2019 IEEE Intelligent Vehicles Symposium (IV)
Paris, France. June 9-12, 2019

Learning Low-Rank Images for Robust All-Day Feature Matching

Mengdan Feng[1], Marcelo Ang[1] and Gim Hee Lee[2]

Abstract— **Image-based localization plays an important role in today's autonomous driving technologies. However, in large scale outdoor environments, challenging conditions, e.g., lighting changes or different weather, heavily affect image appearance and quality. As a key component of feature-based visual localization, image feature detection and matching deteriorate severely and cause worse localization performance. In this paper, we propose a novel method for robust image feature matching under drastically changing outdoor environments. In contrast to existing approaches which try to learn robust feature descriptors, we train a deep network that outputs the low-rank representations of the images where the undesired variations on the images are removed, and perform feature extraction and matching on the learned low-rank space. We demonstrate that our learned low-rank images largely improve the performance of image feature matching under varying conditions over a long period of time.**

I. INTRODUCTION

Nowadays, cameras are widely used in autonomous vehicles since they are cheap, easy to mount and images can provide rich semantic and structural information. Image feature extraction and matching have long been an important topic and have wide applications in autonomous vehicle tasks, such as visual mapping and localization. One of the most popular algorithms is visual simultaneously localization and mapping (vSLAM) [1], [2], [3]. However, challenging outdoor environments make feature extraction and matching very difficult. As a result, the reconstructed 3D map and vehicle locations become inaccurate and unreliable.

This paper focuses on the problem of feature matching across images of changing environments taken over a long period of time. Conventional approaches for feature extraction and matching generally consist of three basic steps: (1) key point localization [4]; (2) feature extraction - including hand-crafted descriptors [5], [6], [7] and learned descriptors from deep networks [8], [9], etc. (3) feature matching based on descriptor similarities. However, images suffer severe appearance changes in outdoor environments and this makes it extremely difficult to identify and assign stable key points and descriptors that can be matched across different images.

In this paper, instead of the popular practice of learning robust image descriptors, we propose to learn the low-rank images where the variations in the original input images due to the changing environments are diminished. As a result, feature detection and matching can be done directly on the learned low-rank images without losing accuracy.

Our main contributions in this paper are: (1) we propose a new method to generate low-rank images from the AMOS dataset[1] [10] for our network training and validation. (2) We train two different deep networks, our LR-cGAN and regressed FCNs, to learn the low-rank images from our training data. (3) We show that the performances of most existing feature detectors and descriptors are improved on the low-rank images learned from our deep networks.

II. RELATED WORK

Long-term image-based localization has great importance to autonomous vehicle technologies. Of all the methods, feature-based localization remains popular. Image feature extraction and matching across time have been intensively investigated over the past two decades.

Traditionally, hand-crafted feature descriptors, such as SIFT [5] and SURF [6], etc, were painstakingly designed through great amount of expertise and experiments to ensure its functionality for the image feature matching task. With the rise of deep learning in the recent years, the paradigm has shifted towards learning image descriptors [9], [11], [12] with the promise of more robust descriptors that can be matched across images with large illumination variance. [9] trained a Siamese Network to learn illumination-invariant feature descriptors for day and night images. [11] presented a patch-based unsupervised Convolutional Kernel Network to learn robust image descriptors. [12] proposed to learn key point detection, descriptor extractions and matching through three convolutional networks. However, this approach is time consuming due to manual labelling and less efficient.

In contrast with the existing patch-based feature learning approaches, we employ image-to-image transitional networks to learn low-rank images which are illumination-invariant. Image-to-image translation networks have been widely explored as either pixel-wise classification [13], [14] or regression problems [15], [16]. Typically, the networks are arranged in a downsample-upsample manner using conv-deconv layers to learn the mapping. [14] learns the mapping for image segmentation as a pixel-wise classification problem. [17] learns to hallucinate plausible colors to grayscale images by adding priors to each color bin. [15] estimates depth by jointly optimizing the energy loss.

Instead of assuming conditional independence to each pixel, the state-of-the-art generative adversarial networks (GANs) [16], [18] consider the output as a whole. These networks learn the data distributions from a generator and a discriminator. We will explore different network for high-resolution image-to-image translation in this paper.

