Image Reconstruction from Local Descriptors Using Conditional Adversarial Networks

Haiwei Wu, Jiantao Zhou and Yuanman Li
Department of Computer and Information Science
University of Macau, Macau, China
E-mails: mb85403@um.edu.mo, jtzhou@um.edu.mo, yuanmanx.li@gmail.com

***Abstract*—Many applications rely on the local descriptors extracted around a collection of interest points. Recently, the security of local descriptors has been attracting increasing attention. In this paper, we study the possibility of image reconstruction from these descriptors, and propose a coarse-to-fine framework for the image reconstruction. By resorting to our gradually reconstructing network architecture, the novel multi-scale feature map generation algorithm, and the strategically designed loss functions, our proposed algorithm can recover the images with very high perceptual quality, even partial descriptors are provided only. Extensive experimental results are reported to show its superiority over the existing algorithms. Our study implies that the local descriptors contain surprisingly rich information of the original image. Users should pay more attention to sensitive information leakage when using local descriptors.**

I. Introduction

Interest points of an image refer to those pixels with prominent characteristics, which generally have many desirable good properties, such as rotation invariance, robustness against illumination changes etc. Finding interest points plays an essential role in a wide range of feature extraction algorithms. Among different types of local feature extraction approaches, Scale Invariant Feature Transform (SIFT) [1] is one of the most popular one and has been extensively investigated from various perspectives. For each interest point, a corresponding local descriptor can be generated by encoding its surrounding information. It has been demonstrated that many local descriptors, including SIFT ones, are of strong discriminability, and excellently robust against geometrical transforms and various kinds of noises. Local descriptors have been widely employed in many existing high level vision tasks, e.g., image recognition [2], image matching [3] and visual tracking [4].

Due to their popularity, the privacy and security issues regarding to the local descriptors have been attracting increasing attention. For example, some recent studies demonstrated that the SIFT features can be maliciously removed and forged, making those decisions from SIFT-based systems untrustworthy [6], [7]. In this work, we consider another security scenario, where the local descriptors may be eavesdropped by malicious attackers in an insecure channel. Take the Content Based Image Retrieval (CBIR) system as an example [8]. Users query interested images from one CBIR system by simply passing the local descriptors through a public network. The local descriptors then can be readily exposed to malicious attackers when the channel is eavesdropped. Though the descriptors extracted from the image only provide a summary of its visual characteristics rather than its most informative fragments, a pirate can still use them to interpret the image content, and potentially causing sensitive information leakage.

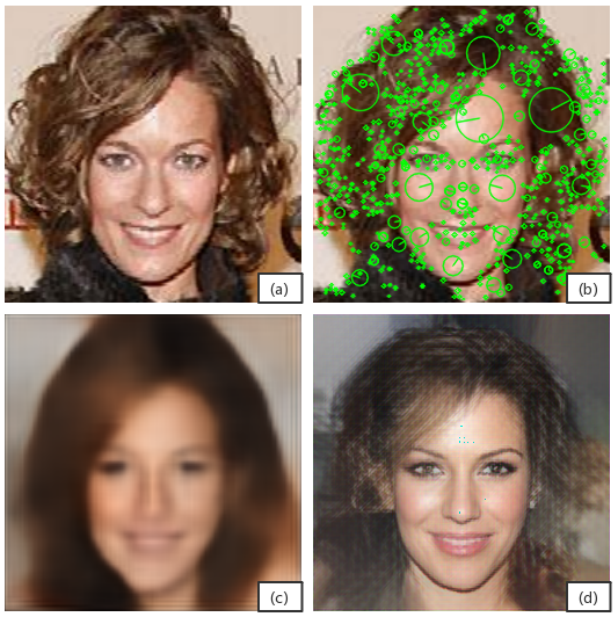


Fig. 1: Reconstructed images of different methods from SIFT descriptors. (a) original image, (b) SIFT descriptors, (c) result of [5], and (d) ours.

