Unpaired Faces to Cartoons: Improving XGAN

Stev H. Ramos Joel Cabrera Daniel Ibáñez Alejandro B. Jiménez-Panta
César Beltrán-Castañón Edwin Villanueva
Pontifical Catholic University of Peru
Lima, Peru
{mhuamanr, jjcabrera, daniel.ibanez, alejandrob.jimenezp, cbeltran, ervillanueva}@pucp.edu.pe

Abstract

Domain Adaptation is a task that aims to translate an image from a source domain to a desired target domain. Current methods in domain adaptation use adversarial training based on Generative Adversarial Networks (GAN). In the present work, we focus on the task of domain adaptation from real faces to cartoon face images. We start from a baseline architecture called XGAN and introduce some improvements to it. Our proposed model is called W-XDGAN, which uses a form of GAN called Wasserstein-GAN, learns to approximate the Wasserstein Distance, and adds a denoiser to smooth the output cartoons. Whereas the original XGAN paper only presented a qualitative analysis, the advantages of this solution are demonstrated both quantitatively and qualitatively by comparing the results with models such as UNIT and original XGAN. Our code and models are publicly available at https://github.com/IAmigos/avatar-image-generator.

1. Introduction

One of the main challenges of machine learning is to obtain significant results with unsupervised approaches. This type of task eliminates the limitation of supervised models in which the training set and the test set must come from the same distribution, and feature-label pairs are required for training [22].

Domain adaptation is defined as the ability to transfer the knowledge to perform a specific task from one domain to another in a way that both are related [10, 20]. A recently used approach is to have a feature representation that is not domain dependent [29]. This is why some methods aim to learn the underlying attributes of both domains using different training objectives such as reconstruction [4] and adversarial [1, 9]. This type of task has a particular application in images, where an attempt is made to transfer the content and shape of an image from one domain to another while preserving the colors and semantic content, better known as image-to-image translation [23].

Several papers on image-to-image translation have been reported among which GAN-based models have shown good results due to adversarial training in an unsupervised manner. This way of training represents an alternative to obtain non-domain-dependent feature representations. Specifically, conditional GANs in which the input is not Gaussian noise, but an image that is translated to another domain [6, 15, 23]. Furthermore, CycleGAN addresses this task with the use of a pixel-level consistency loss [31]. XGAN (Cross-GAN), on the other hand, introduces a consistency at the level of the latent space shared by both domains (semantic representation) [23]. However, the latter lacks stable training in the feature space shared by both domains, leading to mode collapse [3].

In this paper, improvements in XGAN are proposed by replacing the Binary Cross Entropy (BCE), which classifies both domains, with Wasserstein Loss, which approximates the Wasserstein Distance [2, 25], in order to solve the domain adversarial task in the shared latent space. In addition, we propose to add a denoiser following the output of the cartoon generator. This denoiser eliminates speckle noise and incorrect coloring of the generated images [24]. Furthermore, we changed the training procedure of the GAN used to improve sample quality in XGAN [23] to WGAN-GP [2, 13]. The model we propose is called W-XDGAN (Wasserstein-Cross-D-GAN). These improvements were measured with the Fréchet Inception Distance (FID) [14], which is used to compare the distribution of the generated cartoons with the original ones. The VGG-Face [21] and the CartoonSet datasets, proposed by [23], were used for experimentation in the Section 4.

2. Related Work

Some deep learning models have recently been proposed to tackle the problem of unsupervised image-to-image translation. In [23], the authors generalized the problem of style transfer to a higher level of semantics by learning to couple two domains with shared semantic content but