[1]Department of Mechanical Engineering, National University of Singapore, Singapore

[2]Department of Computer Science, National University of Singapore, Singapore

[1]http://amos.cse.wustl.edu/dataset

978-1-7281-0560-4/19/$31.00 ©2019 IEEE

III. Learning Image Low-Rank Representations

In this section, we first briefly describe how to generate low-rank representations from image sequences using the "averaged" stable principle component pursuit (SPCP) algorithm for network training. We then describe the proposed network models, i.e., our low-rank conditional generative network (LR-cGAN) and the fully convolutional networks (regressed FCNs), to learn the mapping from the original images to their low-rank representations. We learn the low-rank representations because the SPCP[19] algorithm can only generate low-rank images from image sequences, while our deep network is able to learn the illumination-invariant representation from a single image.

A. "Averaged" SPCP

Given a noisy data matrix Y where each column is made up of a vectorized image I, the objective of SPCP [19] is to decompose Y into the sum of a low-rank matrix L and a sparse matrix S. Formally, L and S can be recovered from solving the following optimization problem:

$$\begin{aligned} \underset{L,S}{\text{minimize}} \quad & \text{rank}(L) + \gamma \|S\|_0 \\ \text{subject to} \quad & \|L+S-Y\|_2 \leq \varepsilon. \end{aligned} \tag{1}$$

γ controls the balance between L and S, ε prevents unknown perturbations $(Y-(L+S))$ in the data.

However, Eq. 1 is a NP-hard non-convex problem which is difficult to optimize. Thus we turn it into a convex optimization problem by relaxing rank(L) into a convex nuclear norm $\|\cdot\|_*$, and replacing the $L0$-norm $\|\cdot\|_0$ with a $L1$-norm $\|\cdot\|_1$. The optimization problem now becomes:

$$\begin{aligned} \underset{L,S}{\text{minimize}} \quad & \|L\|_* + \lambda \|S\|_1 \\ \text{subject to} \quad & \|L+S-Y\|_2 \leq \varepsilon, \end{aligned} \tag{2}$$

which can be easily solved by convex optimization [19].

To further remove image variations, we propose to compute the average of the low-rank images by averaging each column of the low-rank matrix L and use it as the training data. The last row of Fig. 1 shows the "average images" which we use for training our deep networks. We will use the terms low-rank and "average" images interchangeably for the rest of this paper.

B. Deep Image-to-Image Translation Models

The performance of feature extraction and matching is largely determined by the quality of the image. Hence, it is critical to learn a high-quality representation of the low-rank image L from the original image I. In this paper, we mainly focus on our LR-cGAN and regressed FCN.

1) LR-cGAN: Our low-rank conditional generative network learns a mapping from the observed image I and a random noise vector z to the desired low-rank image L. The network consists of two network branches: a generator G and a discriminator D. G is trained to produce an output L' that is highly similar to L, i.e. $G:(I,z)\rightarrow L'$. The objective is to confuse D with fake image L'. While D aims to learn a classifier to identify the true (I,L) and synthesized (I,L') image pairs with opposite probabilities $p1$ and $p2$, i.e. $D:(I,L)\rightarrow p1, D:(I,L')\rightarrow p2$. Therefore, the loss function is expressed as follows:

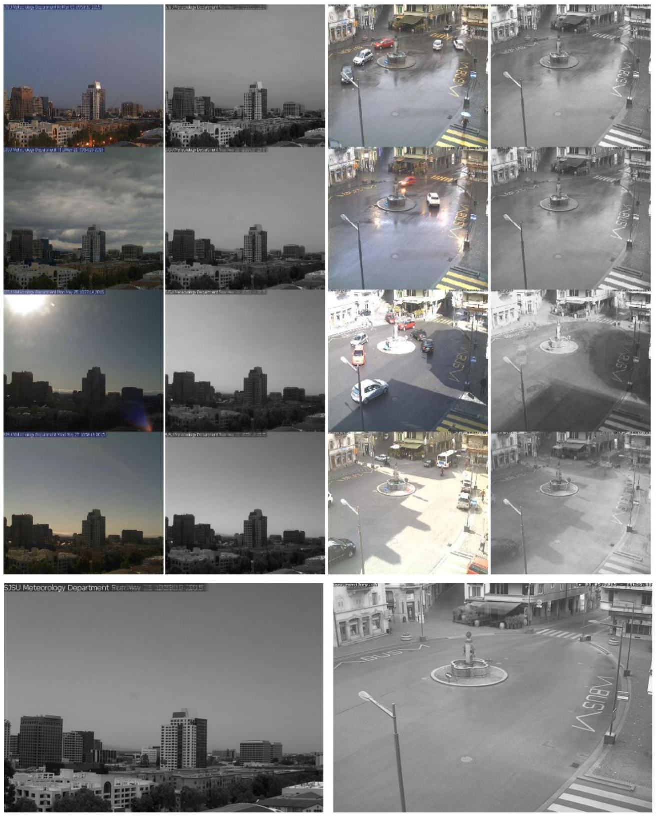

Fig. 1. (1^{st} and 3^{st} columns) Original images and (2^{st} and 4^{st} columns) their corresponding low-rank images recovered from SPCP. (Last row) Proposed "average images" of all the low-rank images.

$$\begin{aligned} L_{cGAN}(D,G) = & E_{I,L\sim p_{data}(I,L)}[logD(I,L)] + \\ & E_{I\sim p_{data}(I),z\sim p_z(z)}[log(I-D(I,G(I,z))] + \\ & \gamma \left\| L' - L \right\|_2 . \end{aligned} \tag{3}$$

The weighted Euclidean loss is added to balance different loss terms and accelerate training. Through experiments, we set γ as 0.95. In contrast to FCNs, our LR-cGAN learns the similarity metric with a classifier and optimize the network based on the probability. Consequently, our LR-cGAN produces higher quality predictions.

2) Regressed FCNs: FCNs are composed of convolutional layers without any fully-connected layers. To regress per-pixel value of the low-rank images, we change the last layer from softmax loss to element-wise Euclidean loss and add the tanh activation function, then train the network with the following objective function:

$$\text{loss}(I) = \|f(I) - L\|_2 . \tag{4}$$

During training, we observe that more detailed and accurate outputs can be obtained by skip connections. However, the direct regression of low-rank estimation is usually blurry and noisy. It was mentioned in [20] that measuring pixel-wise distance with $L2$ loss leads to over-blurring around edges for image generation. Furthermore, the noise could be the results of direct regression without considering neighbourhood relationships. As suggested by [21], we could reduce the noise and blurring to some extent by integrating the unary and pairwise terms into loss function of the network.

C. Network Architectures

In our paper, to learn high resolution images end-to-end, all the networks contain conv-deconv layers in the downsampling-upsampling manner and skip connections to share information between coarse features from lower layers and finer features from deeper layers. The networks are trained on two Nvidia Pascal Titan X GPUs.

1) LR-cGAN: An overview of our network architecture is demonstrated in Fig. 2. Different from [18], for our generator G, we perform feature extraction on input image I and noise z respectively. Suppose *Ck* represent a Convolution-BatchNorm-ReLU layer with k filters, *CDk* means a four fractionally-Strided Convolution-BatchNorm-Drouput-ReLU layer with k filters. For I, we do C64-C128; for z, we do CD16-CD64; then we concatenate feature maps from I and z; and then the following network structure for generator G is: C256-C512-C512-C512-C512-C512-CD512-CD1024-CD1024-C1024-C1024-C512-C256-C128. The network structure for discriminator D is: C64-C128-C256-FC256-Sigmoid.

We randomly crop 256×256 patches from the original images for data augmentation during training. Our LR-cGAN is implemented in Torch [22] with batch size 8 and initial learning rate 2^{-4}. The training took around 2 hours to achieve the results in Fig. 3. We can see that the output images from our LR-cGAN are the closest to the ground truth.

2) Regressed FCNs: We adopt the FCN-VGG16 network as the basic architecture. Two modifications are made based on this model: (1) we replace the final softmax layer with Euclidean loss; (2)we change the output channels to 1 since we learn single channel grayscale low-rank images.

We train three different FCNs, i.e., FCN-32s, FCN-16s and FCN-8s as proposed in [13]. The FCNs are implemented using Caffe [23] and optimized with different base learning rate since they are initialized with different pre-trained models. The results show that FCN-8s generates the best representations with richer information and lesser noise which are shown in Fig. 3.

