Automatic Image Colorization based on Multi-Discriminators Generative Adversarial Networks

Youssef Mourchid, Marc Donias, Yannick Berthoumieu
University of Bordeaux, CNRS, IMS, UMR 5218, Signal and Image Group, F-33405 Talence, France
e-mail: {youssef.mourchid, marc.donias, yannick.berthoumieu}@ims-bordeaux.fr

***Abstract*—This paper presents a deep automatic colorization approach which avoids any manual intervention. Recently Generative Adversarial Network (GANs) approaches have proven their effectiveness for image colorization tasks. Inspired by GANs methods, we propose a novel colorization model that produces more realistic quality results. The model employs an additional discriminator which works in the feature domain. Using a feature discriminator, our generator produces structural high-frequency features instead of noisy artifacts. To achieve the required level of details in the colorization process, we incorporate non-adversarial losses from recent image style transfer techniques. Besides, the generator architecture follows the general shape of U-Net, to transfer information more effectively between distant layers. The performance of the proposed model was evaluated quantitatively as well as qualitatively with places365 dataset. Results show that the proposed model achieves more realistic colors with less artifacts compared to the state-of-the-art approaches.**

I. Introduction

Nowadays image colorization becomes an active area of research in machine learning. This is due to the variety of applications such as style transfer, image translation, *etc*. There are many proposed methods in the literature to tackle down the ill-posed nature of the colorization problem. With the fast development in deep learning representation, traditional techniques have largely be replaced. Most of the work done in image colorization may be classified into two broad approaches: guided and non-guided colorization. The guided colorization [1]–[3] requires user interaction or an example image for providing prior about colorization. In the non-guided colorization [4]–[7], automatic colorization algorithms are used without any prior.

For automatic colorization task, deep learning methods have emerged recently. Deep neural network approaches are able to train a model by feeding large-scale datasets to learn a parametric generation between the grayscale images and their multispectral counterparts.

In [7], Zhang *et al.* proposed to train a convolutional neural network architecture to minimize the multinomial cross-entropy loss for color distribution prediction. For optimal colorization results, the network is initialized with a classification-based network. In [8], authors promoted using unlabeled data for automatic colorization based on self-supervision. In the same time, Zhang *et al.* [9] explored the cross-channel encoders, which are modification of the traditional autoencoder architecture.

Recently, a new generation of learning structure called Generative Adversarial Networks (GANs) [10] was popularly employed to generate new data with the same statistics as the training set. GANs consist of two neural networks competing with each other: a generator and a discriminator respectively. The generator tries to fool the discriminator by generating realistic synthetic images, while the discriminator tries to distinguish generated fake images from real ones.

In the context of colorization, recent works using GAN shown a significant improvement in the perceptual quality of generated images over other deep-learning based models. Isola *et al.* [11] leveraged the power of conditional GANs, coupling a DCGAN [11] based on the U-net architecture for the generator definition and a Markovian definition for the discriminator, i.e. PatchGAN architecture. Nazeri *et al.* [12] proposed generative network-based on L1 loss in an architecture with skip connections, within a "U-Net" shape. However, they seem to be unable to take into account the colored details at higher scales and their performance is deteriorated.These previous works exhibit some colorization artifacts.

Considering the colorization task as a specific image reconstruction problem, it is interesting to consider other contributions based on GAN approach dealing with inverse problem. For super-resolution, Park *et al.* [13] are convinced that the pixel level discriminator only causes the generator to give rise to meaningless high-frequency noise. To deal with this problem, authors attach an additional level of comparison oriented image feature to the discriminator. They operate on high-level representations extracted by a pre-trained VGG network to capture more meaningful potential attributes of real target images.

In the context of style transfer [14] or super-resolution task [15], different works [16] shown the importance to learn the textured style content of images to increase the quality of the generator outputs. This is the reason why we would consider texture descriptors which can be employed as features to obviously improve the colorization results. Motivated by the limitation of previous GAN-based image colorization

This study has been carried out with financial support from the French Direction générale de l'armement (DGA) in the frame of the projet Man-Machine Teaming (MMT).

approaches, we present a novel automatic image colorization approach based on generative adversarial networks (GANs).

In view of the previous discussions, the key contributions of this work are:

- Deriving of a model for colorization that employs two discriminators to promote the restitution of fine-structures of the generated images. The first discriminator is dedicated to the pixel level, and the second one for image feature level.
- Training the generator with discriminators using non-adversarial perceptual losses based on style transfer component, in order to match the texture style of generated images with original ones.
- Employing U-net architecture to ensure the flow of low-level information inside the network using skip connections.

