Exploring Double Cross Cyclic Interpolation in Unpaired Image-to-Image Translation

Jorge López
Department of Computer Science
Universidad Católica San Pablo
Arequipa, Perú
Email: jorge.lopez.caceres@ucsp.edu.pe

Antoni Mauricio
Department of Computer Science
Universidad Católica San Pablo
Arequipa, Perú
Email: manasses.mauricio@ucsp.edu.pe

Guillermo Cámara
Department of Computer Science
Federal University of Ouro Preto
Minas Gerais, Brazil
Email: guillermo@ufop.edu.br

***Abstract*—The unpaired image-to-image translation consists of transferring a sample a in the domain A to an analog sample b in the domain B without intensive pixel-to-pixel supervision. The current vision focuses on learning a generative function that maps both domains but ignoring the latent information, although its exploration is not explicit supervision. This paper proposes a cross-domain GAN-based model to achieve a bi-directional translation guided by latent space supervision. The proposed architecture provides a double-loop cyclic reconstruction loss in an exchangeable training adopted to reduce mode collapse and enhance local details. Our proposal has outstanding results in visual quality, stability, and pixel-level segmentation metrics over different public datasets.**

I. Introduction

Image-to-image translation [1] aims to learn a mapping function to convert an image from a source domain to a target domain while preserving its semantic presentations. This problem implies a wide range of computer vision applications beyond style transfer, such as high-quality image generation [2], [3], inpainting [4], [5] and segmentation [6], [7]. Traditionally, those were pixel-to-pixel supervised tasks that required a big amount of paired data which itself is a hard task. To overcome this problem, the majority of authors have adopted unsupervised learning methods, especially explorative GAN-based approaches. Domain exploration is not explicit supervision, then non-explorative GAN models do not unravel critical features.

According to Zhang et al. [8], the main benefit of GANs inside the image-to-image translation is the image-level feedback which better per-pixel information. Nevertheless, GAN models fail when image-level feedback collapse as an overfits consequence in a set of unpaired domains. Zhu et al. [9] introduce a cyclic architecture, called CycleGAN, to perform an exploratory translation without paired data. In every iteration, the latent vectors from both domains are forced to match while exchanging source A and target B domain, iteratively. CycleGAN establishes a bi-directional correspondence reinforced by a loop which ensures that a sample $a \in A$ is the same one after mapping ($A \rightarrow B$) and mapping back ($B \rightarrow A$). This setting lacks a mechanism to enforce the translation regularity resulting in undesirable semantic changes.

This research paper proposes a double-cycle GAN-based architecture considering latent space as a transferable domain to overcome the mode collapse and preserve quality and resolution. This manifold learning approach alongside a self-regularization term encourages the translation regularity. We use Wasserstein distance to blunt known GAN-based model failures, like vanishing gradient and divergence. Furthermore, every control-loop has different loss functions to prevent distortions in domain-specific attributes. We run experiments in various public datasets[1] for style transfer and image segmentation. In both cases, our proposal achieves noticeable results, quantitatively and qualitatively, which implies a large improvement over CycleGAN.

II. Prior Works

Currently, there are two main approaches regarding the image-to-image translation task in the deep learning era. In the first one, the latent space disentangles into independent and specific features to model an explicit mapping function [10]–[12]. While, the second one includes cyclic architectures techniques to achieve a cross-domain translation by the implicit exploration of latent spaces [9], [13]–[16]. This review only covers the second approach, although it includes some essential details from the first one.

A. Non-Cyclic Architectures

Before Isola et al. [1] pioneered deep learning approach for the image-to-image translation, there were prior attempts such as image analogies [17] and exemplar-based procedures [18]. Isola et al. [1] connect various domains by a pixel-level reconstruction considering supervision constraints, although useful, it requires a wide amount of paired data. To overcome pixel-level supervision requirements and dataset constraints, several authors propose techniques for unpaired datasets [9], [13], [15], [19]–[21]. Gatys et al. [19] develop an encoding/decoding procedure to transfer style vectors from the source domain to a white noise vector, iteratively. Next, Gatys et al. [20] expand their previous approach by extracting content and style from two different domains and transfer them to a white noise vector.

Since GAN-based models explore latent spaces to learn high-representative latent vectors and manifold-domain correlations [22], multiples works propose enhancements over

[1] http://people.eecs.berkeley.edu/~taesung_park/CycleGAN/datasets/

vanilla GAN to face the unpaired image-to-image translation problem. Taigman et al. [23] present a domain transfer network (DTN) composed by multiclass GAN loss, a constancy component, and regularization. DTN has an autoencoder as the generator while exchanging domains. Zhang et al. [8] introduce a smoothness term to attain harmonic functions to enforce consistent mappings. Their model, call HarmonicGAN, based on similarity-consistency metrics reduces semantic variations in translation. Zhu et al. [13] perform a generative visual manipulation on several natural domains by latent operations with manifold smoothness expressed in terms of constrained optimization.