IV. Experiments

In this section, we first describe how we reconstruct the low-rank representations of each image sequence using "averaged" SPCP. Next, we will briefly discuss about training process of the deep networks to learn the low-rank images. Finally, we apply standard feature descriptors on the learned representations and evaluate the matching performances.

A. Retrieve Low-Rank Images from SPCP

We create a data matrix Y for each image sequence containing 240 images taken at different times. In detail, Y is a matrix with 240 columns, and each column is from one vectorized image from the sequence. Given this training data Y, we use the $SPCP_{sum}$ from [19] to generate the low-rank representation L and sparse term S using the convex optimization method. In our experiment, we chose $\lambda = 0.02$ and $\varepsilon = 0.08 * \|Y\|_2$. Each image sequence takes around 2-5 minutes for the optimization. Then we reconstruct the low-rank images of the original sequence by rearranging each column from the low-rank matrix L.

Some results of the reconstructed images are shown in Fig. 1. As we can see, almost all variations from the strong lighting, shadows, clouds and moving objects are removed in the reconstructed low-rank images. Next, we take the "average" image from the mean of all the reconstructed low-rank images of each sequence. We can see from the bottom row of Fig. 1 that almost all variations in the images have been removed. We will use this "average image" as the ground truth for each image sequence to train the network.

B. Results from the Networks

Some results from the LR-cGAN and ressed FCN-8s are visualized in Fig. 3. As can be seen, our LR-cGAN produces high quality results, while the results from the regressed FCN-8s network are blurry and noisy. As analyzed in Sec. III-B.2 and III-B.1, the major reason is that pixel-wise Euclidean loss in FCNs results in over-blurring. We apply wavelet transformations to denoise and improve the quality of the images from the FCN-8s network. The results are shown in Fig. 3. We can see that the denoised output from FCN-8s is much smoother and less noisy.

C. Feature extraction and matching

We apply existing feature descriptors to perform feature extraction and matching on the learned low-rank images from LR-cGAN and regressed FCNs. A variety of local feature descriptors that includes SIFT, SURF, ORB and BRISK are tested on the low-rank images. We set two distance threshold value t_x and t_y for each matched feature pair to remove the outliers. We check the following hard threshold to find out if two descriptors (x_1, y_1) and (x_2, y_2) from image I_1 and I_2 are a matching pair or outliers:

$$d_x = \|x_1 - x_2\|_2 \leq t_x, \tag{5}$$

$$d_y = \|y_1 - y_2\|_2 \leq t_y. \tag{6}$$

d_x and d_y are the distance between two key points in the x and y direction respectively. Through experiments, we set both t_x and t_y to be 10 pixels. Matched pairs that does not satisfy Eq. 5 and 6 are considered outliers.

Fig. 5 shows the comparison of SIFT feature matching on the original, low-rank FCN-8s and LR-cGAN images taken at 5am and 6pm respectively. It can be seen that SIFT

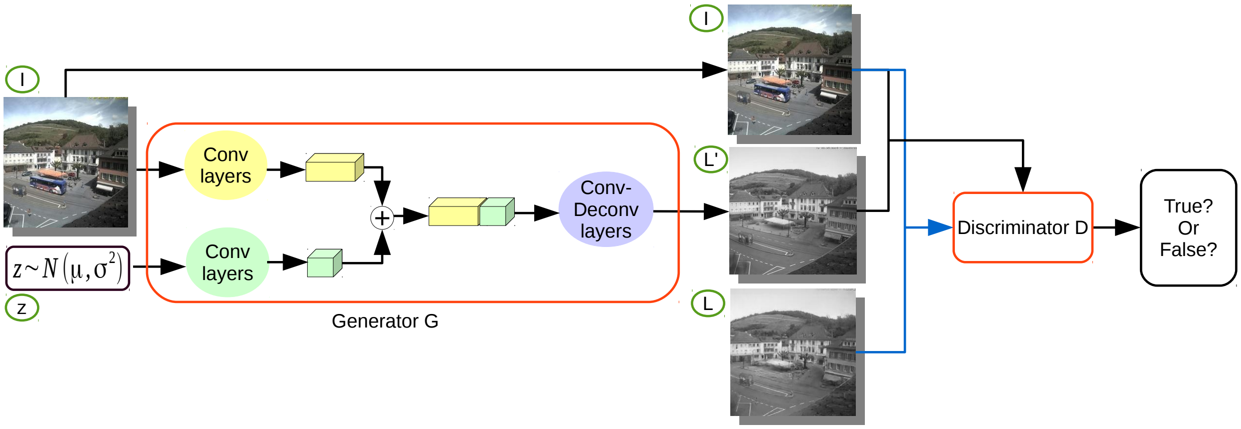

Fig. 2. Our low-rank conditional generative network (LR-cGAN).