We first introduce in Section II the proposed approach. The performance of the method is discussed in Section III, Finally, we conclude in Section IV.

II. Proposed Method

Our goal is to generate a colorized image I_g from a given grayscale image I_{gray}, which looks much similar to the original one (RGB), with a perceptually pleasant content. To achieve colorization, we propose a new GAN architecture exploiting texture style loss in both discriminator and generator. The architecture details of the proposed model are presented in the next section.

A. Architecture of the proposed model

Conventional generative adversarial networks consists of the association of a generator G(.) and a discriminator D(.). The GAN framework tries to solve the minimax problem which is defined as follows:

$$\min_G \max_D \ (\mathbb{E}_{y\sim p_{data}(y)}[\log(D(y))]+\mathbb{E}_{x\sim p_x(x)}[\log(1-D(G(x)))]) \tag{1}$$

where G(x) denotes the output of a generator network for a given x data, D refers to the discriminator network, y is a sample data from a real distribution and x is a random noise. We further denote data distributions as $y \sim pdata(y)$ and $x \sim pdata(x)$. Both G and D are implemented respectively by two neural networks.

Different from conventional GAN network architecture, Park *et al.* [13] introduce architecture based on combination of one generator associated with two discriminators. For colorization task, we propose an extended model, illustrated in Fig.1, which uses two discriminators: an image discriminator D_i and a feature discriminator D_f. The first one discriminates real images (RGB) from colorized images by inspecting their pixel values, while the second discriminates real images from colorized ones by inspecting their feature maps, noted respectively $VGG(y)$ and $VGG(G(x))$. These maps correspond to high-level feature VGG domain [17]. The purpose is to help the generator to cover more meaningful high-frequency details in the training, and to achieve a more spatially stable colorization results. The architecture of our discriminators is developed to be similar to [18] but modified to suit our colorization task. The proposed model employs non-adversarial losses such as content and texture style ones, to ensure that generated images match the target images. Regarding the feature extractor to compute these two losses, a pre-trained VGG19 was used [17] as for the discriminator. The U-Net architecture is employed in our generator which allows low-level information to shortcut across distant layers in the network, based on the performance of skip-connections features.

B. Conventional Loss function

Considering previous colorization work [12], we consider a l_1-loss for pixel level. The l_1-loss is minimized by measuring the distance between the generated image $G(x)$ and the real image y. l_1-loss can be written as:

$$L_{l1} = \mathbb{E}_{x,y}||y-G(x)||_1 \tag{2}$$

where $G(x)$ and y have the same size.

C. Proposed Loss functions

GAN loss $L_{M-Dis}(G,D_i,D_f)$: Based on the above loss functions, we trained the generator with the two discriminators by introducing a loss function that takes the form:

$$L_{M-Dis}(G,D_i,D_f) = \lambda_i L_{GAN}(G,D_i) + \lambda_f L_{GAN}(G,D_f) \tag{3}$$

where λ_i and λ_f denote a defined weighting factors, $L_{GAN}(G,D_i)$ refers to pixel GAN loss which represents high-frequency details in the pixel domain, $L_{GAN}(G,D_f)$ is the features GAN loss that characterizes the structural details in the feature domain. The pixel GAN loss can be defined as follows:

$$\begin{aligned} L_{GAN}(G,D_i) &= \mathop{\mathbb{E}}_{y\sim pdata(y)}[\log(D_i(y)] \\ &+ \mathop{\mathbb{E}}_{x\sim pdata(x)}[\log(1-D_i(G(x))] \end{aligned} \tag{4}$$

To compute the feature GAN loss, we used the height level feature map of the VGG network [17]:

$$\begin{aligned} L_{GAN}(G,D_f) &= \mathop{\mathbb{E}}_{y\sim pdata(y)}[\log(D_f(VGG(y))] \\ &+ \mathop{\mathbb{E}}_{x\sim pdata(x)}[\log(1-D_f(VGG(G(x)))] \end{aligned} \tag{5}$$

where $VGG_l(.)$ denotes the feature maps of a specific layer l in VGG. Since the abstracted image structures can be represented by image features, using $L_{GAN}(G,D_f)$ pushes the generator to build realistic structural high-frequency instead of noisy artifacts in the generated images.