B. CycleGAN-based Architectures

Dual learning approach imports machine translation [24] concepts and models, including cycle consistency GANs [9]. The CycleGAN provides a bi-directional prediction by exchanging source and target domain every iteration. Notwithstanding, Zhu et al. [25] argue that mapping functions are ambiguous, then a low-dimensional latent vector may be linked to multiple feasible translations. Thereon, they explicitly boost latent encoding and output to prevent the many-to-one mapping or mode collapse problem. In the Dual-GAN architecture [15], each generator maps a real and generated sample from one domain to the other while producing generated samples from real ones. Hiasa et al. [26] focus on MRI contrast correction for bone structures by translating CT-style by a gradient-consistency loss to enhance the resolution at the boundaries.

Cross-domain architectures lack quality in high-resolution and different domains [14]. Thereon, recent efforts [14], [27], [28] aim to disentangled high-representative latent features to overlook quality and ambiguity problems. Li et al. [14] decompose translations into multi-stage transformations by Stacked Cycle-Consistent Adversarial Networks. Lee et al. [27] take advantage of ambiguity to achieve a diversity-model using a domain-invariant content space and a domain-specific attribute space. Following the cross-domain disentanglement concept, Gonzalez et al. [28] distribute the internal features into exclusive and shared representations via cross-domain autoencoders.

III. Background

Before methods, we explain most in-depth the basic concepts used by the state-of-the-art, including autoencoders, generative adversarial networks, and the CycleGAN.

A. Autoencoders

As Figure 1 illustrates, autoencoders contain two chained networks: An encoder e which reduces the input's complexity to a low-level domain (Z_X) and a decoder d which reconstructs the original data. This arrangement accomplishes a representation that tends to resemble the optimal latent space. Classical autoencoders work well-enough for denoising, visualization, and very-simple images synthesis. Nonetheless, they fail in complex-domain mapping due to the complexity and discontinuity of latent spaces. Thereon, several improvements have been proposed to overcome its limitations and to extend its range of applications. Equation 1 presents the loss function, known as reconstruction loss $\mathcal{L}_{ress}$, and evaluates the distance between the original sample (X) and the reconstructed one (X') using norm-1.

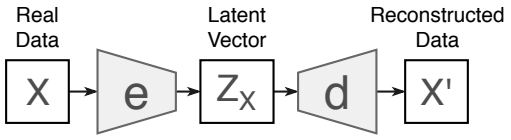

Fig. 1. Autoencoder architecture. e encodes a original sample $x \sim X$ into a latent vector Z_x, then d decodes Z_x to get a reconstructed sample x'.

$$\mathcal{L}_{ress} = \frac{1}{n}\sum_{i=0}^{n} ||x'_i - x_i||_1 \tag{1}$$

B. Generative Adversarial Networks - GANs

Goodfellow et al. [22] proposed a generative model based on adversarial learning. GANs re-sample the domain distribution ρ_r by a competitive game between a generator G and a binary discriminator D. D has to maximize real/fake classification, while G tries to trick D by improving fakes. A random vector ($z \sim \rho_z$) feeds G to generate a fake sample ($x \sim \rho_g$). The adversarial loss $\mathcal{L}_{GAN}$ (Equation 2) represents a min-max adversarial game, where $D(x, \theta_d)$ maximize $log(D(x))$ whereas $G(z, \theta_g)$ minimize $log(1 - D(G(z)))$. $\mathcal{L}_{GAN}$ prioritizes the performance of G, nonetheless, D normally overcome G.

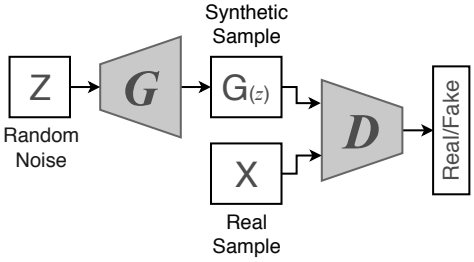

Fig. 2. The generator G maps a noise vector $z \sim Z$ to get a synthetic sample $G(z)$. Then, both the false $G(z)$ and the original x samples feed the discriminator D, which tries to predict whether the samples are real or fake.

$$\begin{aligned}\min_G \max_D \mathcal{L}_{GAN}(D,G) = \mathbb{E}_{x\sim\rho_r(x)}[log(D(x))] \\ +\mathbb{E}_{z\sim\rho_z(z)}[log(1-D(G(z)))]\end{aligned} \tag{2}$$

There are two main problems in GANs training [3], [29]–[31]. First, $\mathcal{L}_{GAN}$ means a two-player non-cooperative game with continuous updating; hence, it is hard to achieve Nash equilibrium. Second, the vanishing gradient problem arises with sigmoid cross-entropy loss [29]. To overcome these problems, we adopt the least-squares loss instead of sigmoid cross-entropy loss (Equations 3 and 4), and a better metric of distribution similarity. Arjovsky et al. [30] introduce the Wasserstein-GAN (WGAN), which uses Wasserstein distance as the loss function (Equation 6). The Wasserstein distance is

the minimum energy cost required to modify the shape of one distribution to another distribution. Given $G(z, \theta_g)$ distribution (ρ_g) and the real distribution (ρ_r), then, Equation 5 shows the Wasserstein ($\mathcal{W}$) distance between ρ_r and ρ_g.