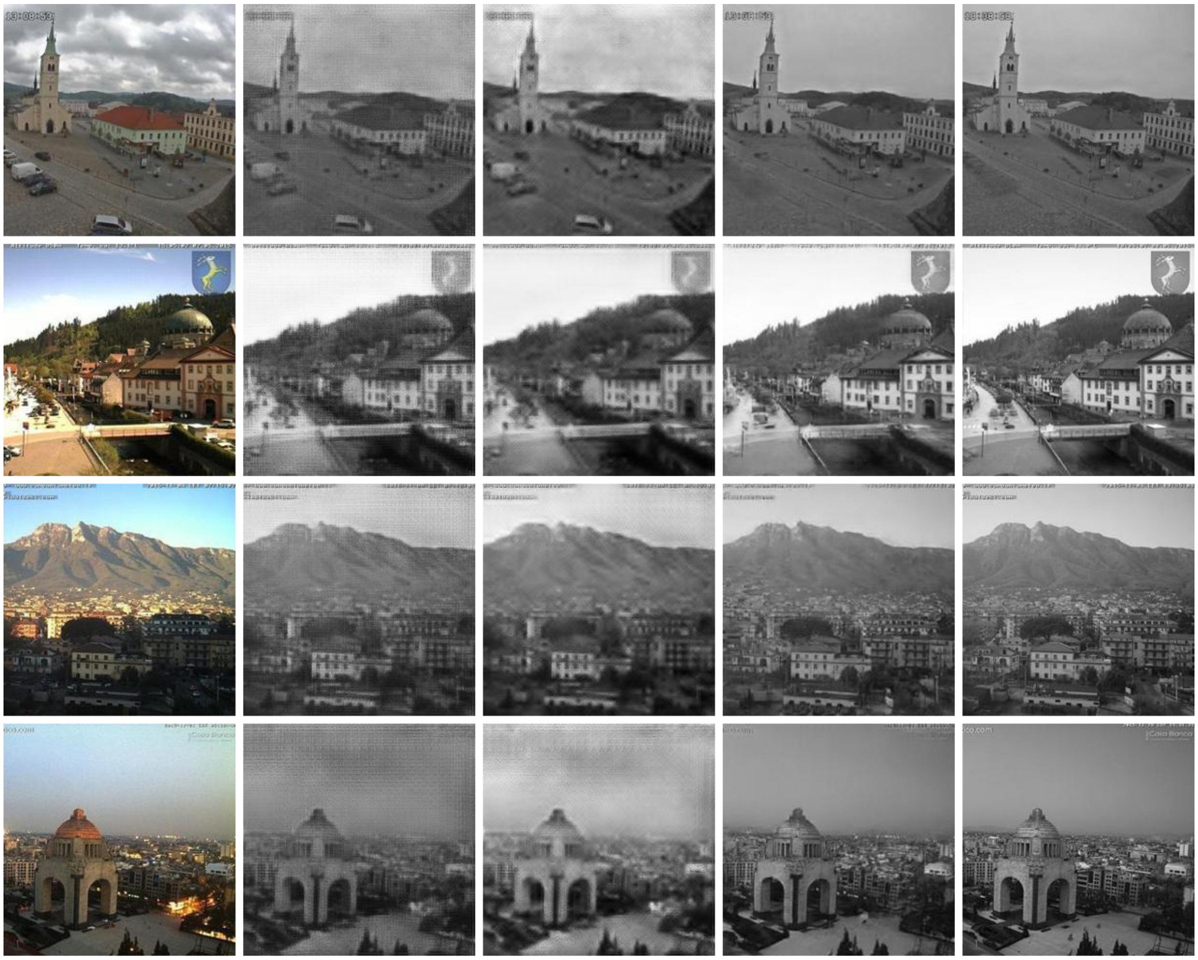

Fig. 3. Some examples of the low-rank images generated from the different networks. From left to right: input images, predictions from FCN-8s, denoised FCN-8s, our LR-cGAN, ground truth.

matches perform the worst on the original image. Both low-rank images from the FCN-8s and LR-cGAN show huge improvements over the original images.

