA. (xueqianl, arangapr, lilu12@andrew.cmu.edu) Y. Chen is with the School of Computer Science, University of Beijing University of Posts and Telecommunications, Beijing. (chenyin@bupt.edu.cn)
(Corresponding author: Peng Yin, Jianmin Ji)

connections of objects are easily lost with the deep convolutional and max pooling operations. For instance, in face detection tasks, even if the facial objects (nose, eyes, mouth, lips) are in incorrect layouts, the traditional CNN method may still consider the image as a human face, since it contains all the necessary features of a human face. This problem also exists in place recognition tasks, since different places may contain similar objects but with different arrangements. CapsuleNet uses an dynamic routing method to cluster the shallow convolutional layer features in an unsupervised way. In this paper, we demonstrate another application of CapsuleNet, which could capture feature distribution under specific conditions.

The main contributions of this work can be summarized as follows:

- We propose the use of CapsuleNet-based feature extraction module, and show its robustness in the conditional feature learning for the visual place recognition task.
- We propose a feature separation method for the visual place recognition task, where features are indirectly separated based on the relationship between condition-related and condition-invariant features in an information-theoretic view.

The outline of the paper is as follows: Section II introduces the related works on visual-based place recognition methods. Section III describes our conditional visual place recognition method, which has two components: feature extraction and feature separation. In Section IV, we evaluate the proposed method on two challenging datasets: the *NORDLAND* [6] dataset which has same trajectories under multiple season conditions and a *GTAV* dataset which is generated on the same trajectory under different weather conditions in a game simulator. Finally, we provide concluding remarks in Section V. The linked video[1] provides a visualization of the results of our method.

II. Related Work

Visual place recognition (VPR) methods have been well studied in past several years, and can be classified into two categories: feature- and appearance-based. In feature-based VPR, descriptive features are transformed into local place descriptors. Then, place recognition can be achieved by extracting the current place descriptors and searching similar place indexes in the bag of words. On the contrary, appearance-based VPR uses feature descriptors that are extracted from the entire image, and performs place recognition by assessing feature similarities. SeqSLAM [3] describes image similarities by directly using the sum of absolute difference (SAD) between frames, while vector of locally aggregated descriptors (VLAD) [7] aggregates local invariant features into a single feature vector and uses Euclidean distance between vectors to quantify image similarities.

Recently, many works have investigated CNN-based features for appearance-based VPR tasks. Sünderhauf *et al*. [8] first used pre-trained VGG model to extract middle-layer CNN outputs as image descriptors in the sequence matching pipeline. However, a pre-trained network can not be further trained for place recognition task, since the data labels are hard to define in VPR task. Recently, Chen *et al*. [9] and Garg *et al*. [5] address the conditional invariant VPR as an image classification task and rely on precise but expensive human labeling for semantic labels. Arandjelovic *et al*. [10] developed NetVLAD, which is a modified form of the VLAD features, with CNN networks to improve the feature robustness.

The approach that comes closest to our method is the work of Porav *et al.* [11], where they learn invertible generators based on the CycleGAN [12], The original CycleGAN method can transform the image from one domain to another domain, but such transformation is limited to only two domains. Thus, for multiple domain place recognition task, the method of Porav *et al.* requires transformation model between each pair of conditions. In contrast, our method can learn more than two conditions in the same structure.

III. Proposed Method

In this section, we investigate the details of two core modules in our conditional visual place recognition method.

A. Feature Extraction

a) VLAD: VLAD is a feature encoding and pooling method, which encodes a set of local feature descriptors extracted from an image by using a clustering method such as K-means clustering. For the feature extraction module, we extract multi-domain place features from the raw image, by utilizing a CapsuleNet module. Let q_{ik} be the strength of the association of data vector x_i to the cluster μ_k, such that $q_{ik} \geq 0$ and $\sum_{k=1}^{K} q_{ik} = 1$, where K is the clusters number. VLAD encodes feature x by considering the residuals

$$v_k = \sum_{i=1}^{N} q_{ik}(x_i - \mu_k),$$

and the joint feature description $\{v_1, v_2, v_3, ..., v_N\}$, where N is the local features number.