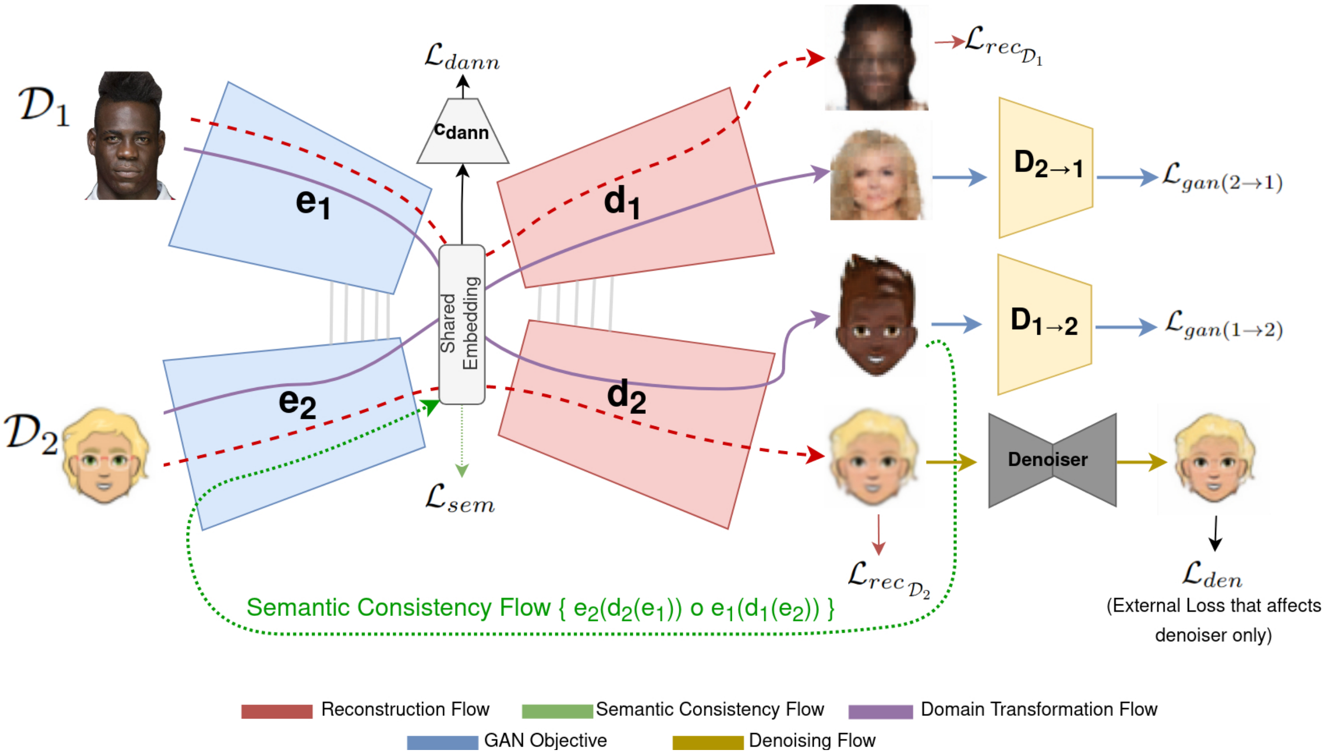

$\mathcal{D}_1$
$\mathcal{D}_2$
$\mathcal{L}_{dann}$
c_{dann}
e1
e2
Shared Embedding
d1
d2
$\mathcal{L}_{sem}$
$\mathcal{L}_{rec_{\mathcal{D}_1}}$
D2→1
$\mathcal{L}_{gan(2\to1)}$
D1→2
$\mathcal{L}_{gan(1\to2)}$
Denoiser
$\mathcal{L}_{rec_{\mathcal{D}_2}}$
$\mathcal{L}_{den}$
(External Loss that affects denoiser only)
Semantic Consistency Flow { e2(d2(e1)) o e1(d1(e2)) }
Reconstruction Flow
Semantic Consistency Flow
Domain Transformation Flow
GAN Objective
Denoising Flow

Figure 1. **Cohesive View of the proposed W-XDGAN based on XGAN**. The components e_1, e_2, d_1, d_2 represent the autoencoder that acts as a generator. First, an image from face domain (D_1) or cartoon domain (D_2) fed the encoder to generate a feature representation. Then, the decoder receives the latent space and learns to generate an image from the other domain. Finally, if the image is a face, which means that it belongs to D_1, the generated cartoon is sent to the denoiser in order to enhance the image quality.

different representation. They experimented with the subdomains of cartoon face images and real face images. The model they introduced is XGAN, which is based on two encoder-decoder networks with a shared latent space that generates a shared representation of the common domain semantic content. Their main contribution is the semantic consistency loss which encourages the model to preserve semantics in the learned embedding space. Similar to this work, UNIT [18] implemented a coupled VAEGAN architecture with a shared latent space trained with pixel level cycle-consistency. Their model learns a joint feature-level representation but they do not use a semantic consistency component. The works of CycleGAN [31], DualGAN [27], and DiscoGAN [16] which are based on GANs report good results for image-to-image translation when both domains are close to each other, but fail for more significant shifts. These models do not share the latent space and only consider pixel to pixel consistency.

3. Proposed Method

In this section, we detail the architecture we propose based on XGAN.

3.1. XGAN Model

We rely on XGAN as a baseline model. The goal of this model is to learn the domain transfer from two unpaired domains [23] in an unsupervised way using two encoder-decoder networks, one for each domain, in such a way that both share the last layers of the encoders and the first layers of the decoders to push the representations to the same latent space. Both the encoder and the decoder use convolutional and deconvolutional layers, respectively. Furthermore, XGAN uses the objective of a GAN in which the discriminator tries to make the generated examples as realistic as possible.

The training objective consists of five weighted loss functions that control the contribution of each of them and the architecture is shown in Figure 1. Given domains $\mathcal{D}_1$ and $\mathcal{D}_2$, L_{rec} reconstructs the image of both domains by comparing it to the original one using L_{2norm}. L_{dann} acts as a small GAN in the middle of the latent space using a classifier as a discriminator. This method was introduced by [10], whose goal is to keep the latent space of both domains indistinguishable. In this case, instead of using typical GAN training, the gradient encoder layer is connected to