VGG loss L_{VGG}: In addition to the above losses, we suggest a non-adversarial losses from recent style methods. The texture loss $L_{texture}$ is used to penalize the discrepancy in the texture representations between the generated image $G(x)$ and its corresponding original image y. The texture loss

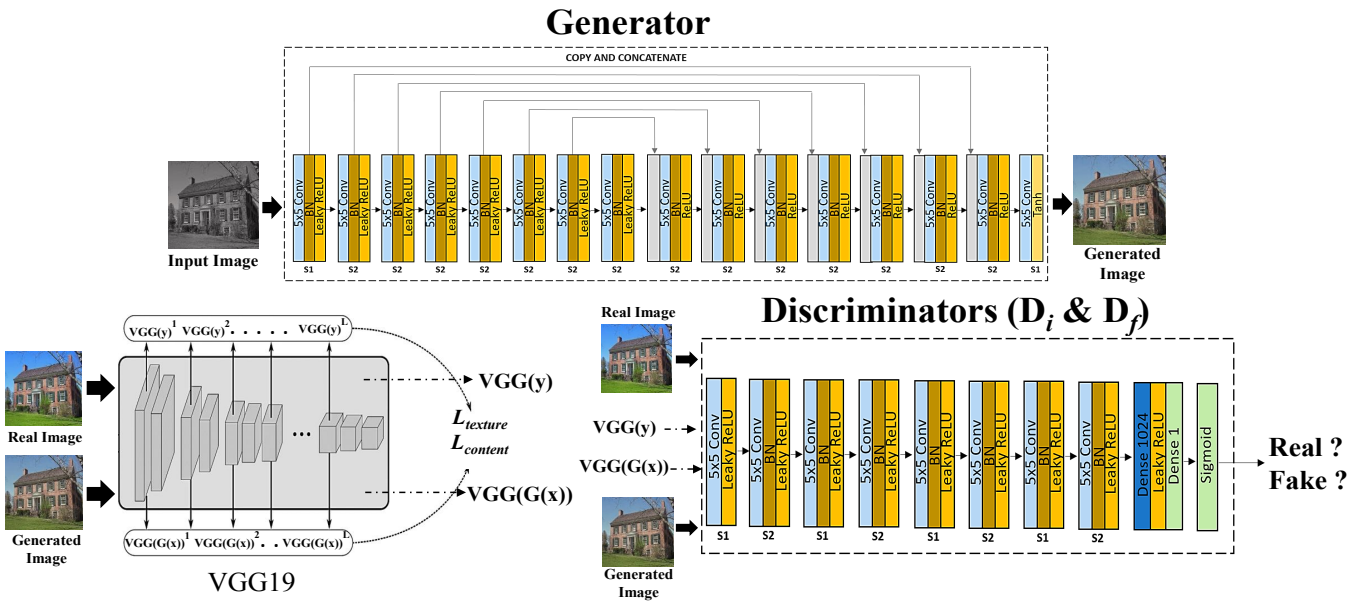

Fig. 1: Figure is better seen zoomed on the digital version of this document. Architecture of the proposed model, we employ two discriminators, an image discriminator in the pixel domain and a feature discriminator in neural space domain. $VGG(y)$ and $VGG(G(x))$ refer to the extracted features of a pre-trained VGG-19, for real image and generated image respectively.

can be obtained by computing correlations between feature representations in the spatial extent. These feature correlations are obtained using the Gram Matrix as defined in [14]. We can define the texture loss as follows:

$$L_{texture} = \sum_{l=1}^{L} \lambda_l^l \frac{1}{4N_l^2 M_l^2} ||Gr(VGG_l(y)) - Gr(VGG_l(G(x)))||_F^2 \tag{6}$$

where $Gr(.)$ is the Gram Matrix, $||.||_F$ denotes the Frobenius norm, L is the number of layers at the VGG network, λ_t^l refers to weighting factors of each layer l, N_l denotes the number of features maps of size M_l at each layer, M_l is the size of the factorized feature map.

In the other hand, the content loss $L_{content}$ penalizes directly the difference between feature representations. Unlike the texture loss, $L_{content}$ does not capture discrepancies in texture, instead, it ensures global consistency of the generated images with original ones by enhancing low frequency components. We define the content loss by:

$$L_{content} = \frac{1}{2} \sum_{l=1}^{L} ||VGG_l(y) - VGG_l(G(x))||_F^2 \tag{7}$$

Our total VGG loss L_{VGG} can be defined as follow:

$$L_{VGG} = \lambda_T L_{texture} + \lambda_C L_{content} \tag{8}$$

where λ_T and λ_C are weighting factors for texture and content loss, respectively. In contrast to the VGG loss where the input image is transformed into sensitive representations, that become relatively invariant to its precise appearance, the features GAN loss $L_{GAN}(G, D_f)$ improves the colorization process by producing plausible valid colorized images.