Assume we can extract N lower feature descriptors (each is denoted as x_i) from the raw image, we can construct a new VLAD like module with the following equation,

$$v_k = \sum_{i=1}^{N} Q_k(l_i) r(x_i, \mu_k), \tag{1}$$

where $r(x_i, \mu_k)$ is the residual function measuring similarities between x_i and μ_k, and $Q_k(l_i)$ is the weighting of capsule vector l_i involved with the k^{th} cluster center.

b) Modified CapsuleNet: In order to transform Eq.1 into an end-to-end learning block, we consider two aspects:

1) Constructing the residual function $r(x_i, \mu_k)$;
2) Assigning the weights $Q_k(l_i)$.

With lower layer features extracted from the shallow convolution layer, we use $N \times D_{property}$ matrix to map lower level

[1] https://youtu.be/dS028yXKNlw

features into higher level features, where N is the CNN unit number in the shallow convolution layer.

In order to integrate the lower-higher feature mapping within a single layer, the local lower level feature x_i should have a linear mapping layer to represent the residual function $r(x_i, \mu_k) = f(x_i, \mu_k) - \frac{1}{N}\sum_k f(x_i, \mu_k)$, where $f(x_i, \mu_k) = \mathbf{W}_{ik}x_i + b_k$, $\mathbf{W}_{ik}$ and b_k are the linear transformation weighting and bias for the k^{th} capsule center.

Furthermore, to estimate the local capsule features weighting $Q_k(x_i)$, we apply a soft assignment estimation defined as $Q_k(x_i) = \frac{\exp(b_{ik})}{\sum_j^K \exp(b_{ij})}$, where b_{ik} is the probability that the i^{th} local capsule feature belonging to k^{th} capsule cluster c_k. Therefore, Eq.1 can be written in the following format,

$$v_k = \sum_{i=1}^{N} \frac{\exp(b_{ik})}{\sum_j^K \exp(b_{ij})}(f(x_i, \mu_k) - \frac{1}{N}\sum_k f(x_i, \mu_k)).$$

In order to learn the parameters b_{ij}, W_{ik}, and c_k, we apply the iterative dynamic routing mechanism as described in [4]. For the output of N_{object} higher level features, we assume the last D_C dimensions are assigned as the condition features, e.g. $D_C = 4$ is in the case where the condition is *season*.

B. Feature Separation

In the previous section, we described the feature extraction module p_θ. In this section, we use an additional decoder module q_ϕ, and two reconstruction modules on feature $\mathcal{L}_{Feature}$ and image $\mathcal{L}_{Image}$ domain to achieve the feature separation. Naturally, condition-invariant feature Z_G and condition-related feature Z_C are highly correlated. Fig. 2 shows the relationship between information Z_G and Z_C. $H(Z_G, Z_C)$ and $I(Z_G; Z_C)$ are the joint entropy and the mutual entropy respectively, while $H(z_G|z_C)$ and $H(z_C|z_G)$ are the conditional entropy.

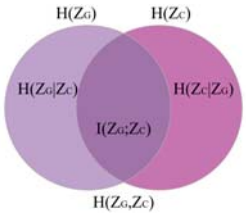


Fig. 2: The relationship of condition-related Z_C and condition-invariant Z_G feature in the information theory view.

From the view of information theory, feature separation can be achieved in the following ways:

- Decrease the conditional entropy $H(z|x)$: less conditional entropy enforces the unique mapping from $x \in \chi$ to $z \in (Z_G, Z_C)$;
- Improve the geometric feature extraction capability: the more accurate geometry we capture, the higher LCD accuracy we can achieve;
- Reduce the mutual entropy $I(Z_G; Z_C|x)$: use environmental conditions to direct feature extraction.

We add these three restrictions in our feature separation module.

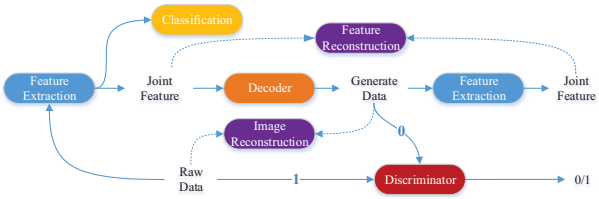


Fig. 3: The framework of feature separation. The networks are combined with four modules: the feature extraction module as given in the previous section; a classification module estimating the environmental conditions; a decoder module mapping the extracted feature back to the data domain; a discriminator module distinguishing the generated data and raw data; and two reconstruction loss modules on data and feature domain respectively.