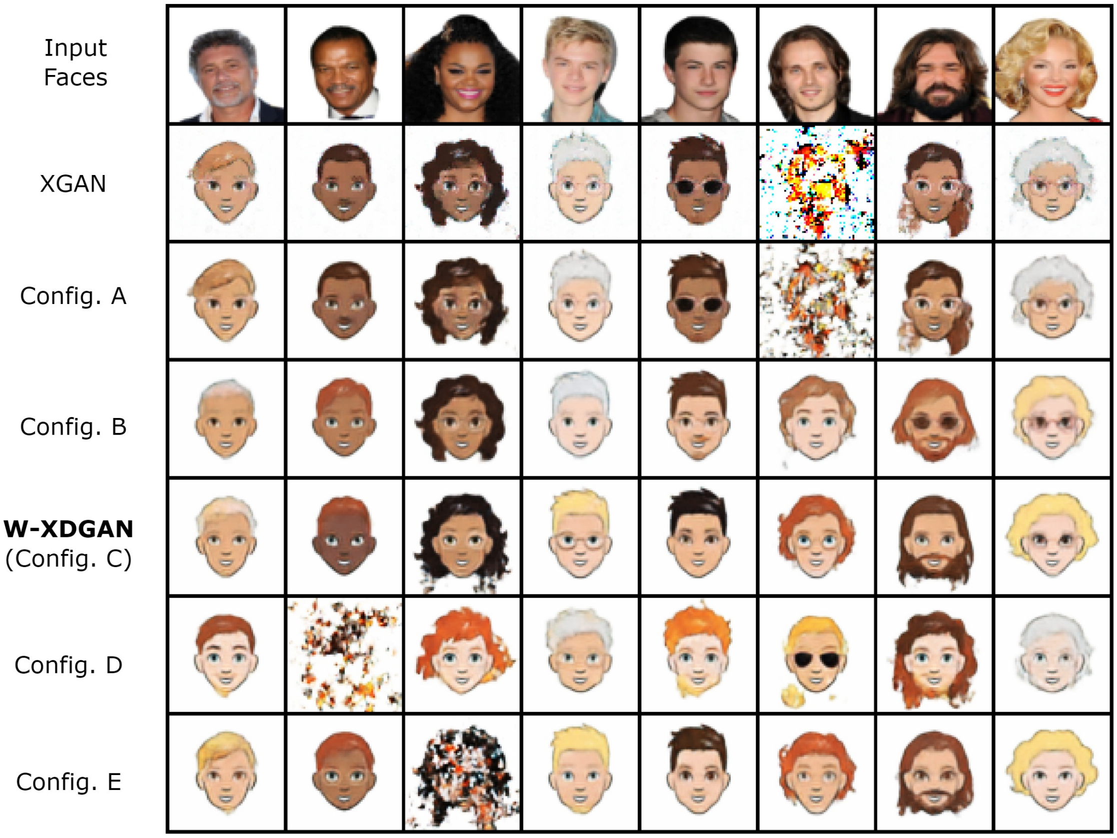


Figure 2. **Random Generated Cartoon Samples by the configurations described in Table 1. Configuration A:** XGAN model + Denoiser. **Configuration B:** XGAN model + Denoiser + Wasserstein Loss for Domain Adversarial Task. **Configuration C (W-XDGAN):** XGAN model + Denoiser + Wasserstein Loss for Domain Adversarial Task + WGAN-GP instead of GAN. **Configuration D:** W-XDGAN without spectral normalization. **Configuration E:** W-XDGAN trained with Cartoon to Face Discriminator. Visual artifacts were observed across most configurations such as background noise, asymmetry in the eyes, uneven coloring and incorrect color matching in hair and skin, accidental glasses, and messy blurred hair. The most stable cartoons were obtained by the W-XDGAN model.

the reverse layer which transmits the gradient with the opposite sign [10]. L_{sem} is the main contribution of XGAN, as it acts as semantic consistency, i.e., instead of comparing the output images at the pixel level, it compares them in a high-level representation in the semantic latent space. This is calculated from the L_{1norm} distance between two vectors obtained by domain translation of the input image in the latent space at the end of the encoders [23], as shown in Figure 1. L_{gan}, on the other hand, is the typical loss function of adversarial generative models [12]. Finally, we decided not to use the optional loss function L_{teach}, which acts as a kind of regularization to guide the latent space of faces to a better representation, but only acts in one domain, which is why it is asymmetric. Consequently, there is a risk of generating the opposite effect of L_{dann} [23]. All these components are summarized in the Equation 1.

$$\mathcal{L}_{xgan} = \omega_r \mathcal{L}_{rec} + \omega_d \mathcal{L}_{dann} + \omega_s \mathcal{L}_{sem} + \omega_g \mathcal{L}_{gan} \quad (1)$$

3.2. Improving Quality and Stability

In this section we will present the main contributions of our proposed model.

3.2.1 WGAN-GP instead of GAN

The type of Generative Adversarial Networks introduced by [25] have shown good results in generating images from two models that compete with each other while each one improves at the same time. The original method uses a classifier to determine if the sample is real or fake using binary cross entropy. However, the main problem with this approach is that the discriminator usually outperforms the generator by far, since the task of the former is to determine a value between 0 and 1, while the generator tries to assemble an image in three channels [26]. For this reason, it is difficult to train a GAN.

Wasserstein GAN (WGAN) was introduced by [2] as an alternative to cope with the aforementioned problems. This algorithm approximates the Earth Mover's Distance, which, can be understood as the minimum amount of land that must be passed from one distribution to another so they are equal. Its main benefit is that the loss function no longer has the flat regions that caused the vanishing gradient problem. Consequently, the training is more stable. In addition, [13] introduces a regularizer component at the gradient level that forces it to have a maximum of 1, which is called Gradient Penalty (GP). The formulation of this WGAN with Gradient Penalty (WGAN-GP) loss function described before can be seen in Equation 2.