$$\begin{aligned} \min_D \mathcal{L}_{LSGAN}(D) = \frac{1}{2}\mathbb{E}_{x\sim\rho_r(x)}[(D(x)-1)^2] \\ +\frac{1}{2}\mathbb{E}_{z\sim\rho_z(z)}[(D(G(z)))^2] \end{aligned} \tag{3}$$

$$\min_G \mathcal{L}_{LSGAN}(G) = \mathbb{E}_{z\sim\rho_z(z)}[(D(G(z))-1)^2] \tag{4}$$

$$\mathcal{W}(\rho_r, \rho_g) = \inf_{\gamma\sim\prod(\rho_r,\rho_g)} \mathbb{E}_{(x,y)\sim\gamma}[\|x-y\|] \tag{5}$$

$$\begin{aligned} \mathcal{L}_{WGAN}(\rho_r, \rho_g) = \mathcal{W}(\rho_r, \rho_g) = \max_D \mathbb{E}_{x\sim\rho_r}[D(x)] \\ -\mathbb{E}_{z\sim\rho_r(z)}[D(G(z))] \end{aligned} \tag{6}$$

C. Cycle Consistency GANs - CycleGANs

Being A and B two unpaired domains, and G_A and G_B their one-way mapping functions, respectively. Then, G_A transfer the features from A to B, and G_B from B to A. Meanwhile, discriminators (D_A and D_B) tries to classify the real samples X from fakes (X^*) given by the opposed generator. G_A and G_B are cross-domain autoencoders, such that, $G_B(A) = d_B(e_A(A)) = B^*$ and $G_A(B) = d_A(e_B(B)) = A^*$. Figure 3 shows that reconstructions (X^r) are obtained when a generator maps X^*, thereupon, $d_A(e_B(B^*)) = A^r \approx A$ and $d_B(e_A(A^*)) = B^r \approx B$. Finally, the CycleGAN loss function (Equation 8) includes both $\mathcal{L}_{GAN}$ and cycle reconstruction error $\mathcal{L}_{cycle}$ (Equation 7) weighted by a factor λ.

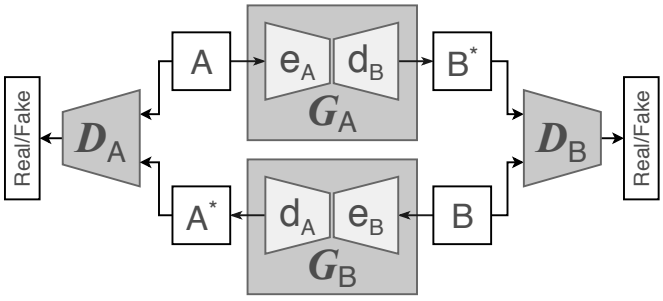

Fig. 3. $A \to B$: G_A transfers $a \in A$ to $b^* \in B^*$. $B \to A$: G_B transfers $b \in B$ to $a^* \in A^*$. Meanwhile, D_A and D_B try to recognize the original samples (x) from fakes (x^*).

$$\begin{aligned} \mathcal{L}_{cycle}(G_A, G_B) = \mathbb{E}_{A\sim\rho_r(A)}[\|G_A(G_B(A)) - A\|_1] \\ +\mathbb{E}_{B\sim\rho_r(B)}[\|G_B(G_A(B)) - B\|_1] \end{aligned} \tag{7}$$

$$\begin{aligned} \mathcal{L}_{CycleGAN} = \mathcal{L}_{GAN}(D_A, G_A) + \mathcal{L}_{GAN}(D_B, G_B) \\ +\lambda\mathcal{L}_{cycle}(G_A, G_B) \end{aligned} \tag{8}$$

IV. Double-cycle GAN architecture

Our proposal is a double-cycle GAN architecture that considers latent spaces an intermediate domain to support translation. We use feedback loops to control the transfer quality considering a consistency cycle loss function per loop. Furthermore, generators are cross-domain autoencoders composed of many ResNet blocks to face the vanishing gradient problem inside the translation stage. Figure 4 shows an extended CycleGAN architecture with the latent-domain Z. Thus, $Z_A = e_A(A) \wedge Z_B = e_B(B)$ are the latent vectors; while, $B^* = d_B(Z_A) \wedge A^* = d_A(Z_B)$ are the transferred style samples from the pixel-level domains.

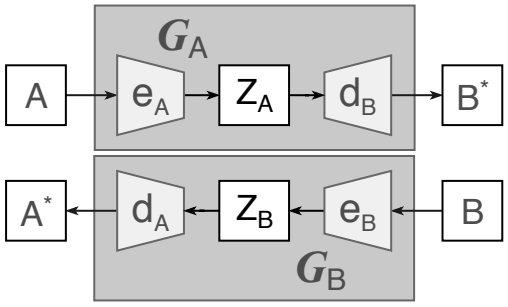

Fig. 4. Latent-space Z as a low-dimensional domain to support the translation.