In order to evaluate the risk of the information leakage from local descriptors, it is necessary to know what kind of information, and how much information are carried by the local descriptors. A natural idea to this problem is to investigate how much the latent content can be recovered through the local descriptors. Along this line, several approaches have been devised to reconstruct the images from local descriptors [9]–[14]. The pioneer study on this problem was conducted by Weinzaepfel *et al.* [9], who attempted to reconstruct the image from its SIFT descriptors. Specifically, they first built up a large patch database, then restored the image by simply pasting and smoothing those searched patches with similar descriptors. Further, with a certain degree of the user interaction, the color information of the image could also be reproduced to some extent. However, due the limited number of descriptors in one

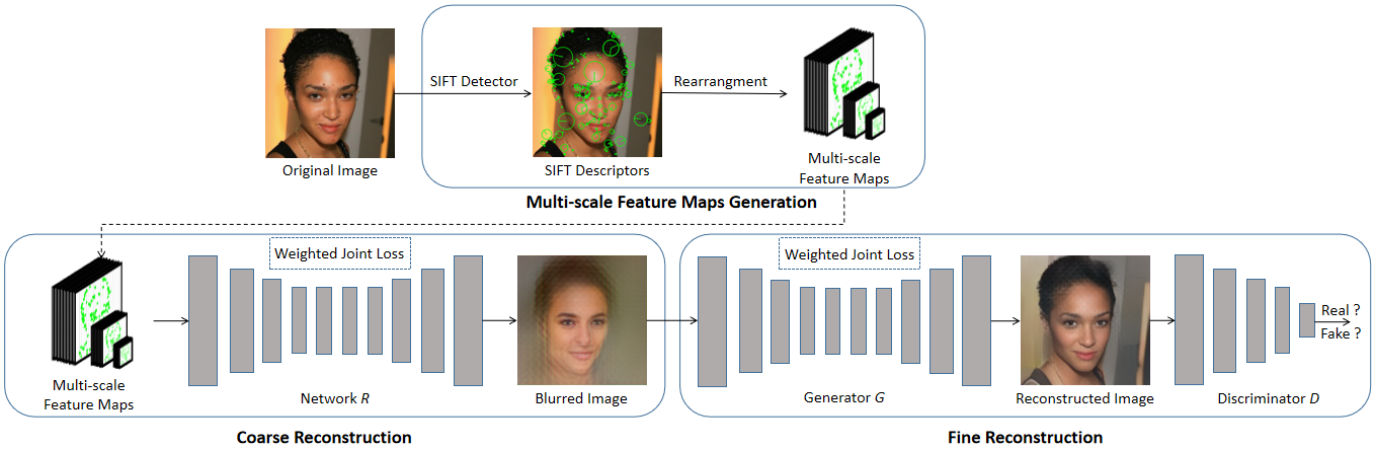


Fig. 2: The proposed coarse-to-fine framework. Training stage: after extracting the SIFT descriptors from the original image, the pre-processing is performed to generate multi-scale feature maps. The network R is pre-trained to generate blurred images, and finally a conditional adversarial network consisting of a generator G and a discriminator D, is concatenated to R and perform an end-to-end training in terms of the designed weighted joint loss. Testing stage: feed the generated feature maps to R and the generator G outputs the reconstructed image.

image, the algorithm [9] can only reveal some sharp structures rather than the fine textures. By analyzing the property of the local binary descriptors, Angelo *et al.* [10] proposed an inversion algorithm tailored for the local binary descriptors without using any external databases. Vondrick *et al.* [11] addressed the problem of image reconstruction from the histograms of gradient orientations (HOG) descriptors by using the dictionary representation. Through estimating the spatial arrangement of local descriptors over a large-scale image database, Kato *et al.* [12] presented a method to reconstruct the image from its Bag-of-Visual-Words (BoVW). Desolneux *et al.* [13] devised two reconstruction models for local HOG features. By adopting the Poisson editing, their methods can recover global shapes and many geometric details of the images without requiring any external databases. Recently, some researchers also addressed the reconstruction problem based on the neural networks, capitalizing on their powerful representation capacity [5], [14]. Specifically, Dosovitskiy *et al.* [5] proposed a reconstruction approach for several types of local descriptors through a designed up-convolutional neural network. Although the reconstructed images obtained by [5] are substantially better than those of previous approaches, they are very blurry and most of details are still missing.