$$\begin{aligned}\mathcal{L}_{gan} = \min_G \max_D & \overbrace{\mathbb{E}_{\boldsymbol{x}\sim\mathbb{P}_r}[D(\boldsymbol{x}|\boldsymbol{y})] - \mathbb{E}_{\tilde{\boldsymbol{x}}\sim\mathbb{P}_g}[D(\tilde{\boldsymbol{x}}|\boldsymbol{y})]}^{\text{Discriminator Loss}} \\ & + \lambda \overbrace{\mathbb{E}_{\hat{\boldsymbol{x}}\sim\mathbb{P}_{\hat{\boldsymbol{x}}}}[(||\nabla_{\hat{\boldsymbol{x}}} D(\hat{\boldsymbol{x}}|\boldsymbol{y})||_2 - 1)^2]}^{\text{Gradient Penalty}},\end{aligned} \tag{2}$$

Additionally, [19] proposes spectral normalization as an additional way to stabilize the training of GANs. This is applied after the convolutional layers in the discriminator. We demonstrate in the experiment Section 4.3 that the cartoons generated with this approach are more diverse and more realistic according to the FID metric.

3.2.2 Replacing the Domain Classifier

As stated above, the current XGAN model uses a domain classifier at the end of the shared encoder layers. This is to enhance cross-domain transformations [23]. In other words, it tries to get both learned embeddings to be in the same latent space. We observed with the use of t-SNE that the vectors fall into different spaces of the domain as it is trained. That is, the classifier destabilizes over the epochs.

In that sense, we chose to replace the domain classifier with a network that learns to approximate the Wasserstein Distance [25]. The reason is to provide more stable training by avoiding to fall into those flat regions where the gradient is zero, and thus verifying that the learned embeddings fall into the same latent space even over long training periods. This phenomenon is verified in the Section 4.

This approximation of the Wasserstein Distance [2] can be achieved from the L_{dann} shown in Equation 3, keeping the use of the gradient reversal layer to invert the order of the transmitted gradients [10]:

$$\begin{aligned}\min_{\theta_{e_1},\theta_{e_2}} \max_{\theta_{cdann}} & \overbrace{\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_1}[c_{dann}(e_1(\boldsymbol{x}))] + \mathbb{E}_{\tilde{\boldsymbol{x}}\sim\mathcal{D}_2}[c_{dann}(e_2(\tilde{\boldsymbol{x}}))]}^{\text{Domain Adversarial Loss}} \\ & + \lambda \overbrace{\mathbb{E}_{\hat{\boldsymbol{x}}\sim\mathcal{D}_{\hat{\boldsymbol{x}}}}[(||\nabla_{\hat{\boldsymbol{x}}} c_{dann}(\hat{e}(\hat{\boldsymbol{x}}))||_2 - 1)^2]}^{\text{Gradient Penalty}},\end{aligned} \tag{3}$$

3.2.3 Denoising and Color Correction of the Generated Cartoons

The results of the XGAN model usually show incomplete reconstructions and poor colorization. To deal with this problem, we experimented with the use of a denoising autoencoder as mentioned in [24]. This denoiser improves the results at the FID level (see experiment Section 4.3) since it removes noise from poorly generated images. This is achieved from a reconstruction of the original image with the generated image, just like L_{rec}, but the difference is that this is a separate model, so the gradients do not propagate to the generator (see Equation 4).

$$\mathcal{L}_{den} = \mathbb{E}_{\boldsymbol{x}\sim p\mathcal{D}_2}(||x - den(\hat{x})||_2), \hat{x} = d_2(e_2(x)) \tag{4}$$