The vanishing gradient problem affects the transference quality in every loop over time. To overcome its effects, we add reconstruction loops and reconstructed variables (X^r) which come from mapping a transferred variable (X^*). Hence, $X^r \approx X$ including latent vectors Z, which play a control variables role inside every cycle (Equation 9). Figure 5 illustrates the reconstruction cycle pipeline regardless of domain. Finally, Equation 10 is the reconstruction loss at pixel-level given the current setting.

$$\begin{aligned} \mathcal{L}_{cycle_Z}(e_A, e_B, d_A, d_B) = \mathbb{E}_{A\sim p_r(A)}[||z_A^r - z_A||_1] \\ +\mathbb{E}_{B\sim p_r(B)}[||z_B^r - z_B||_1] \end{aligned} \tag{9}$$

$$\begin{aligned} \mathcal{L}_{ress}(e_A, e_B, d_A, d_B) = \mathbb{E}_{B\sim p_r(A)}[||B^i - B||_1] \\ +\mathbb{E}_{A\sim p_r(B)}[||A^i - A||_1] \end{aligned} \tag{10}$$

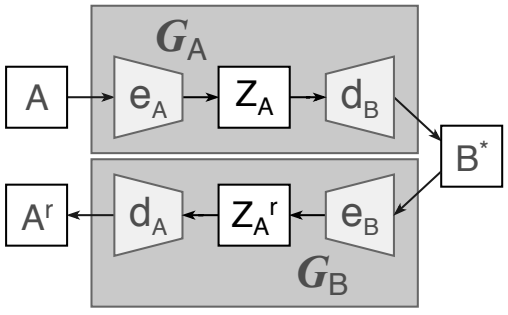

Fig. 5. Reconstruction cycle: $e_B(B^*) = Z_A^r \approx Z_A$ and $d_A(e_B(B^*)) = A^r \approx A$ are reconstructed samples.

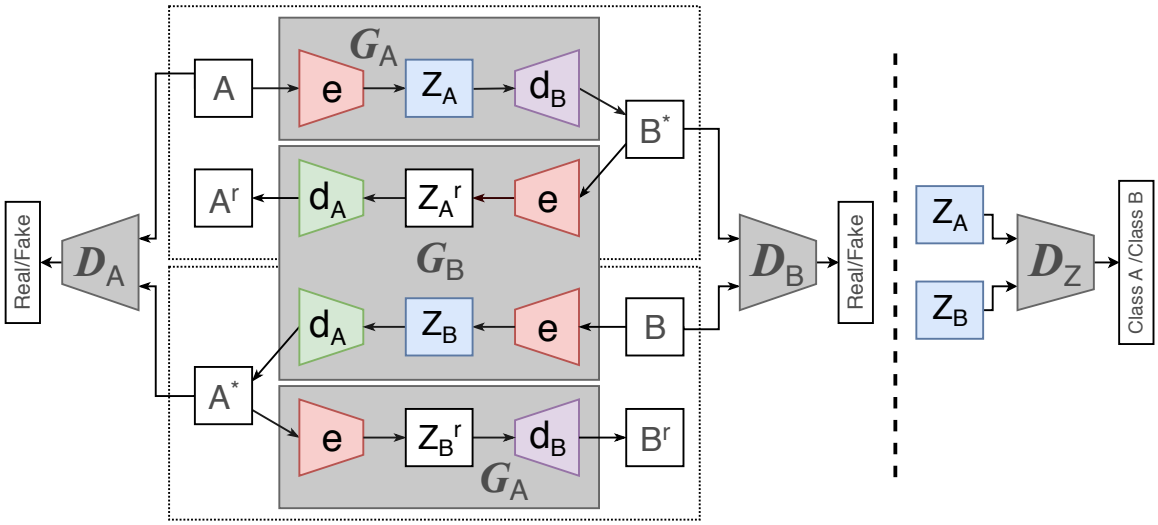

Fig. 6. Double-cycle GAN architecture. $A \rightarrow A^r$: e encodes $a \in A$ into $z_a \in Z_A$, then d_B decodes z_a to get a transferred sample $b^* \in B^*$. The reconstruction cycle closes to get the reconstructed samples at pixel-level ($a^r \in A^r$) and latent-level $z_a^r \in Z_A^*$. $B \rightarrow B^r$: the second cycle follows the same statements but exchanging A and B. Discriminators (D_A and D_B) learn to differentiate between real and transferred samples. Discriminator D_Z helps to avoid overfitting and/or mode collapse inside feature vectors.

We join both reconstruction cycles ($A \rightarrow A^r$ and $B \rightarrow B^r$) using the same encoder to create a continuous latent space for both domains. Therefore, latent vectors Z begin to resemble their reconstructed versions Z^r, and decoders become the primary support of translation. Figure 6 presents a full view of our architecture. As can be seen, $Z_A = e(A)$, $Z_B = e(B)$, $d_B(Z_A) = B^*$ and $d_A(Z_B) = A^*$, being A^* and B^* transferred samples. From both reconstruction cycles, we establish that $Z_A^r = e(B^*)$, $Z_B^r = e(A^*)$, $d_A(Z_A^r) = A^r$ and $d_B(Z_B) = B^r$. Discriminators work like in the CycleGAN, also, we add a latent-variable discriminator D_Z. D_Z aims to distinguish Z_A from Z_B to homogenize their latent vectors and unfold the hidden information.