Considering the drawbacks of the existing methods, in this paper, we propose a new end-to-end face reconstruction model from local descriptors based on the conditional adversarial networks. Due to the space limit, SIFT descriptors are adopted in our work. However, all the results can be readily extended to other descriptors. As illustrated in Fig. 2, our framework consists of three phases: 1) the multi-scale feature maps generation; 2) the coarse reconstruction network; and 3) the fine reconstruction network. First, the descriptors are pre-processed to generate a set of feature maps at multiple scales. Then, the feature maps are fed into the coarse reconstruction network, which is pre-trained over a large database. As a well-known fact, the local descriptor and the image patch do not obey the one-to-one mapping, i.e., different patches could have the same descriptors. This makes the coarse reconstruction network often produce blurry outputs with global structures. To reveal the fine details, we further propose to concatenate the coarse reconstruction network with an appropriately designed conditional adversarial network (named fine reconstruction network), which takes the output of the coarse reconstruction network as the input, and aims to recover a more realistic image. Furthermore, in order to generate more visually pleasing images, we strategically design two loss functions for the coarse and fine reconstruction networks. As shown in Fig. 1, most of the facial details can be recovered by our approach, even in those areas where no SIFT keypoint exists. Compared with the other algorithms, our reconstructed image achieves substantially better perceptual quality. More results will be given in the experimental stage.

The rest of this paper is organized as follows. In Section II, we briefly review the conditional adversarial networks. Section III discusses our proposed reconstruction algorithm. Extensively experimental results are reported in Section IV, and we finally conclude in Section V.

II. Review of Conditional Adversarial Networks

Generative adversarial networks (GANs) proposed in [15] define a minimax game between two competing networks, i.e., the generator G and the discriminator D. The generator $G : z \rightarrow y$ takes a random noise z as input, and then generates a sample y. The discriminator D tries to distinguish the real samples and generated samples. During training, the generator G and the discriminator D are competing with each other, finally making the generator G to produce samples *indistinguishable* from the real ones. In addition, when the generator and discriminator are conditioned on some extra information

x, then GANs are extended to the conditional generative adversarial networks (cGANs) [16]. The extra information x could be any kind of auxiliary information, e.g., the class label or an image. The cGANs are driven by feeding x into G and D as additional input layers. Typically, the objective function of cGANs can be expressed as

$$L_{cGAN}(G,D) = \mathbb{E}_{x,y}[\log D(x,y)] + \mathbb{E}_{x,z}[\log(1 - D(x, G(x,z)))], \quad (1)$$

where $G^* = \arg\min_G \max_D L_{cGAN}(G,D)$. Note that the network could still learn a mapping from x to y without z, but would produce deterministic outputs. In recent years, cGANs have been widely studied in many vision tasks, e.g., image-to-image translation [17]. For more details about cGANs, please refer to [16].

III. Proposed Method

With only the SIFT descriptors extracted from the original image I_o, the proposed method aims to reconstruct an image I_r, which should be similar to I_o. As shown in Fig. 2, our framework works in a coarse-to-fine manner, and contains three phases: 1) multi-scale feature maps generation; 2) coarse image reconstruction, and 3) fine image reconstruction. All the details regarding to these three components will be given in the following subsections.

A. Multi-scale Feature Maps Generation

Note that the SIFT descriptors are a collection of vectors, and the number of descriptors are also highly varied from different images. This makes it impossible to directly feed the descriptors into a network for training. In the first stage, we propose to rearrange the SIFT descriptors of an image as a set of feature maps, which can accommodate the input of the coarse image reconstruction component.