We compare the performances of both SIFT and SURF feature matching on the original and low-rank images learned from our LR-cGAN and the FCN-8s. We show the comparisons of SIFT feature matching across time over 16 image sequences on the original images, FCNs, denoised-FCNs and our LR-cGAN in Fig. 4. Comparisons of the binary descriptors are omitted since they perform poorly.

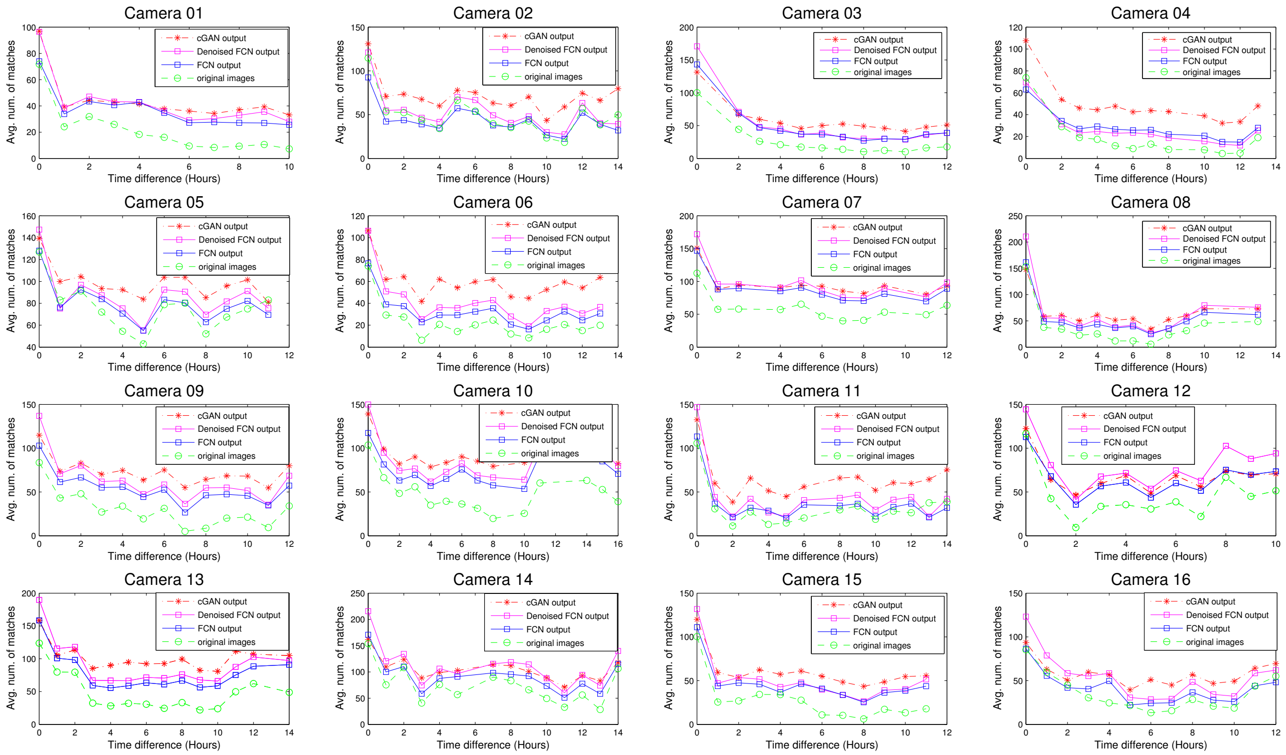

Fig. 4. Inlier SIFT matches for all image sequences from the 16 cameras at various time differences. It is obvious that SIFT matching from the learned representations, i.e. LR-cGAN, denoised FCNs and FCNs maintained high matching performance through time, showing that they are much more robust to lighting changes.