a) Conditional Entropy Reduction: $H(z|x)$ measures the uncertainty of feature z given the data sample x. The conditional entropy $H(z|x) = 0$ can be achieved, if and only if z is the deterministic mapping of x. Thus, reducing $H_{p_\theta}(z|x)$ can improve the uniqueness mapping from x to z, where p_θ is the parameter in the encoder module. However, improving the condition entropy $H_{p_\theta}(z|x)$ is intractable, since we can not access the data-label pair (x, z) directly. An alternative approach is to optimize the upper bound of $H_{p_\theta}(z|x)$, and the upper bound can be obtained through the following equation,

$$\begin{aligned}
\min_{\theta,\phi} H_{p_\theta}(z|x) &\triangleq \min_{\theta,\phi} -\sum p_\theta(z|x)\log(p_\theta(z|x)) \\
&= \min_{\theta,\phi} -\sum p_\theta(z|x)[\log(q_\phi(z|x))] \\
&\quad -\sum p_\theta(z|x)[\log(p_\theta(z|x)) - \log(q_\phi(z|x))] \\
&= \min_{\theta,\phi} H_{p_\theta(z|x)}[\log(q_\phi(z|x))] \\
&\quad - E_{p_\theta(z|x)}[\mathbf{KL}(p_\theta(z|x)\|(q_\phi(z|x)))], \qquad (2)
\end{aligned}$$

where $\mathbf{KL}$ is the Kullback-Leibler divergence. And $H_{p_\theta}(\log(q_\phi(z|x)))$ measures the uncertainty of the predicted feature with a given sample data x. Since we can not extract features from the q_θ directly, we add an additional feature encoder module after the decoder module (see Fig .3). Eq. 2 can be converted into

$$\begin{aligned}
\min_{\theta,\phi} H(z|x) &\leq \min_{\theta,\phi} H_{p_\theta(z|x)}[\log(q_\phi(z|x))] \\
&\triangleq \min_{\theta,\phi} H_{\hat{z}\sim p_\theta(z|x),\hat{x}\sim q_\phi(x|\hat{z})}[\log(p_\theta(z=\hat{z}|\hat{x}))] \\
&= \mathcal{L}_{Feature}(z, \hat{z}), \qquad (3)
\end{aligned}$$

where $\mathcal{L}_{Feature}$ is the *Feature Reconstruction Loss* between feature extracted from the raw data and the reconstructed data. As we can see in Eq. 3, the original $H_{p_\theta}(z|x)$ is transformed into its upper bound $\mathcal{L}_{Feature}(z, \hat{z})$, and the upper bound is reduced only when the feature domain and data domain are perfectly matched.

b) Feature Extraction Improvement: Condition entropy reduction sub-module can restrict the mapping uncertainty from data domain to the feature domain, this restriction is

highly related to the generalization ability of the encoder module. For the place recognition task, there will be highly diverse scenes in practice, however, we can only generate limited samples for network training. In theory, the GAN uses a decoder and discriminator module to learn the potential feature-to-data transformation with limited samples. Thus, we improve the data generalization ability by applying GANs.

$$\mathcal{L}_{GAN} = \min_{\phi} \max_{\omega} E(\log(D_\omega(x)) + E_{x\sim q_\phi(x|z)}(\log(1 - D_\omega(x))). \quad (4)$$

As demonstrated by Goodfellow *et.al* [13], with iterative updating of the decoder module q_ϕ and the discriminator module D_ω, GAN could pull the generated data distribution closer to the real data distribution, and improve the generalization ability of the decoder module q_ϕ.

c) Mutual entropy reduction: $I(z_G; z_C|x)$ is the mutual entropy, which can be extended by

$$I(z_G; z_C|x) = H(z_G|x) + H(z_C|x) - H(z_G, z_C|x),$$

where, reducing the mutual entropy is equivalent to reducing the right-hand term in the above equation. Since the conditional entropy satisfies $0 \leq H(z_G, z_C|x)$, we can find the upper bound of $I(z_G; z_C|x)$ by ignoring $H(z_G, z_C|x)$

$$\min_{\theta} I(z_G; z_C|x) \leq \min_{\theta}(H(z_G|x) + H(z_C|x)).$$

For the condition-related features, we apply a soft-max based classification module $\mathcal{L}_{Cond}$, to reduce the conditional entropy $H(z_C|x)$. Furthermore, we apply an L_2 image reconstruction loss to further restrict the uncertainty $H(z_G|x)$ given a sample data x,

$$\mathcal{L}_{Image} = \|x_{raw} - x_{reco}\|, \quad (5)$$

where x_{raw} and x_{reco} are the raw image and reconstructed one respectively.