The original SIFT detector works only for grayscale images. For each color image with RGB channels, we first transform it into the grayscale domain by using a function $T = (\Gamma(R) + \Gamma(G) + \Gamma(B))/3$, where $\Gamma(t) = t^{1/2.2}$ is standard gamma correction function [19]; R, G and B respectively denote the red, blue and green channels. For each corresponding grayscale image, SIFT algorithm is applied to generate a set of keypoints $\{\mathbf{k}_1, ..., \mathbf{k}_n\}$ and their corresponding descriptors $\{\mathbf{v}_1, ..., \mathbf{v}_n\}$, where each descriptor is an 128-dimensional vector. For the i-th keypoint, $\mathbf{k}_i$ is represented as a four dimensional vector

$$\mathbf{k}_i = (x_i, y_i, \sigma_i, \theta_i). \quad (2)$$

Here, (x_i, y_i) are the coordinates in the image plane, σ_i serves as the scale, and θ_i is its dominant orientation.

For each image, we then generate the feature maps by rearranging its descriptors in different manners. Note that the SIFT descriptor $\mathbf{v}_i$ contains 16 histograms with 8 bins, where each histogram encodes the information on the 4×4 pixel neighborhoods. This inspires us to restore a small local area centered at each keypoint according to its descriptor. Similar to [5], we propose to divide the image into cells of the size $d \times d$ ($d = 4$), which yields totally $\lceil W/d \rceil \times \lceil H/d \rceil$ cells. Then each cell is assigned by a descriptor $\mathbf{v}_i$ based on its coordinates (x_i, y_i). Empty vectors are assigned to those cells without any keypoints. In some cases, there may exist more than one descriptors in a single cell, then we propose to place the additional descriptors in the adjacent empty cells.

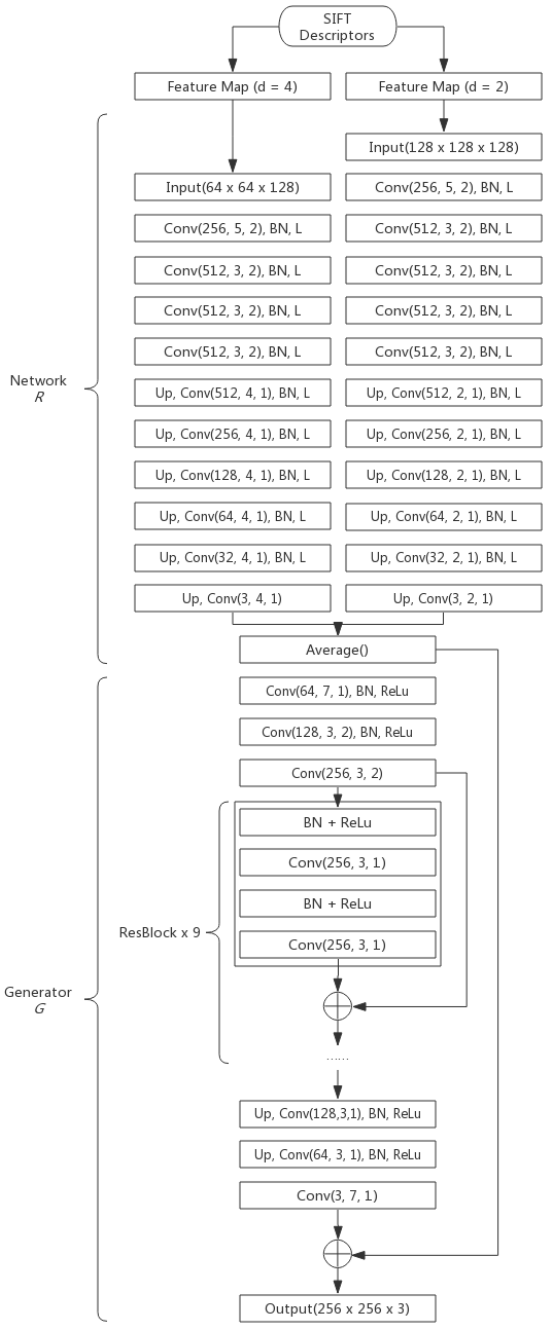


Fig. 3: Proposed architectures of R and G. Conv(α, β, γ) means a convolution layer with α filters, kernel size β and stride γ. Up denotes up-sampling operator. BN serves as the batch normalization. ReLu represents the rectified activation and L is LeakyReLu.