A. Loss Function

We import the WGAN loss functions to the cyclic loss to face instability and the mode collapse. Due to the high-hierarchy, we split the adversarial loss into the generator loss (Equations 12) and the discriminator loss (Equations 11). Further, a saturation block prevents overflow-values in discriminator training.

$$\begin{aligned}\mathcal{L}_{GAN}(D) = \mathbb{E}_{A\sim\rho_r(A)}[min(0, -1 + D(A))] \\ +\mathbb{E}_{z\sim\rho_z(z)}[min(0, -1 - D(G(z)))]\end{aligned} \tag{11}$$

$$\mathcal{L}_{GAN}(G) = -\mathbb{E}_{z\sim\rho_z(Z)}[D(G(z))] \tag{12}$$

Considering $G_A(x) = e(d_A(x))$ and $G_B(x) = e(d_B(x))$ for Equations 11 and 12. Then, Equations 13 and 14 are loss functions of D_A and D_B, respectively. We merge the loss functions of G_A and G_B into Equation 15, which integrates e, d_A and d_B as variables.

$$\begin{aligned}\mathcal{L}_{GAN}(D_A) =&\mathbb{E}_{A\sim\rho_r(A)}[min(0, -1 + D_A(A))]+ \\ &\mathbb{E}_{B\sim\rho_r(B)}[min(0, -1 - D_A(d_A(e(B))))]\end{aligned} \tag{13}$$

$$\begin{aligned}\mathcal{L}_{GAN}(D_B) =&\mathbb{E}_{B\sim\rho_r(B)}[min(0, -1 + D_B(B))]+ \\ &\mathbb{E}_{A\sim\rho_r(A)}[min(0, -1 - D_B(d_B(e(A))))]\end{aligned} \tag{14}$$

$$\begin{aligned}\mathcal{L}_{GAN}(d_A, d_B, e) = -\mathbb{E}_{A\sim\rho_r(A)}[D_B(d_B(e(A)))] \\ -\mathbb{E}_{B\sim\rho_r(B)}[D_A(d_A(e(B)))]\end{aligned} \tag{15}$$

Latent variables suffer the same pixel-level problems, specially the vanishing gradient problem. Thus, we integrate the LSGAN loss in the inner cycle for the training of D_Z and e (Equations 16 and 17).

$$\begin{aligned}\mathcal{L}_{LSGAN_Z}(D_Z) = \frac{1}{2}\mathbb{E}_{A\sim\rho_r(A)}[(D_Z(e(A)) - 1)^2] \\ +\frac{1}{2}\mathbb{E}_{B\sim\rho_r(B)}[(D_Z(e(B)))^2]\end{aligned} \tag{16}$$

$$\begin{aligned}\mathcal{L}_{LSGAN_Z}(e) = \mathbb{E}_{A\sim\rho_r(A)}[(D_Z(e(A)))^2]+ \\ \frac{1}{2}\mathbb{E}_{B\sim\rho_r(B)}[(D_Z(e(B)) - 1)^2]\end{aligned} \tag{17}$$

Equation 18 presents the reconstruction loss for both reconstruction cycles considering a single encoder e. Further, Equation 19 presents the cyclic loss from Z perspective.

$$\begin{aligned}\mathcal{L}_{ress}(e, d_A, d_B) = \mathbb{E}_{B\sim p_r(A)}[||B^i - B||_1] \\ +\mathbb{E}_{A\sim p_r(B)}[||A^i - A||_1]\end{aligned} \tag{18}$$

$$\mathcal{L}_{cycle_Z}(e, d_A, d_B) = \mathbb{E}_{A\sim p_r(A)}[||z_A^r - z_A||_1] + \mathbb{E}_{B\sim p_r(B)}[||z_B^r - z_B||_1] \quad (19)$$

Since $e = e_A = e_B$, Equations 9 and 10 stay with three input variables each instead of four but preserving their syntaxis. Then, $\mathcal{L}_{cycle_Z}(e, d_A, d_B) = \mathcal{L}_{cycle_Z}(e_A, e_B, d_A, d_B)$ and $\mathcal{L}_{auto}(e, d_A, d_B) = \mathcal{L}_{auto}(e_A, e_B, d_A, d_B)$. Equations 13, 14 and 16 updates D_A, D_B and D_Z, respectively. Lastly, Equation 20 integrates the reconstruction loss at pixel-level (Equation 18), the cyclic loss at latent-level (Equation 19), the adversarial loss of the encoder e (Equation 17) and the decoders (Equation 15). Parameter λ controls the reconstruction and cyclic loss functions to overflow. Meanwhile, β switches the training process among the encoder and decoders.

$$\mathcal{L}_{Total}(D) = \lambda\mathcal{L}_{ress}(e, d_A, d_B) + \lambda\mathcal{L}_{cycle_Z}(e, d_A, d_B) + \beta\mathcal{L}_{LSGAN_Z}(e) + (1-\beta)\mathcal{L}_{GAN}(d_A, d_B, e) \quad (20)$$

B. Training Details

We use 256 256 pixels datasets, including Cityscapes, Horse2zebra, Monet2photo, and Photo2VanGogh. Dataset split into 80% to train and 20% to validate. Adam optimizer is used to train e, d_A, d_B, D_A, and D_B with $\alpha = 0.00002$; and $\alpha = 0.0002$ for D_Z. To force the convergence, we set 200 epochs and a decay factor of 20% every 100 epochs for α. $\lambda = 10$ to minimize $\mathcal{L}_{ress}$ and $\mathcal{L}_{cycle_Z}$. The model process 1 sample per batch due computational limitations. Finally, we define a switch factor $n = 5$ to control β value. Algorithm 1 is the cross-domain training scheme, which prioritizes latent space reconstruction before image generation.