By combining Eq.[3, 4, 5] and $\mathcal{L}_{Cond}$, the joint loss function can be obtained as

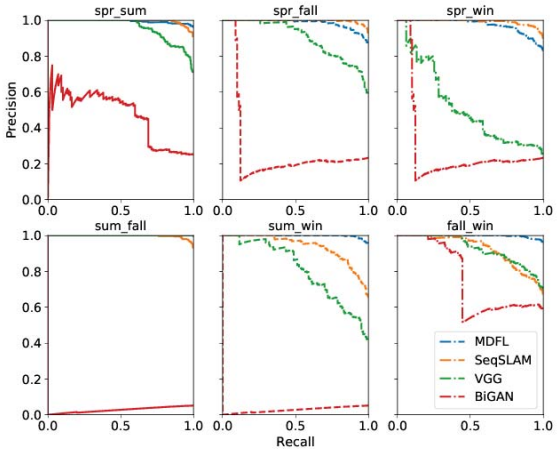
$$\mathcal{L}_{Joint} = \mathcal{L}_{Feature} + \mathcal{L}_{GAN} + \mathcal{L}_{Cond} + \mathcal{L}_{Image}. \quad (6)$$

IV. Experiment Results

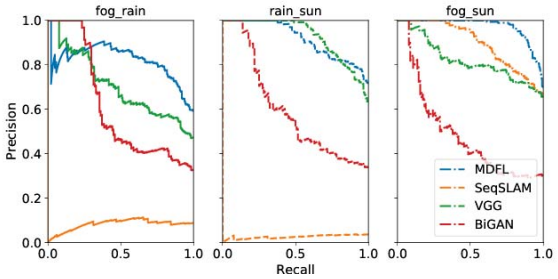
In this section, we analyze the performance of our method on two datasets and compare it with three feature extraction methods for the visual place recognition task [2].

a) Datasets: The datasets we used here are the *Nordland* dataset [14] and the *GTAV* dataset [15]. The *Nordland* dataset was recorded on a train in Norway during four different seasons, and each sequence follows the same track. In each sequence, we generate 17885 frames from the video at 12 Hz, and the first 16885 frames of each sequence is used for training, and the last 1000 frames for testing. Note that we train on all four *Norldand* seasonal datasets, using the seasonal labels to find the condition dependent/invariant features, and then test on the last 1000 frames of each dataset. In the training procedure, we randomly select frames and their corresponding status labels from the four sequences.

[2]The experiments are conducted on a single NVIDIA 1080Ti card with 64G RAM on the Ubuntu 14.04 system. During testing, the framework takes around 2000MB of GPU memory



(a) *Nordland* datasets

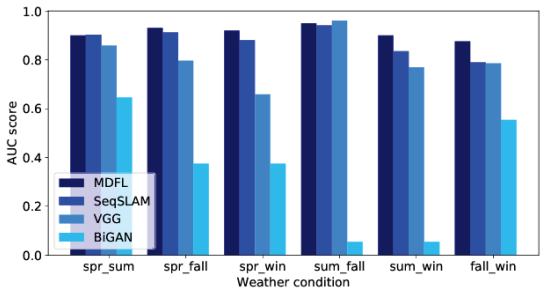


(b) *GTAV* datasets

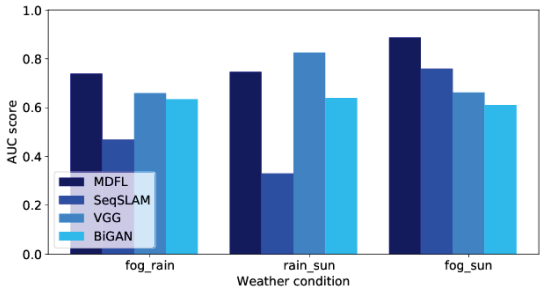
Fig. 4: Precision-Recall curve of various VPR methods on the two datasets. The method is considered to be good if the curve is in the upper-right corner. As we see, MDFL outperforms other methods in most of the cases.