TABLE I

AVERAGE INCREASE RATE OF ALL THE 16 IMAGE SEQUENCES AT EACH TIME DIFFERENCE USING SIFT AND SURF (OTHER DESCRIPTORS NOT WORK WELL FOR LONG-TERM IMAGES) MATCHING ON LEARNED IMAGES

Method		**Time difference between images (Hours)**																
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
SIFT [5]	FCNs[13]	0.09	0.28	0.47	0.61	0.65	0.76	1.09	1.38	1.16	0.75	0.95	0.82	0.60	0.32	0.19	0.61	0.81
	Denoised FCNs	**0.37**	0.50	0.68	0.82	0.81	0.91	1.27	1.69	1.46	1.05	1.10	0.98	0.75	0.50	0.38	0.85	**1.04**
	LR-cGAN	0.22	**0.62**	**0.83**	**1.44**	**1.29**	**1.69**	**2.10**	**2.47**	**2.33**	**1.79**	**1.94**	**1.85**	**1.67**	**1.30**	**0.68**	**0.86**	0.92
SURF [6]	FCNs[13]	-0.05	0.35	0.56	0.72	1.25	1.37	1.92	1.55	1.63	1.23	2.68	9.86	2.49	0.63	0.26	0.83	1.10
	Denoised FCNs	-0.20	0.16	0.46	0.47	1.26	1.21	1.37	1.39	1.35	0.93	2.66	9.63	1.66	0.30	0.03	0.53	0.98
	LR-cGAN	**0.03**	**1.04**	**1.64**	**1.89**	**2.92**	**2.98**	**4.73**	**4.13**	**6.80**	**3.41**	**5.46**	**7.67**	**6.50**	**2.01**	**0.54**	**1.25**	**3.32**

We also show the rate of increase in the number of inlier matches over different local feature descriptors from the low-rank images learned from the three different networks benchmarked against the original image in Table. I. The conclusion is in consistence with Fig. 4.

D. Analysis and discussion

In our experiments, we perform feature matching on each image sequence using different descriptors that includes SIFT, SURF, BRISK and ORB. We found that the binary descriptors, i.e., BRISK and ORB, do not perform well in both the original and learned low-rank images. On the contrary, the performances of SIFT and SURF feature matching are greatly improved on the low-rank representations.

We can conclude that: (1) the performance of SIFT and SURF feature matching on the original images fall drastically when the time different is larger than 2-4 hours. (2) SIFT and SURF feature matching on the low-rank images from the deep networks achieve better results than the original images over all the time differences. (3) Our LR-cGAN has the highest number of inlier matches compared with FCN-8s and original images. This is an expected result since our LR-cGAN learn really high quality low-rank images from the original images. (4) Although the predictions from FCN-8s are blurry and noisy, they still outperform the SIFT and SURF feature matchings on the original image.

Although matching of hand-crafted descriptors across images of changing scenes, we show in our experiments that