Algorithm 1: Cross-domain training algorithm.

Input: A and B: Input domains. K: Epochs.
λ: Saturation factor. α: Learning rate.
m: Batch size. n: Switch factor.
Output: D_A, D_B, D_Z, e, d_A and d_B trained.

```
for c = 1, 2, ..., K do
    j ← 0
    for i = 1, 2, ..., |A| do
        Sample m tuples < a, b >; a ∈ A ∧ b ∈ B
        if j < n then
            Eq. 13, 14 and 16: Update D_A, D_B and D_Z
            Eq. 20 with β = 1: Update e, d_A and d_B
            j ← j + 1
        else
            Eq. 20 with β = 0: Update e, d_A and d_B
            j ← 0
```

V. Experiments and results

In this section, we show the experimental tests to contrast our results against CycleGAN results for We contrast our experimental results against CycleGAN for different datasets and hyperparameters. Performed tasks include style transfer (Figures 8 and 9) and image segmentation (Figure 7). The Cityscapes dataset consists of paired tuples <real image, segmented image>. Monet2photo, Horse2zebra, and Photo2VanGogh are two-domain unpaired datasets.

A. Metrics

We use three metrics from FCN semantic segmentation and scene parsing evaluations [7]. Given n_{ij} pixels of class i labeled as class j from k different classes, and let $t_i = \sum_j n_{ij}$ be the total number of pixels from class i. Then, we compute:

1) Pixel accuracy: $\sum_i n_{ii} / \sum_i t_i$.
2) Mean accuracy: $\frac{1}{k}\sum_i n_{ii}/t_i$.
3) Mean of region intersection over unions (UI): $\frac{1}{k}\sum_i n_{ii}/(t_i + \sum_j n_{ji} - n_{ii})$.

We employ these metrics to evaluate the segmentation task using the Cityscapes dataset. Table I shows a comparison between our results against CycleGAN.

TABLE I
METRIC COMPARISON.

	Pixel accuracy	Mean accuracy	Mean UI
CycleGAN	0.52	0.17	0.11
Ours	0.65	0.22	0.16

VI. Conclusions and future works

Latent spaces as a control-domain replace quite right the end-to-end approach. Then, adjustments, like unifying the encoders, improve the transfer quality by forcing both latent-domains to get closer. Hyperparameters must be tuned according to the task and domains to avoid overfitting and underfitting problems. The learning stage works in two blocks. The first one specializes in feature maps generation, while the second one focuses on more realistic details. We outperform the CycleGAN considering FCN metrics and visual conditions.

In brief, we take advantage of cyclic reconstructions to enhance quality while subduing the mode collapse problem. Our proposal shows improvements over the CycleGAN in terms of quality and boundary resolution. However, our method is not interpretable nor easy to implement. Hence, better metrics are required to evaluate generative models in different tasks. Also, it requires a bigger computational capacity and time to converge.

Upcoming works will examine an explicit disentanglement of latent vectors to surpass the domain conditions, like size and resolution. Furthermore, we will explore new evaluation metrics to achieve an adequate semantic evaluation.

Acknowledgment

The present work was supported by grant 234-2015-FONDECYT (Master Program) from Cienciactiva of the National Council for Science, Technology and Technological Innovation (CONCYTEC-PERU).