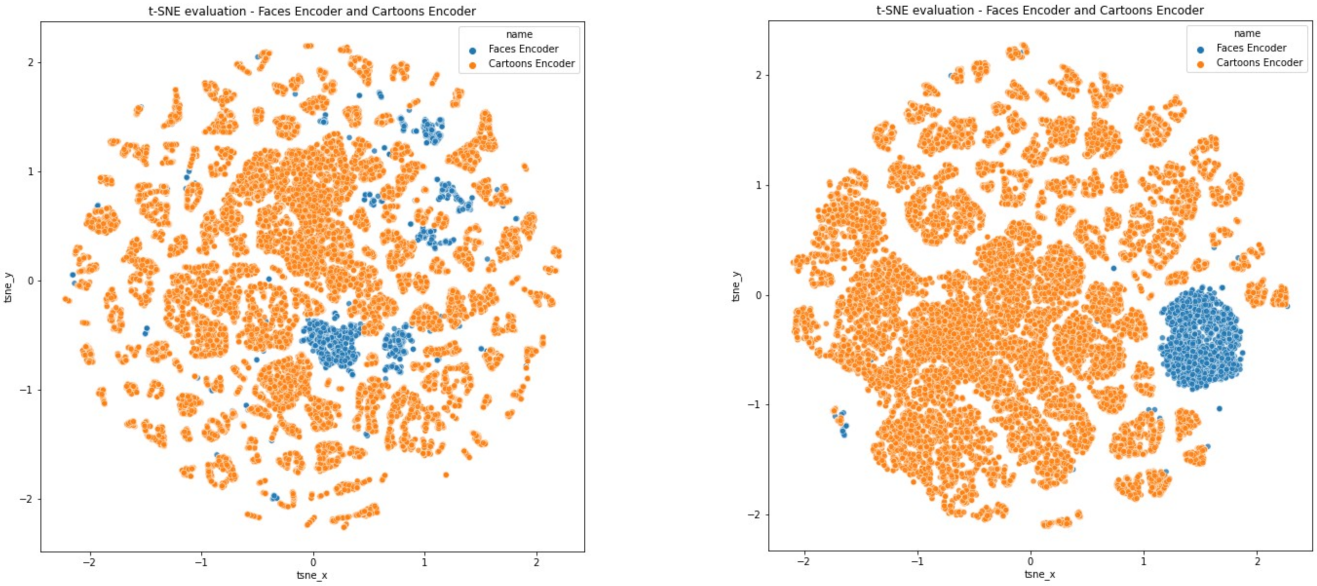
4. Experiments and Discussion

In this section, we will show the experiments followed quantitatively and qualitatively to demonstrate the effectiveness of our improvement proposal. First, we compare the improvements with respect to the original XGAN model, without the introduction of the teacher loss and with a single discriminator which is the domain of cartoons. Then, we show from grid search that some contributions of the loss functions mentioned in Section 3 have a more relevant weight in the task of generating images. Finally, we compare our best model, given in the previous step, with the UNIT [18] model, which has shown promising results in the domain adaptation task. For this reason, we evaluate the task in generating images from the domain of faces to cartoons. In addition, we use more face images to demonstrate that more diverse images are necessary for more favorable results.

We implemented the baseline model XGAN from scratch because no code was available in the original work [23] and used the original implementation [1] in the case of UNIT [18].

The model was trained with 64x64 images and 4 ADAM optimizers [17] were used in total for the generator, discriminator, denoiser and domain adversarial task. All the models

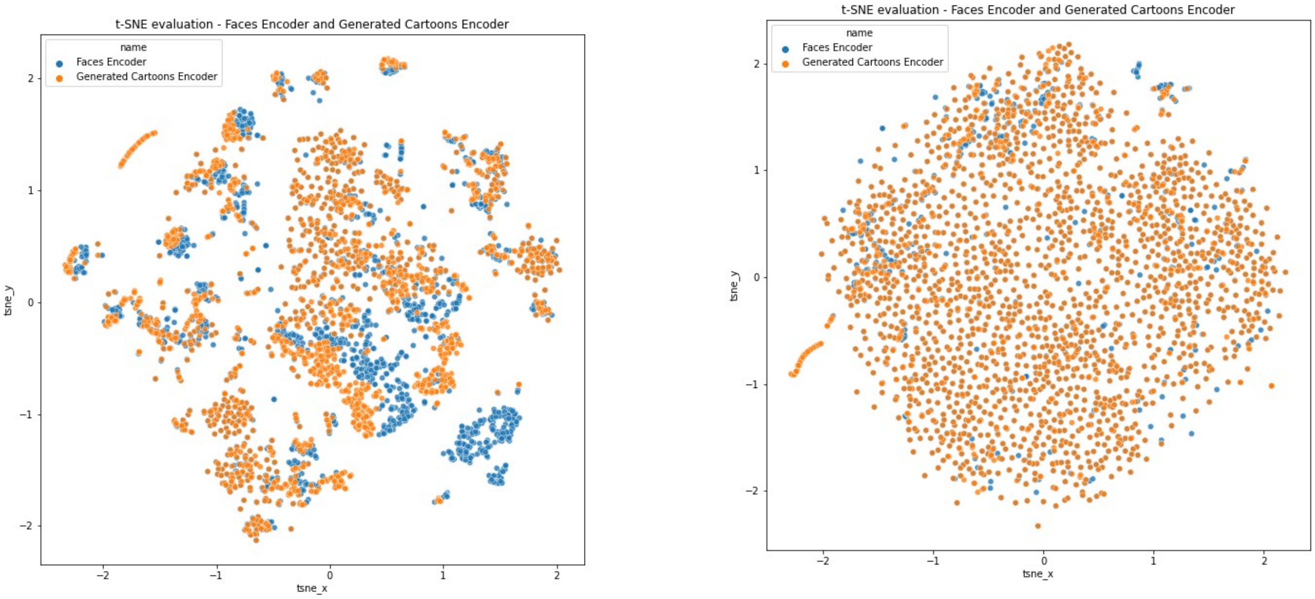
[1] https://github.com/mingyuliutw/UNIT



(a) XGAN: Faces Encoder and Cartoons Encoder. SD=-0.06

(b) W-XDGAN: Faces Encoder and Cartoons Encoder. SD=0.03

Figure 3. t-SNE Embedding - Visualization of the latent spaces for Faces Encoder and Cartoons Encoder



(a) XGAN: Faces Encoder and Generated Cartoons Encoder. SD=0.10

(b) W-XDGAN: Faces Encoder and Generated Cartoons Encoder. SD=0.01

Figure 4. t-SNE Embedding - Visualization of the latent spaces for Faces Encoder and Generated Cartoons Encoder

were run at 200 epochs on a Lambda Deep Learning Workstation (lambdalabs.com) with four GeForce RTX 2080 Ti GPU cards installed on a system running the Ubuntu 18.04.5 LTS operating system.