Full Objective: Our full objective loss can be written as follows:

$$L_{Multi-GAN}(G, D_i, D_f) = L_{M-Dis}(G, D_i, D_f) + \lambda_{l1} L_{l1} + L_{VGG} \tag{9}$$

where λ_{l1} is a parametric factor. Our goal is to solve the minimax game problem with the value function:

$$G, D_i, D_f = arg \min_G \max_{D_i, D_f} L_{Multi-GAN}(G, D_i, D_f) \tag{10}$$

To optimize our networks, we alternate between one gradient descent step on discriminators, then two steps on the generator, for balancing training through different network weights updates.

III. Experimental Analysis

In this section, the performance of the proposed model is discussed. The quantitative evaluation is performed on the places365 dataset [19] which contains more than 1.8 million training images in 365 different high-resolution scene categories.

A. Training details

In this section, we present the training details of our experiments. First, we randomly select 1400 images from places365 dataset. For training our network, Adam is selected as our optimizer for generator and discriminators, with an initial learning rate $\alpha_1 = 0.001$ for the generator and $\alpha_2 = 0.0001$ for discriminators. For the feature extractor, a pre-trained

recognition network VGG-19 [17] was used which is found to be beneficial in representing content and texture information as will be shown in the following results section. Regarding VGG_l in Eqs. (5), (6), (7) we used the last layer (Conv5) of the VGG-19 in our experiments, as we have observed that Conv5 generally produces better results than other layers. As a result of extensive hyper-parameters optimization, we set the weight factors $\lambda_i = 1$ and $\lambda_f = 10^{-3}$ (Eq.3), $\lambda_T = \lambda_C = 0.0001$, $\lambda_{l1} = 100$. The number of training epochs is 90 where the batch size is 4. We have implemented our model in Tensorflow.

B. Result analysis

To study the improvements of the proposed losses function from the GAN framework [12] (Baseline), we have performed an ablation study. We consider different cases by removing content loss only and texture loss only. Some comparison results are presented in Fig.2. We can observe that even with high PSNR and SSIM, using only one loss (Texture or Content) may generate unsatisfactory visual results with some artifacts in the colorized image as shown in black rectangles in Fig.2. So, the combination of these loss functions is essential to capture the low frequencies, ensuring global consistency, as well as the high frequency details of the desired target images, which improve the visual quality of the generated image with a good value of PSNR and SSIM.

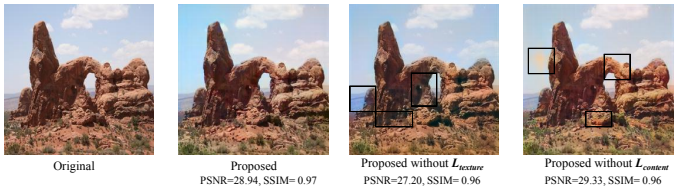

Fig. 2: Figure is better seen zoomed on the digital version of this document. Influence of texture and content losses in the proposed model.

Due to space constraints, we have limited our comparison only with methods that employ a pair of data (Grayscale image, Color Image), such as Nazeri *et al.* [12] and Pix2Pix [11]. The experiment results in Fig.3 illustrate better and more reasonable colors results of the proposed model, which also avoids the color artifacts generated by state-of-the-art methods. For quantitative comparison by the PSNR and SSIM metrics, recent studies [20], [21] have proven that even with pleasing visual results, some models can not show significant improvement in PSNR and SSIM or they produce color images which do not correspond to the real ones, so, the evaluation metrics could not be calculated. As shown in table I, the proposed model produces images with good PSNR and SSIM values and ensure a tradeoff with visual results. This is reflected by the generated images having sharper details and more fine-tuned structures due to matching the target's textural and global content.