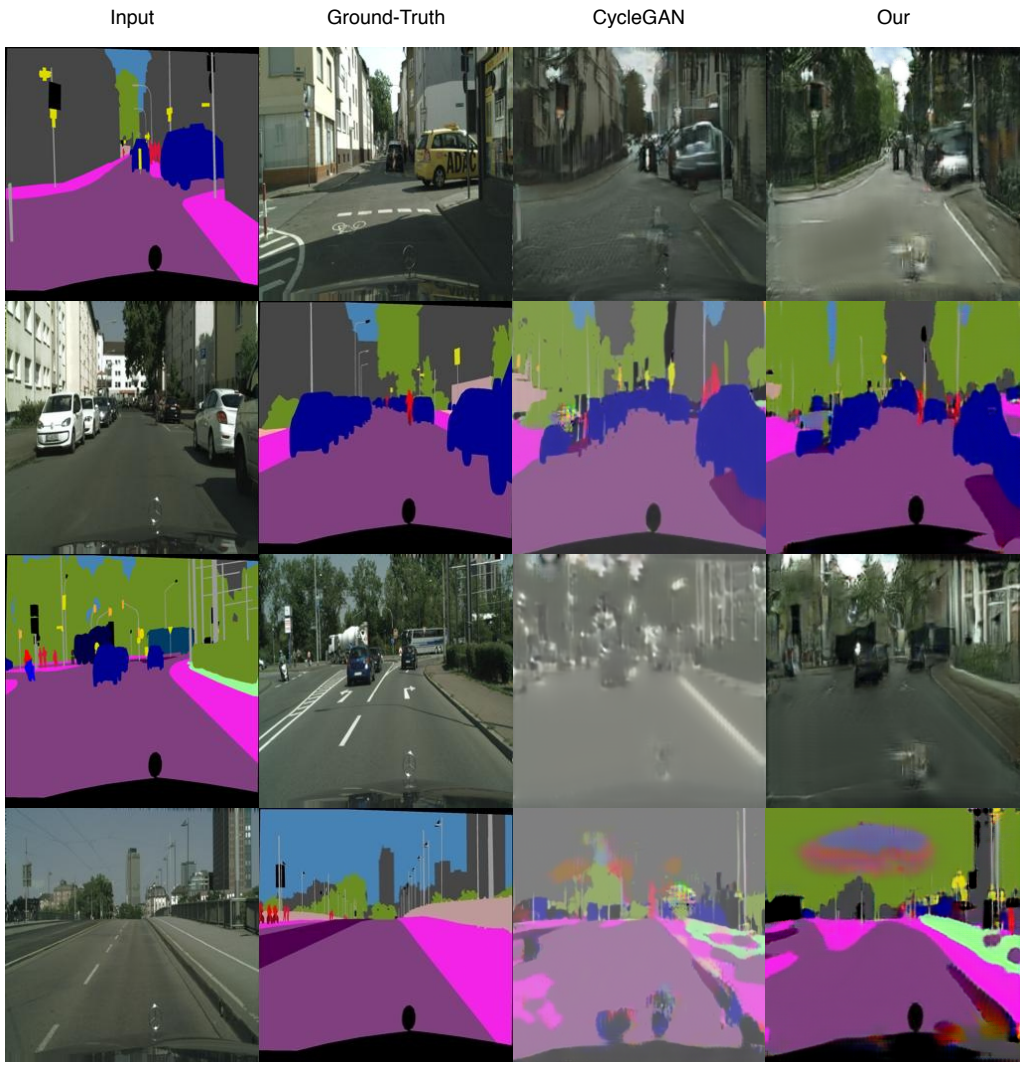

Fig. 7. Comparison of results for the semantic segmentation task using the Cityscapes dataset.

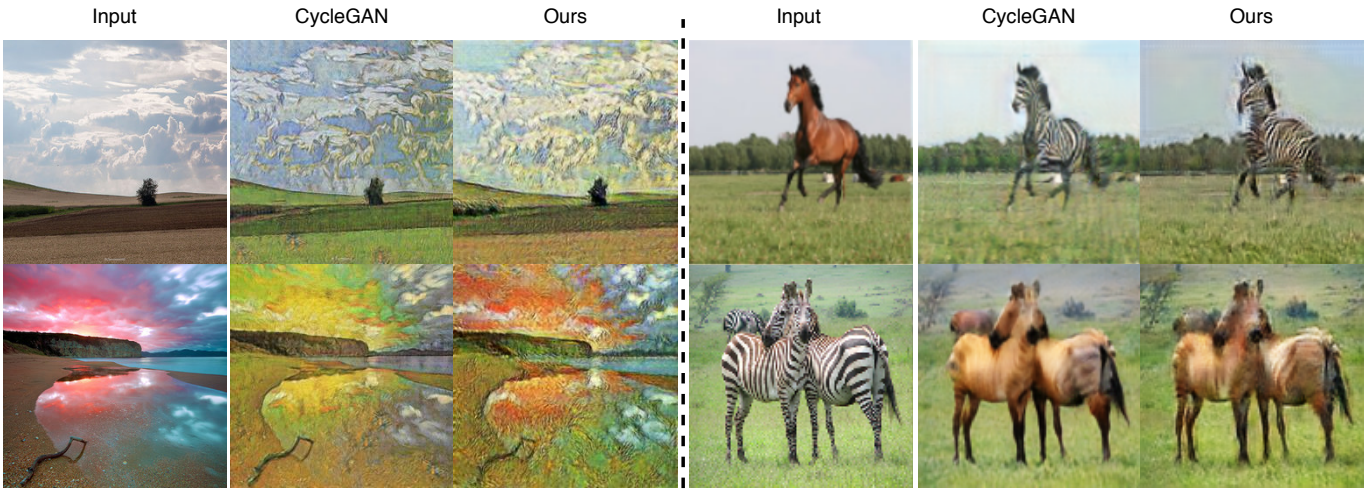

Fig. 8. Comparison of results for style transfer task using Monet2photo and Horse2zebra datasets.