4.1. Datasets

We have used the same datasets used by [23] for this cross-domain translation task. The dataset used as the target domain is called CartoonSet [23]. This dataset is a collection of random-generated 2D cartoon face images, provided in 10k and 100k sets. The second one, used as the source

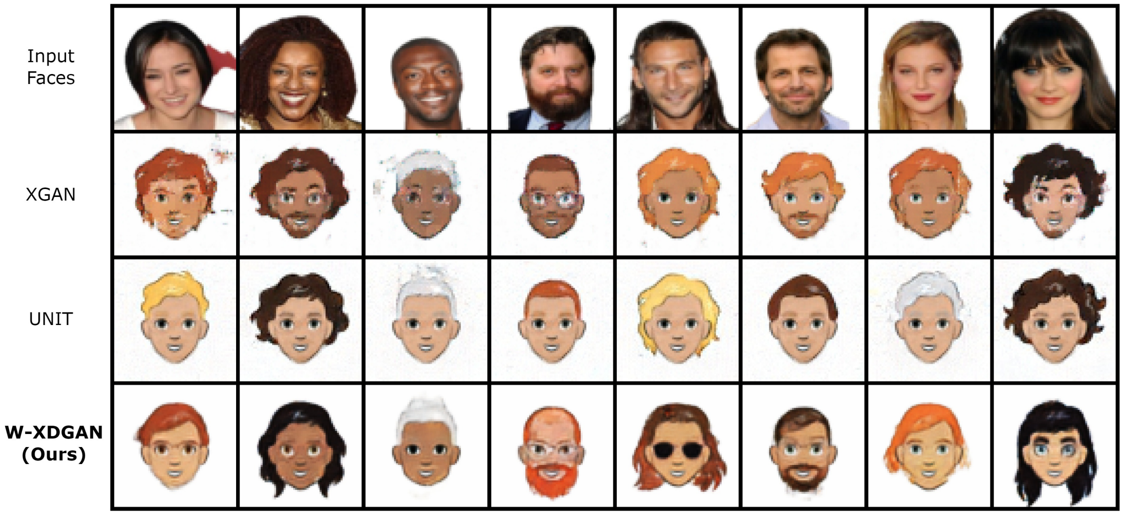


Figure 5. **Random Generated Cartoon Samples by different models: XGAN, UNIT and Our Proposal**. Some important visual artifacts appear with less frequency in the generated images by the W-XDGAN model such as background noise, blurred and messy hair, and incorrect color matching in hair and skin.

domain, is called VGG-Face dataset [21]. This is composed of real people's frontal-face images.

As the CartoonSet contains labeled attributes, an exploratory analysis was possible to do. On the contrary, the VGG-Face dataset does not have additional information about the images. This is why an analysis of the distribution of the latter dataset was not achieved and the possibly unbalanced data could cause a bias in the trained model.

Once the exploratory analysis was performed, the balance of the attributes was verified. In addition, a filtering of the images was carried out to reduce the number of atypical attribute combinations. The total amount of images used from the CartoonSet was 35,134, using a 90/10 training/test split.

Moreover, we preprocessed the VGG-Face dataset by cropping the face bounding box and removing the background of the images using the Keras' PSPNet [30] pretrained segmentation model trained on the Pascal VOC 2012 dataset [8]. As there were some images in which the preprocessing did not have good results, these images were removed. The total amount of images used from the VGG-Face dataset was 2,361, using a 90/10 training/test split.

4.2. Evaluation Metrics

The task of unpaired domain translation is known to be better evaluated through qualitative analysis. However, we propose image comparison through Fréchet Inception Distance (FID) as proposed by [14]. We used FID in order to better capture statistics between synthetic and real images, as explained by [14]. Statistics from distributions of both synthetic and real images were obtained in the test dataset explained previously with the Inception V3 model using the final average pooling layer. FID has been used in several other works to evaluate the quality of synthetic images generated by GANs such as in [28], [5], [11].

Additionally, experiments on the learned shared latent space were conducted. We applied t-SNE on the encoded latent space of the test images previously referred, and obtained a Silhouette Distance (SD) to evaluate the clustering capacity and coherence. Finally, we verified these results with qualitative analysis on the t-SNE in a similar way as in [7], and on the synthetic and real images for each cluster obtained.

4.3. Domain Translation

Before diving into the comparison with UNIT, we first demonstrate experimentally that the properties we added improve sample quality.

In Table 1, we compare the FID for various architectures of the proposed model starting from the baseline XGAN model. It is observed that all improvements outperform XGAN. It is important to mention that the added configurations have a high computational cost.

We start with configuration "A" which contains the denoiser. With this, we confirm the importance of adding this at the end of the generator since the FID is 106.85 which is lower compared to 159.36 of XGAN. Then, we replace the domain classifier loss by Wasserstein Loss as mentioned in Section 3 and use the spectral norm. Compared to configuration "A", this configuration ("B") clearly improves the

Configuration	Denoiser	Domain Adv. Loss	GAN Objective	Spectral Norm.	Discriminators	FID
XGAN	-	BCE	GAN	-	$D_{1\to 2}$	159.36
A	✓	BCE	GAN	-	$D_{1\to 2}$	106.85
B	✓	W. Loss	GAN	✓	$D_{1\to 2}$	86.48
C (W-XDGAN)	✓	**W. Loss**	**WGAN-GP**	✓	$D_{1\to 2}$	**70.96**
D	✓	W. Loss	WGAN-GP	-	$D_{1\to 2}$	123.32
E	✓	W. Loss	WGAN-GP	✓	$D_{1\to 2}$ and $D_{2\to 1}$	81.74

Table 1. **Evaluating different configurations.** The FID score between real and generated cartoons. The second and fifth columns describe whether the Denoiser and the Spectral Normalization are used, respectively. The third column describes whether Binary Cross Entropy or Wasserstein Loss was used as the Domain Adversarial Loss. The fourth column describes the Generative Adversarial Network objective: base GAN or Wasserstein GAN with Gradient Penalty. The sixth column describes the Discriminators used: Face to Cartoon translation ($D_{1\to 2}$) and/or Cartoon to Face translation ($D_{2\to 1}$). Random samples generated by these configurations are shown in Figure 2. The Weight hyperparemeter values of all these configurations are $\omega_d = 0.93, \omega_g = 0.98, \omega_r = 1, \omega_s = 0.45$.