Using above strategy, we finally can generate a feature map $F_d \in \mathbb{R}^{\lceil W/d \rceil \times \lceil H/d \rceil \times 128}$ for each image.

It should be noted that the number of features is fixed for a given image; then a small d will lead the descriptors to distribute very sparsely in the feature map. This makes the training of the network unstable since most of the cells are empty vectors. On the other hand, a large d will let many descriptors highly clustered, potentially losing fine details due to the interactions among different descriptors. To tackle this problem, we propose a multi-scale feature maps generation strategy as shown in Fig. 2. Specifically, with different settings of d, we can generate a set of feature maps at different scales. For simplicity, denote $F_{d=x}$ as the feature map resulted by setting $d = x$. In our experiments, we find that using feature maps generated by $d = 4$ and $d = 2$ can well balance the performance and model complexity. Then the multi-scale feature maps can be represented as

$$F = \{F_{d=2}, F_{d=4}\}. \tag{3}$$

As will be clear in the experimental stage, our proposed multi-scale feature maps generation strategy can greatly improve the quality of the reconstructed images.

B. Coarse Reconstruction Network

The architecture of our proposed coarse reconstruction network R is illustrated in Fig. 3. As can be seen, the network R consists of two subnetworks, which take the feature maps at different scales as the input. The outputs of these two subnetworks are averaged in the final layer to generate an intermediate image of the same size as the original one. The left subnetwork has 10 convolution layers. The beginning 4 convolution layers with stride 2 are designed to extract both the local and global information of the feature map. Then 6 deconvolution layers are concatenated to interpret the information as an image of the same size as the original one. In our experiment, we adopt an up-sampling layer followed by a convolution layer with stride 1 to implement deconvolution operations. The number of kernels and the kernel size of each layer list in Fig. 3 are carefully tuned to achieve the best performance. Each convolution layer (except the last one) is followed by a BatchNormalization layer [29] and the LeakyReLu activation layer [30] with $\alpha = 0.2$. Similarly, we design the right subnetwork with 11 convolution layers, which takes the feature map with $d = 2$ as its input.

The loss function for R can be naturally defined in a ℓ_2 sense. Mathematically,

$$L_{2,R} = \mathbb{E}\Big[||I_o - R(F)||^2\Big], \tag{4}$$

where the expectation is calculated over all the training images $\{I_o\}$, and F is the collection of feature maps, which is defined in (3). We experimentally find that the ℓ_2 loss often causes the results highly blurred, thus losing many details. Fig. 4 shows an example, where we can observe that ℓ_2 loss fails to restore the details in the facial region. To tackle this problem, in addition to ℓ_2 loss, we further introduce two losses, i.e., perceptual loss [22] and style loss [23].

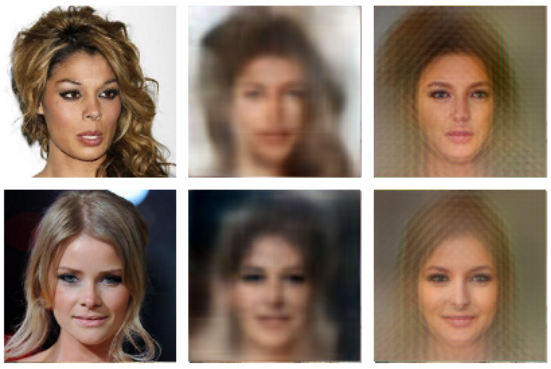


Fig. 4: Results of the network R with different loss functions. From left to right: original image, ℓ_2 loss function and our proposed loss function.

As the name suggests, perceptual loss penalizes results that are not perceptually similar to original image, which can be defined as

$$L_{p,R} = \mathbb{E}\Big[||\varphi(I_o) - \varphi(R(F))||^2\Big], \tag{5}$$

where φ in our work is chosen as the activation map of the 3-th layer of the pre-trained VGG16 network on ImageNet. On the other hand, the style loss is used to measure the differences between covariances of the activation maps, which is an effective strategy to eliminate "checkerboard" artifacts cased by deconvolution layers [24]. Typically, the style loss can be defined as

$$L_{s,R} = \mathbb{E}\Big[||M^{\varphi}(R(F)) - M^{\varphi}(I_o)||_1\Big], \tag{6}$$

where M^{φ} is a $C \times C$ Gram matrix constructed from the activation map φ.