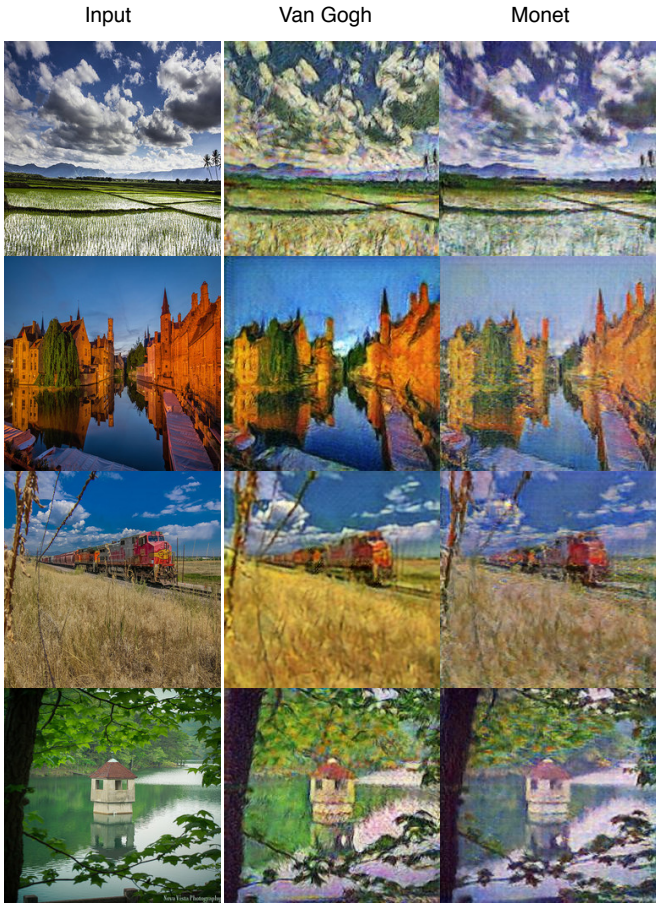

Fig. 9. Results of transfer style using the Monet2photo and Photo2VanGogh datasets.

REFERENCES

[1] 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.

[2] S. U. Dar, M. Yurt, L. Karacan, A. Erdem, E. Erdem, and T. Çukur, "Image synthesis in multi-contrast mri with conditional generative adversarial networks," *IEEE transactions on medical imaging*, 2019.

[3] A. Mauricio, J. López, R. Huauya, and J. Diaz, "High-resolution generative adversarial neural networks applied to histological images generation," in *International Conference on Artificial Neural Networks*. Springer, 2018, pp. 195–202.

[4] Y. Wang, X. Tao, X. Qi, X. Shen, and J. Jia, "Image inpainting via generative multi-column convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2018, pp. 331–340.

[5] W. Wang, Q. Huang, S. You, C. Yang, and U. Neumann, "Shape inpainting using 3d generative adversarial network and recurrent convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2298–2306.

[6] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, "Semantic segmentation using adversarial networks," *arXiv preprint arXiv:1611.08408*, 2016.

[7] 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*, 2015, pp. 3431–3440.

[8] R. Zhang, T. Pfister, and J. Li, "Harmonic unpaired image-to-image translation," *arXiv preprint arXiv:1902.09727*, 2019.

[9] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[10] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 172–189.

[11] S. Benaim and L. Wolf, "One-shot unsupervised cross domain translation," in *Advances in Neural Information Processing Systems*, 2018, pp. 2104–2114.

[12] L. Chen, H. Zhang, J. Xiao, W. Liu, and S.-F. Chang, "Zero-shot visual recognition using semantics-preserving adversarial embedding networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1043–1052.

[13] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros, "Generative visual manipulation on the natural image manifold," in *European Conference on Computer Vision*. Springer, 2016, pp. 597–613.

[14] M. Li, H. Huang, L. Ma, W. Liu, T. Zhang, and Y. Jiang, "Unsupervised image-to-image translation with stacked cycle-consistent adversarial networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 184–199.

[15] Z. Yi, H. Zhang, P. Tan, and M. Gong, "Dualgan: Unsupervised dual learning for image-to-image translation," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2849–2857.

[16] J. López, A. Mauricio, J. Diaz, and C. Guillermo, "Cross-domain interpolation for unpaired image-to-image translation," in *International Conference on Computer Vision Systems*. Springer, 2019, pp. 120–129.

[17] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, "Image analogies," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001, pp. 327–340.

[18] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001, pp. 341–346.

[19] L. Gatys, A. S. Ecker, and M. Bethge, "Texture synthesis using convolutional neural networks," in *Advances in neural information processing systems*, 2015, pp. 262–270.

[20] 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.

[21] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2107–2116.

[22] 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.

[23] Y. Taigman, A. Polyak, and L. Wolf, "Unsupervised cross-domain image generation," *arXiv preprint arXiv:1611.02200*, 2016.

[24] D. He, Y. Xia, T. Qin, L. Wang, N. Yu, T.-Y. Liu, and W.-Y. Ma, "Dual learning for machine translation," in *Advances in Neural Information Processing Systems*, 2016, pp. 820–828.

[25] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, "Toward multimodal image-to-image translation," in *Advances in Neural Information Processing Systems*, 2017, pp. 465–476.

[26] Y. Hiasa, Y. Otake, M. Takao, T. Matsuoka, K. Takashima, A. Carass, J. L. Prince, N. Sugano, and Y. Sato, "Cross-modality image synthesis from unpaired data using cyclegan," in *International Workshop on Simulation and Synthesis in Medical Imaging*. Springer, 2018, pp. 31–41.

[27] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, and M.-H. Yang, "Diverse image-to-image translation via disentangled representations," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 35–51.

[28] A. Gonzalez-Garcia, J. van de Weijer, and Y. Bengio, "Image-to-image translation for cross-domain disentanglement," in *Advances in Neural Information Processing Systems*, 2018, pp. 1287–1298.

[29] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2794–2802.

[30] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.

[31] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Advances in Neural Information Processing Systems*, 2017, pp. 5767–5777.