FID score of the model, which decreases from 106.85 to 86.48 (19% less). Also, as demonstrated in [2], the use of the Wasserstein Loss avoids falling into mode collapse; that is, the generated cartoons are more diverse. The last configuration ("C") contains the improvements mentioned above along with the change from GAN (discriminator) to WGAN-GP (critic). Although the difference is not so great with respect to configuration "B", it is observed that there is a total difference between the baseline model XGAN with configuration "C". Mainly, this is due to the fact that, as mentioned in [23], the combination of L_{gan} and L_{sem} can be understood as two subtasks of L_{dann}. Because the way of training has not changed, we say that this behavior does not depend on the type of loss function (Binary Cross Entropy or Wasserstein Loss), but on the weights that each of them have in the final function.

Additionally, in order to demonstrate the use of spectral norm, we train the model with configuration "C" without the use of the spectral norm ("D"). In Table 1, a value higher that the one obtained with its use ("C") is observed. This suggests that the stability of the model is affected by the spectral norm. On the other hand, as suggested in [23], it is not necessary to train two discriminators. With one of them, which should be the domain for cartoons, it is enough. To verify this, configuration "C" was trained but with the use of the cartoon-to-face domain discriminator ("E") and the results show that there is not much difference, but more time is needed since more parameters are used. The generated samples by all these configurations are shown in Figure 2.

4.4. Evaluation in the Latent Space

We investigate the structure of the latent space of the embeddings learned by the encoders of each domain. For this, we take the model of configuration "C" (see Table 1) and compare the latent space with the XGAN model using t-SNE. In this case there are two types of embeddings. The first one, face encoder and cartoon encoder, is obtained from the independent coding of images from both domains until reaching the end of the shared layers of the encoders at the bottleneck of the model. The second one, face encoder and generated cartoon encoder, is obtained from the encoding of an original face image and its corresponding generated cartoon encoded again in the latent space. This is used to determine if the semantic consistency, proposed by [23], preserves its characteristics when a face image is transformed into a cartoon.

4.4.1 Face Encoder vs Cartoon Encoder

In this analysis, the L_{dann} loss function is taken into account, which task is to make the domains fall into the same latent space, but at the same time distinguish between them [23]. This means that they should not be far from each other and also that they should be grouped together. In Figures 3 and 4, it is observed that in both cases the latent space of the faces encoder falls in the same space as the cartoons encoder, but in the original XGAN model these are indistinguishable. On the other hand, with the "C" configuration it is evident that a group of only faces encoders was formed. This is also supported by the Silhouette Distance which is 0.03 compared with 0.06 of the XGAN model. This arrangement of the latent space is a better point of equilibrium to share the semantic information between the domains, as seen in the results evaluated with FID in Table 1 compared to the original XGAN. In fact, it is shown that the use of Wasserstein Loss as a replacement for the Domain Classifier Loss in the latent space performs this task in a better way.

4.4.2 Face Encoder vs Generated Cartoon Encoder

In this case, the L_{sem} task, that tries to push the semantic characteristics of a face image and its generated cartoon in the same space, is taken into account. This means that even though the face has been transformed, its high-level

ω_d	ω_g	ω_r	ω_s	FID
100	0.1	100	100	139.21
0.1	0.1	100	0.1	127.93
0.1	10	100	10	110.39
100	10	100	100	104.38
0.1	10	100	100	99.73
0.1	0.1	0.1	0.1	94.90
0.1	10	0.1	10	93.24
0.1	100	100	100	90.68
100	10	100	10	89.20
0.1	10	100	0.1	85.61
0.1	100	100	0.1	84.79
0.1	100	100	10	82.04
100	0.1	100	10	79.54
100	0.1	100	0.1	74.80
0.93	0.98	1	0.45	* 70.96
100	**10**	**100**	**0.1**	**60.43**

Table 2. **Grid Search in W hyperparameters.** The FID score for the configuration C with different values of the Weight contribution. FID of the configuration C with fixed W hyperparemeters (*)

semantic features are comparable. In both, the "C" configuration and the XGAN model, this behavior is visible. Also, the Silhouette Distance has a value very close to 0. This means that the group vectors (original faces and generated cartoons), which we know to be related, fall at the same points (See Figures 3 and 4).

4.5. Sensitivity to Hyperparameters

In the experiments shown above, fixed hyperparameters were used to control the contribution of each loss function. However, we believe that the training procedure is sensitive to these values. Therefore, we train various models from a grid search with values of 0.1, 10, 100. Table 2 shows the FID values according to this experimentation. It is observed that the most determining value is the weight of the L_{dann}. This is understandable since it is the part that determines whether the embeddings of both domains fall into the same latent space as shown in Section 4.4. The better the embeddings are represented in the latent space, the better the adaptation of the domain of faces to the domain of cartoons will be. Another detail to analyze is that the weight given to the L_{sem} loss function should not be so high, since it was observed in Section 4.4 that this can be easily achieved. A high value in the contribution of this value causes the even representation in the latent space to be lost, as demonstrated by observing the value of the FID in Table 2.