Finally, we define the loss function for R as the combination of above three losses

$$L_R = L_{2,R} + \lambda_p L_{p,R} + \lambda_s L_{s,R}. \tag{7}$$

In our paper, we call above loss as the weighted joint loss, where we empirically set $\lambda_p = 1$ and $\lambda_s = 1e^{-4}$.

As shown in Fig. 4, our proposed weighted joint loss substantially improves the reconstruction results of the network R. Compared with the traditional ℓ_2 loss, the proposed loss function helps to reveal much more fine structures.

C. Fine Reconstruction Network

We build the fine reconstruction network as a conditional adversarial network, which consists of two components: a generator G and a discriminator D. For the generator G, we adopt a similar structure design as proposed in [25], which contains 2 convolution blocks with stride 2, 9 ResBlock [26] and 2 deconvolution blocks. Each ResBlock is comprised of a normalization layer, ReLu [27] activation and a convolution layer. Dropout [28] regularization is adopted with a probability 0.5 in every ResBlock to prevent overfitting. A global skip connection introduced in [25] is employed to make the training faster. The whole architecture of G is shown in the bottom

Fig. 5: Reconstruction results of different algorithms. First row: original images. Second row: results of [5]. Bottom row: results of ours.

of Fig. 3. For the discriminator D, we simply adopt a same architecture as proposed in [17]. Interested readers are invited to see [17] for more details.

For the loss function of G and D, besides the adversarial loss defined in (1), we also add the ℓ_2 loss, perceptual loss and style loss, which can be similarly defined as (4), (5) and (6), respectively. The final loss function of G and D then can be written as

$$L_{GD} = L_{cGAN} + \lambda_{\ell_2} L_{2,G} + \lambda_p L_{p,G} + \lambda_s L_{s,G}. \quad (8)$$

In our experiment, we set $\lambda_{\ell_2} = \lambda_p = 100$ and $\lambda_s = 0.01$.

D. Model Training and Inference

Our model is implemented using the Keras deep learning framework [31]. The training procedure is performed on a desktop equipped with a Core-i7 and a single GTX 1080 GPU. To stabilize the training process and alleviate the gradient vanishing problem, we first train the network R on the training set until convergence. Then we concatenate fine reconstruction module to R, and perform an end-to-end training over R, G and D simultaneously. Adam algorithm [32] is adopted in optimization, where we set $\beta_1 = 0.9$, $\beta_2 = 0.999$, the learning rate $r = 10^{-4}$, and the batch size is equal to 8. We perform 5 times weights updates on D, then one on R and G. Our model reaches convergence after about 200 training epochs.

At the inference stage, we discard the discriminator D in the Fig. 2. Given the descriptors of one image, the generator G directly outputs the predicted image.

IV. Experiments

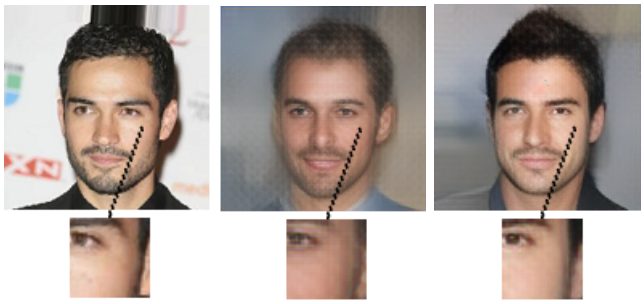
This section provides extensive experimental results to evaluate our proposed reconstruction framework.

A. Datasets

The CelebFaces Attributes Dataset (CelebA, [33]) is adopted in our experiment, which is a large-scale face attributes dataset containing over 200K celebrity images. We randomly select 8,000 facial images for training and another 1,000 images of different identities are selected for testing. We experimentally find that more training images makes little improvement on the final results. For each selected image, we extract the largest square area at the center, and then rescale it to the size 256×256.