The second dataset *GTAV* [15] contains trajectories on the same track under three different weather status (sunny, rainy and foggy). This dataset is more challenging than the *Nordland* dataset, since the viewpoints are variant in the *GTAV* dataset. We generate more than $10,000$ frames in each sequence, 9000 frames are used as training data, and the remaining 1000 frames for testing. For each dataset, all images are resized to $64 \times 64 \times 3$ in RGB format. The loop closure detection mechanism is followed as in the original SeqSLAM method; sequences of image features are matched instead of a single image. For more details about the structure of the SeqSLAM, we refer the reader to [3].

b) Accuracy Analysis: To investigate the place recognition accuracy, we compare our feature extraction method with three methods in sequential matching – (1) the original feature in SeqSLAM that uses sum of absolute difference as local place feature description, (2) convolution layer feature from VGG network, which is trained on the large-scale image



(a) AUC on *Nordland* datasets



(b) AUC on *GTAV* datasets

Fig. 5: AUC index of the various VPR methods. The methods match the images from the two datasets under different conditions.

classification dataset [16], (3) adversarial feature learning-based unsupervised feature obtained from the generative adversarial networks [17]. The place is considered as being matched when the distance between current frame and target frame is limited within 10 frames. We evaluate the performances in the precision-recall curve (PR-curve), area under curve (AUC) index, inference time, and storage requirement.

Fig.4 and Fig. 5 show the precision-recall curve and AUC index respectively for all the methods on the *Nordland* datasets and the *GTAV* datasets. In Fig. 5, the label spr-sum, spr-fall, etc. refer to the performance using the same network and same model, with different testing sequences.

In general, all the methods perform better in the *Nordland* dataset than in the *GTAV* dataset, since the viewpoints are stable and the geometric changes are smooth due to the constant speed of the train. In contrast, test sequences in *GTAV* datasets have significant viewpoints differences. Furthermore, limited field of view and multiple dynamic objects in *GTAV* also introduces additional feature noises, which causes a significant difference in scene appearance.

VGG features perform well under normal conditions, such as summer-winter in *Nordland*, but perform poorly under unusual conditions, which indicates poor generalizability of VGG features. BiGAN does not perform well on either datasets since it does not take into account the condition of the scene and considers all the images as a joint manifold. For example, the same place under different weather conditions will be encoded differently using BiGAN. Since SeqSLAM uses gray images to ignore the appearance changes under different environmental conditions, the image based features in SeqSLAM are robust to changing conditions. But the matching accuracy decreases greatly in the *GTAV* dataset, since raw image features are very sensitive to the changing viewpoints.

In general, MDFL outperforms the above features in most cases of the *NORDLAND* and *GTAV* datasets, and can handle complex situation well, but is not the best in some situations, such as spring-summer in *NORDLAND* and rain-summer in *GTAV*. One potential reason is that, in each dataset, we only consider one type of environmental condition (Season or weather), but we did not take into account the illumination changes. Since the illumination changes continuously, it is not easy to set this type of environmental condition in training.

Table I shows the average AUC results of different methods in both datasets, and our MDFL method outperforms all the other methods. Notice that for all the above results, the MFDL framework was trained only once end-to-end, and this is one of the advantages of our framework. We do not need to retrain different networks for each pair of conditions.

TABLE I: Average AUC index

Dataset	Caps	SeqSLAM	VGG16	BiGAN
GTAV	**0.790**	0.518	0.715	0.627
Nordland	**0.912**	0.876	0.804	0.345

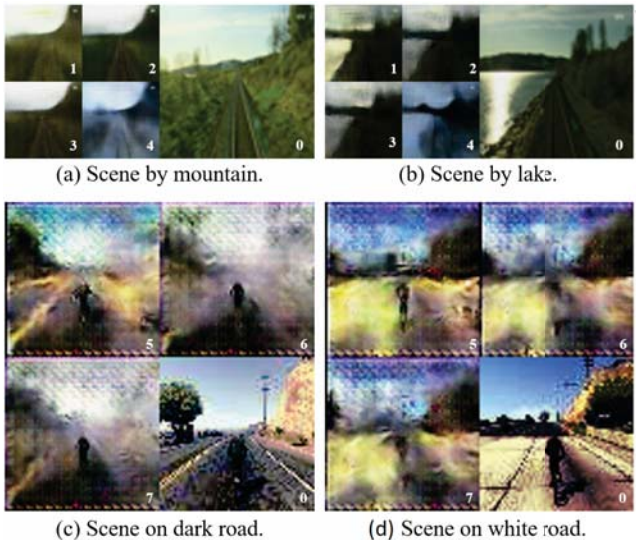


Fig. 6: Image reconstruction using MDFL on *Nordland* (a,b) and *GTAV* datasets (c,d). For (a) and (b), MDFL extracts geometry features from right side image (labelled 0.), and reconstructs them into four different seasons (1. spring, 2. summer, 3. fall, 4. winter) shown on the left side. For (c) and (d), MDFL extracts geometry features from the right bottom image of each scene, and reconstructs it into three images with different weather conditions (5. sunny, 6. rainy and 7. foggy).