4.6. Comparison with UNIT

Moreover, we compare our best model, obtained by optimization with grid search, with one of the models that has good results in the domain adaptation task, which is UNIT [18]. This was trained with the same settings as our proposed model. Table 3 shows the result under two different amounts of images of the domain of faces. We increase more face images, because there is a clear imbalance between the amounts of images of both domains. In both cases, the FID value suggests that our model outperforms UNIT. The generated samples are shown in Figure 5.

Method	FID (original data)	FID (aug. data)
XGAN	159.36	-
UNIT	114.25	225.81
W-XDGAN	**60.43**	**55.30**

Table 3. **Comparison between XGAN, Proposed W-XDGAN and UNIT.** The FID score between real and generated cartoons under original data and augmented data

5. Conclusions and Further Work

This paper introduced several improvements to the XGAN model for the domain matching task. By stabilizing the training, it was possible to generate better cartoons compared to the XGAN and UNIT ones. This was achieved thanks to the replacement of the domain classifier loss with Wasserstein Loss, which helped regularize the latent space. Besides, the denoiser plays an important role in order to remove possible bad cartoon generation, as it improves the quality of the cartoons making them more realistic. Finally, although its function is not so relevant at the beginning, replacing the GAN discriminator with the WGAN-GP critic, that tries to approximate the Wasserstein Distance, stabilized the training. We demonstrated all of this quantitatively by showing the value of the FID in the experiments.

As future work, we believe that it is important to balance the number of images of the domains in question. This would add diversity, as some of the generated cartoons tend to show red colors and other biased features partly due to the data imbalance between domains. In addition, new architectures such as diffusion models have shown promising results in image generation in recent years. It would be interesting to experiment with these in the domain translation task.

Acknowledgement

This work was supported by the Artificial Intelligence Group IA-PUCP which provided the computational infrastructure for the experimental part of this research. Special thanks to Piero Molino for his mentoring at the beginning of this project; and to Gissella Bejarano and Pablo Fonseca for reviewing this research.

References

[1] Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, and Mario Marchand. Domain-adversarial neural networks, 2015. 1

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan, 2017. 1, 4, 7

[3] Sanjeev Arora, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. Generalization and equilibrium in generative adversarial nets (GANs). In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 224–232. PMLR, 06–11 Aug 2017. 1

[4] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 343–351, Red Hook, NY, USA, 2016. Curran Associates Inc. 1

[5] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. *CoRR*, abs/1809.11096, 2018. 6

[6] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8789–8797, 2018. 1

[7] Victor Costa, Nuno Lourenço, João Correia, and Penousal Machado. Demonstrating the evolution of gans through t-sne. In Pedro A. Castillo and Juan Luis Jiménez Laredo, editors, *Applications of Evolutionary Computation*, pages 618–633, Cham, 2021. Springer International Publishing. 6

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 6

[9] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1180–1189, Lille, France, 07–09 Jul 2015. PMLR. 1

[10] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016. 1, 2, 3, 4

[11] Fei Gao, Shengjie Shi, Jun Yu, and Qingming Huang. Composition-aided sketch-realistic portrait generation. *CoRR*, abs/1712.00899, 2017. 6

[12] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. 3

[13] Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein gans. *CoRR*, abs/1704.00028, 2017. 1, 4

[14] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 1, 6

[15] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. CyCADA: Cycle-consistent adversarial domain adaptation. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1989–1998. PMLR, 10–15 Jul 2018. 1

[16] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. *CoRR*, abs/1703.05192, 2017. 2

[17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 4

[18] Ming-Yu Liu, Thomas M. Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. *CoRR*, abs/1703.00848, 2017. 2, 4, 8

[19] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *CoRR*, abs/1802.05957, 2018. 4

[20] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010. 1

[21] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In Mark W. Jones Xianghua Xie and Gary K. L. Tam, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 41.1–41.12. BMVA Press, September 2015. 1, 6

[22] Vishal M Patel, Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Visual domain adaptation: A survey of recent advances. *IEEE Signal Processing Magazine*, 32(3):53–69, 2015. 1

[23] Amélie Royer, Konstantinos Bousmalis, Stephan Gouws, Fred Bertsch, Inbar Mosseri, Forrester Cole, and Kevin Murphy. *XGAN: Unsupervised Image-to-Image Translation for Many-to-Many Mappings*, pages 33–49. Springer International Publishing, Cham, 2020. 1, 2, 3, 4, 5, 7

[24] Chao Shang, Aaron Palmer, Jiangwen Sun, Ko-Shin Chen, Jin Lu, and Jinbo Bi. Vigan: Missing view imputation with generative adversarial networks. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 766–775, 2017. 1, 4

[25] Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018. 1, 3, 4

[26] Lilian Weng. From GAN to WGAN. *CoRR*, abs/1904.08994, 2019. 3

[27] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation. *CoRR*, abs/1704.02510, 2017. 2

[28] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International conference on machine learning*, pages 7354–7363. PMLR, 2019. 6

[29] Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. On learning invariant representations for domain adaptation. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 7523–7532. PMLR, 09–15 Jun 2019. 1

[30] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6230–6239, 2017. 6

[31] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017. 1, 2