B. Reconstruction Results

Fig. 5 depicts the reconstruction results of five images. The reconstructed images obtained by the algorithm [5] are also reported for comparison. Note that the method [5] is also based on deep networks. Due to the space limit, we do not compare with other previous approaches, such as [9], [13]. However, readers are invited to see the results shown in their papers, where the reconstructed images are far from real images. As can been seen from Fig. 5, our method can recover most of details. Compared with [5], the proposed algorithm substantially improves the reconstructed images, including both the facial areas and those highly textured regions (e.g., hair and beard). Due to the powerful learning ability of the conditional adversarial networks, we can also note that the colors of skin are also restored to some extent through our



(a) (b) (c)

Fig. 6: Reconstruction results of different architectures. (a): original image, (b) result obtained with a single feature map ($d = 4$), and (c) result obtained by our proposed architectures.

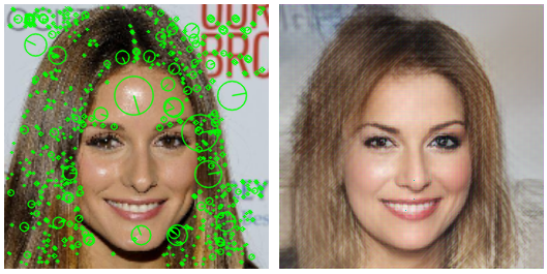


Fig. 7: Image reconstruction from partial descriptors. Left: SIFT descriptors (the middle ones are removed manually). Right: reconstruction result.

Fig. 8: Reconstruction results using 8 categories from ImageNet. First row: original images, and second row: reconstructed images.

framework. Our results demonstrate that the SIFT descriptors contain surprisingly rich information of the original image.

To discuss the benefit of our proposed multi-scale feature maps generation strategy and the parallel architecture design of the network R, we also report the results obtained based on a single feature map. The image shown in Fig. 6 (b) is obtained by using a single feature map with $d = 4$. We can observe that Fig. 6 (b) introduces many blurred artifacts, and the reconstructed image is much worse than that of our proposed method (shown in Fig. 6 (c)). This demonstrates that our proposed multi-scale architecture indeed helps for image reconstruction.

In addition, we also evaluate the robustness of our proposed framework. In some cases, the attacker may only obtain a part of descriptors. Unfortunately, even under this scenario, most of the information could still be potentially recovered. One example is shown in Fig. 7, where we manually remove the descriptors in a local region of the face. We can see that many details in that region can still be reconstructed using only the remaining descriptors.

C. Limitations

Although our model generally can generate facial images with very high perceptual quality, we discuss some limitations in this subsection.

First, we experimentally find that the facial regions are recovered much better than the other areas, such as hair and background. The underlying reasons can be two-fold: 1) for those highly textured regions (e.g., hair), the descriptors may not enough to encode their texture information; 2) the conditional adversarial networks mainly focuses on the facial regions rather than the other areas. In this case, those generated images with blurred hair or background may not be rejected by the discriminator.

Second, the current model is tailored for facial images reconstruction. Extending our model to multiple categories is not straightforward. To demonstrate this problem, we train our framework using 8 categories with totally 8,000 images from ImageNet dataset. Fig. 8 shows that the reconstructed images are very blurry. We think that this phenomenon is caused by the following two aspects: 1) the number of training images for each category is not enough; 2) currently, the discriminator is designed to distinguish the generated images from real ones, totally ignoring their categories. One potential solution is to re-design the discriminator, forcing the generator to produce an image of a certain category. This can be an interesting problem that we will investigate in the future.

V. Conclusions

In this paper, we have proposed a novel end-to-end face reconstruction model from local descriptors based on the conditional adversarial networks. Our model works in a coarse-to-fine manner. By resorting to the well designed multi-scale feature maps generation algorithm and the conditional adversarial networks, our approach has substantially improved the reconstruction results compared with existing ones. Extensive experimental results are provided to demonstrate its superiority. An implication of our study is that users should pay more attention on the privacy issues when using local descriptors, as they contain surprisingly rich information of the original image. If the local descriptors (even a part of them) are obtained by illegal users, the sensitive information can be leaked in a high probability.