c) Image Reconstruction: In order to check whether our framework has learned the geometry features irrespective of the conditions in the scene, we perform a scene reconstruction task using our method. As seen in Fig. 6, for each dataset, we reconstruct images in different conditions from the same input frame. In the *Nordland* dataset, the reconstructed scene around a mountain (Fig. 6(a)) can capture different terrain styles in the seasons; the reconstructed scene around the lake (Fig. 6(b)) can also generalize the lake appearance in winter condition. Still, the strong contrast between sky and road leads to reconstruction failure in other weather conditions, as shown in the case of the white road in Fig. 6(d).

d) Inference time and Storage: The joint features combining both geometry and condition information are stored in a 64×16 matrix with a *char* format. The space required for a local place description is only 1 kB. Assuming that a robot is encoding at 1 Hz, the memory space required for one day of recording is only around 85 MB. This is suitable for long term mobile robot navigation tasks, where the storage is a critical concern. Furthermore, the average inference time for one frame on a GTX 1080Ti and a Jetson Tx1 is around 3.5 ms and 30 ms relatively. This allows the MDFL method to be applied in real-time, on mobile platforms.

V. Conclusion

In this paper, we introduced a novel multi-domain feature learning method for visual place recognition task. At the core of our framework lies the idea of extracting condition-invariant features for place recognition under various environmental conditions. A CapsuleNet-based module was used to capture multi-domain features from the raw image, and apply a feature separation module to indirectly separate condition-related and condition-invariant features. Experiments on the multi-season and multi-weather datasets demonstrate the robustness of our method. The major limitation for our method is that the shallow layer CapsuleNet-based module only clusters lower level features, and is not effective at capturing the semantic descriptions for the place recognition. In our future work, we will investigate hierarchical CapsuleNet network module to extract higher level semantic features for place recognition.

References

[1] H. Zhou, K. Ni, Q. Zhou, and T. Zhang, "An SFM algorithm with good convergence that addresses outliers for realizing mono-SLAM," *Transactions on Industrial Informatics*, vol. 12, no. 2, pp. 515–523, 2016.

[2] S. Lowry, N. Snderhauf, 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.

[3] M. J. Milford and G. F. Wyeth, "Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights," in *IEEE International Conference on Robotics and Automation*, May 2012, pp. 1643–1649.

[4] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Advances in Neural Information Processing Systems*, 2017, pp. 3859–3869.

[5] S. Garg, A. Jacobson, S. Kumar, and M. Milford, "Improving condition-and environment-invariant place recognition with semantic place categorization," in *IROS*. IEEE, 2017, pp. 6863–6870.

[6] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[7] Y. Wang, Y. Cen, R. Zhao, S. Kan, and S. Hu, "Fusion of multiple vlad vectors based on different features for image retrieval," in *13th International Conference on Signal Processing*. IEEE, 2016, pp. 742–746.

[8] N. Sünderhauf, F. Dayoub, S. Shirazi, B. Upcroft, and M. Milford, "On the performance of ConvNet features for place recognition," in *IROS*, 2015, pp. 4297–4304.

[9] 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 *ICRA*. IEEE, 2017, pp. 3223–3230.

[10] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5297–5307.

[11] H. Porav, W. Maddern, and P. Newman, "Adversarial training for adverse conditions: Robust metric localisation using appearance transfer," *arXiv preprint arXiv:1803.03341*, 2018.

[12] Y. Lu, Y.-W. Tai, and C.-K. Tang, "Conditional cyclegan for attribute guided face image generation," *arXiv preprint arXiv:1705.09966*, 2017.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[14] N. Sünderhauf, P. Neubert, and P. Protzel, "Are we there yet? Challenging SeqSLAM on a 3000 km journey across all four seasons," in *Proc. of Workshop on Long-Term Autonomy, ICRA*, 2013.

[15] P. Yin, Y. He, N. Liu, and J. Han, "Condition directed multi-domain adversarial learning for loop closure detection," *arXiv preprint arXiv:1711.07657*, 2017.

[16] M. A. E. Muhammed, A. A. Ahmed, and T. A. Khalid, "Benchmark analysis of popular imagenet classification deep cnn architectures," in *SmartTechCon*, Aug 2017, pp. 902–907.

[17] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," *arXiv preprint arXiv:1605.09782*, 2016.