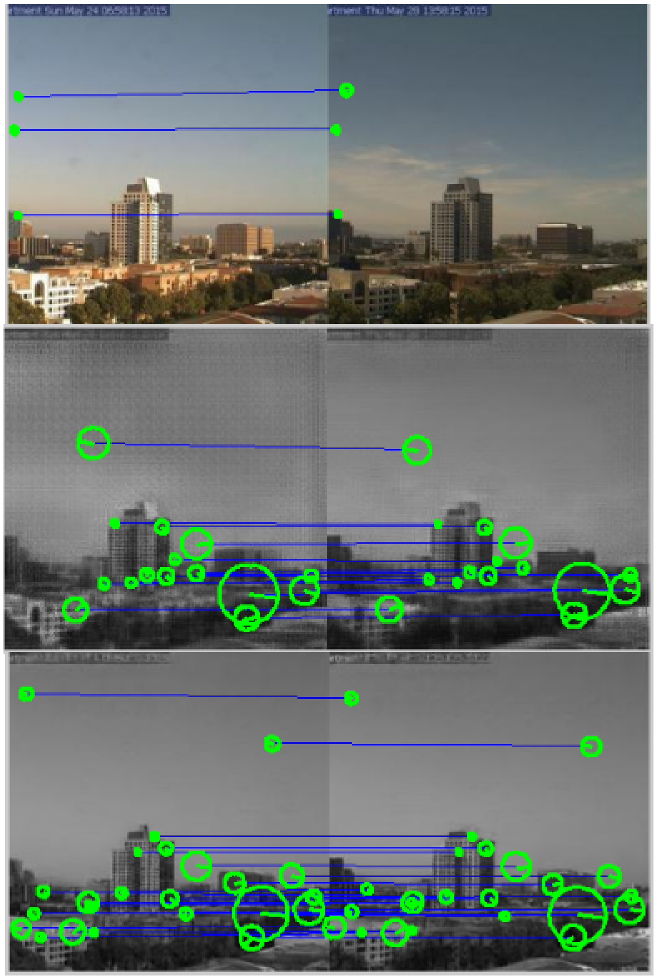

Fig. 5. An example of feature matching results from two images taken at 5am (left) and 6pm (right). SIFT matches on the (top row) original images, (second row) low-rank images from FCN-8s and (bottom row) low-rank images from our LR-cGAN.

matching performances can be boosted by removing the variations in the image with the low-rank image learned from deep learning.

V. Conclusion

In this paper, we proposed a new approach to enhance long-term image feature matching under challenging outdoor environments. We show that the performance of feature extraction and matching on the low-rank representations that we learned from deep networks improves a lot compared with directly matching on original images. In the future work, we like to extend the approach to learning low-rank images for robust feature matching under different viewpoints instead of static cameras.

VI. ACKNOWLEDGMENT

This work is supported in part by a Singapore MOE Tier 1 grant R-252-000-A65-114. We are grateful for the support.

References

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, “Orb-slam: a versatile and accurate monocular slam system,” *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[2] J. Engel, T. Schöps, and D. Cremers, “Lsd-slam: Large-scale direct monocular slam,” in *European Conference on Computer Vision*, pp. 834–849, Springer, 2014.

[3] J. Engel, V. Koltun, and D. Cremers, “Direct sparse odometry,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2018.

[4] Y. Verdie, K. M. Yi, P. Fua, and V. Lepetit, “TILDE: A temporally invariant learned detector,” *CoRR*, vol. abs/1411.4568, 2014.

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

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

[7] S. Leutenegger, M. Chli, and R. Y. Siegwart, “Brisk: Binary robust invariant scalable keypoints,” in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 2548–2555, IEEE, 2011.

[8] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg, “Matchnet: Unifying feature and metric learning for patch-based matching,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3279–3286, 2015.

[9] N. Carlevaris-Bianco and R. M. Eustice, “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.

[10] R. P. Nathan Jacobs, Nathaniel Roman, “Consistent temporal variations in many outdoor scenes,” in *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[11] M. Paulin, M. Douze, Z. Harchaoui, J. Mairal, F. Perronin, and C. Schmid, “Local convolutional features with unsupervised training for image retrieval,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 91–99, 2015.

[12] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, “Lift: Learned invariant feature transform,” in *European Conference on Computer Vision*, pp. 467–483, Springer, 2016.

[13] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, 2015.

[14] K. Chen, Papandreou and A. L. Yuille, “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs,” *CoRR*, vol. abs/1606.00915, 2016.

[15] F. Liu, C. Shen, and G. Lin, “Deep convolutional neural fields for depth estimation from a single image,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5162–5170, 2015.

[16] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” *arXiv preprint arXiv:1511.06434*, 2015.

[17] I. Zhang and A. A. Efros, “Colorful image colorization,” *CoRR*, vol. abs/1603.08511, 2016.

[18] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” *arxiv*, 2016.

[19] A. Aravkin, S. Becker, V. Cevher, and P. Olsen, “A variational approach to stable principal component pursuit,” in *Conference on Uncertainty in Artificial Intelligence (UAI)*, July 2014.

[20] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, “Autoencoding beyond pixels using a learned similarity metric,” *arXiv preprint arXiv:1512.09300*, 2015.

[21] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, “Conditional random fields as recurrent neural networks,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1529–1537, 2015.

[22] R. Collobert, K. Kavukcuoglu, and C. Farabet, “Torch7: A matlab-like environment for machine learning,” in *BigLearn, NIPS Workshop*, no. EPFL-CONF-192376, 2011.

[23] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” in *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 675–678, ACM, 2014.