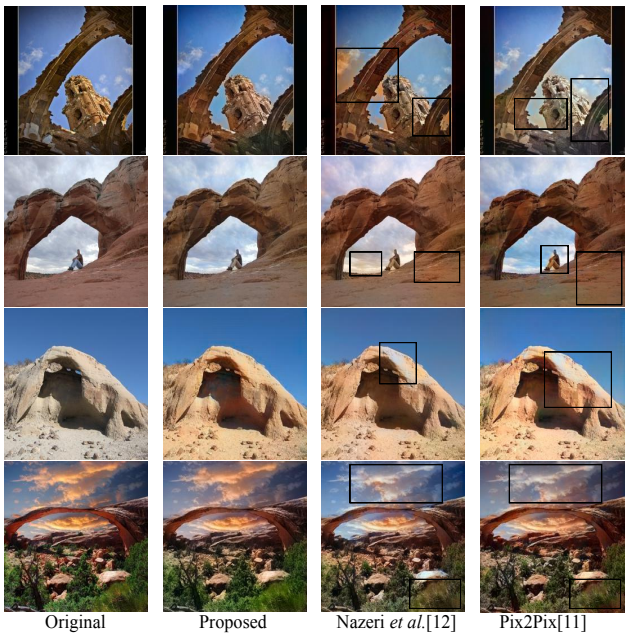

Fig. 3: Figure is better seen zoomed on the digital version of this document. Comparison of the proposed model with Pix2Pix [11] and Nazeri *et al.* [12] on test images of places365.

Method	PSNR	SSIM
Nazeri *et al.* [12]	25.74	0.94
Pix2Pix [11]	18.38	0.60
Proposed Model without $L_{texture}$	21.82	0.89
Proposed Model without $L_{content}$	21.80	0.90
Proposed Model	**23.14**	**0.91**

TABLE I: Qualitative comparison between the proposed model and state-of-the-art methods.

IV. Conclusion

In this paper, we have proposed an effective colorization model to automatically colorize grayscale images. Based on GAN architecture, the proposed model employs two discriminators for features and content image and style transfer techniques which transfer textures and details of a given image. The experimental results show that the proposed model was able to consistently produce pleasing visual colorized images with less artifacts than state-of-the-art methods, moreover, it ensures a tradeoff between qualitative and quantitative results by producing images with good PSNR and SSIM. As a future work, we want to employ the Resnet architecture in our generator, which can retain more of the low-scale features due to the reason that it is shallow, which will improve the colorization results.

References

[1] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," in *ACM Transactions on Graphics (tog)*, vol. 23, no. 3. ACM, 2004, pp. 689–694.

[2] L. Yatziv and G. Sapiro, "Fast image and video colorization using chrominance blending," *IEEE Transactions on Image Processing*, vol. 15, no. 5, pp. 1120–1129, 2006.
[3] Y.-C. Huang, Y.-S. Tung, J.-C. Chen, S.-W. Wang, and J.-L. Wu, "An adaptive edge detection based colorization algorithm and its applications," in *Proceedings of the 13th Annual ACM International Conference on Multimedia*. ACM, 2005, pp. 351–354.
[4] G. Charpiat, M. Hofmann, and B. Schölkopf, "Automatic image colorization via multimodal predictions," in *European Conference on Computer Vision*. Springer, 2008, pp. 126–139.
[5] Z. Cheng, Q. Yang, and B. Sheng, "Deep colorization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 415–423.
[6] M. He, D. Chen, J. Liao, P. V. Sander, and L. Yuan, "Deep exemplar-based colorization," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, p. 47, 2018.
[7] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *European Conference on Computer Vision*. Springer, 2016, pp. 649–666.
[8] G. Larsson, M. Maire, and G. Shakhnarovich, "Learning representations for automatic colorization," in *European Conference on Computer Vision*. Springer, 2016, pp. 577–593.
[9] R. Y. Zhang, J. Y. Zhu, P. Isola, X. Geng, A. S. Lin, T. Yu, and A. A. Efros, "Real-time user-guided image colorization with learned deep priors," *ACM Transactions on Graphics*, vol. 36, no. 4, p. 119, 2017.
[10] 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.
[11] 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.
[12] K. Nazeri, E. Ng, and M. Ebrahimi, "Image colorization using generative adversarial networks," in *International Conference on Articulated Motion and Deformable Objects*. Springer, 2018, pp. 85–94.
[13] S.-J. Park, H. Son, S. Cho, K.-S. Hong, and S. Lee, "Srfeat: Single image super-resolution with feature discrimination," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 439–455.
[14] 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.
[15] 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.
[16] B. J. Balas, "Texture synthesis and perception: Using computational models to study texture representations in the human visual system," *Vision Research*, vol. 46, no. 3, pp. 299–309, 2006.
[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2015.
[18] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE Conference on Computer vision and Pattern Recognition*, 2017, pp. 4681–4690.
[19] B. Zhou, A. Khosla, A. Lapedriza, A. Torralba, and A. Oliva, "Places: An image database for deep scene understanding."
[20] G. Ozbulak, "Image colorization by capsule networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.
[21] M. Sharma, M. Makwana, A. Upadhyay, A. Pratap Singh, A. Badhwar, A. Trivedi, A. Saini, and S. Chaudhury, "Robust image colorization using self attention based progressive generative adversarial network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.