Acknowledgments: This work was supported in part by the Macau Science and Technology Development Fund under Grants FDCT/022/2017/A1 and FDCT/077/2018/A2, and in part by the Research Committee at the University of Macau under Grant MYRG2016-00137-FST and MYRG2018-00029-FST.

REFERENCES

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

[2] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce, "3d object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints," *International Journal of Computer Vision*, vol. 66, no. 3, pp. 231–259, 2006.

[3] Y. Li and J. Zhou, "Fast and effective image copy-move forgery detection via hierarchical feature point matching," *IEEE Trans. on Inf. Forensics and Security*, vol. 14, no. 5, pp. 1307–1322, 2019.

[4] S. Gauglitz, T. Höllerer, and M. Turk, "Evaluation of interest point detectors and feature descriptors for visual tracking," *International journal of computer vision*, vol. 94, no. 3, p. 335, 2011.

[5] A. Dosovitskiy and T. Brox, "Inverting visual representations with convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4829–4837.

[6] Y. Li, J. Zhou, and A. Cheng, "SIFT keypoint removal via directed graph construction for color images," *IEEE Trans. on Inf. Forensics and Security*, vol. 12, no. 12, pp. 2971–2985, 2017.

[7] Y. Li, J. Zhou, A. Cheng, X. Liu, and Y. Y. Tang, "SIFT keypoint removal and injection via convex relaxation," *IEEE Trans. on Inf. Forensics and Security*, vol. 11, no. 8, pp. 1722–1735, 2016.

[8] H. Lejsek, F. Asmundsson, B. Jonsson, and L. Amsaleg, "NV-tree: An efficient disk-based index for approximate search in very large high-dimensional collections," *IEEE Trans. on Pattern Anal. and Mach. Intell.*, vol. 31, no. 5, pp. 869–883, 2009.

[9] P. Weinzaepfel, H. Jégou, and P. Pérez, "Reconstructing an image from its local descriptors," in *CVPR 2011*. IEEE, 2011, pp. 337–344.

[10] E. d'Angelo, A. Alahi, and P. Vandergheynst, "Beyond bits: Reconstructing images from local binary descriptors," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*. IEEE, 2012, pp. 935–938.

[11] C. Vondrick, A. Khosla, T. Malisiewicz, and A. Torralba, "Hoggles: Visualizing object detection features," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1–8.

[12] H. Kato and T. Harada, "Image reconstruction from bag-of-visual-words," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 955–962.

[13] A. Desolneux and A. Leclaire, "Stochastic image reconstruction from local histograms of gradient orientation," in *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer, 2017, pp. 133–145.

[14] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5188–5196.

[15] 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.

[16] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[17] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[18] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2536–2544.

[19] W. Hoeffding, "Probability inequalities for sums of bounded random variables," in *The Collected Works of Wassily Hoeffding*. Springer, 1994, pp. 409–426.

[20] C. Kanan and G. W. Cottrell, "Color-to-grayscale: does the method matter in image recognition?" *PloS one*, vol. 7, no. 1, p. e29740, 2012.

[21] T. G. Dietterich, "Ensemble methods in machine learning," in *International workshop on multiple classifier systems*. Springer, 2000, pp. 1–15.

[22] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*. Springer, 2016, pp. 694–711.

[23] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2414–2423.

[24] M. S. Sajjadi, B. Scholkopf, and M. Hirsch, "Enhancenet: Single image super-resolution through automated texture synthesis," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4491–4500.

[25] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, "Deblurgan: Blind motion deblurring using conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8183–8192.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European conference on computer vision*. Springer, 2016, pp. 630–645.

[27] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814.

[28] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[29] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[30] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," *arXiv preprint arXiv:1505.00853*, 2015.

[31] F. Chollet *et al.*, "Keras," 2015.

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[33] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3730–3738.