IGAN: Inferent and Generative Adversarial Networks

Luc VIGNAUD

ONERA, the French aerospace lab, dept. of radar and electromagnetism,
University Paris-Saclay, FRANCE

Luc.vignaud@onera.fr

ABSTRACT

This article presents IGAN (Inferent Generative Adversarial Networks), a neural architecture that learns both a generative and an inference model on a complex high dimensional data distribution, i.e. a bidirectional mapping between data samples and a simpler low-dimensional latent space. It extends the traditional GAN (Generative Adversarial Networks) framework with inference by rewriting the adversarial strategy in both the image and the latent space with an entangled game between data-latent encoded posteriors and priors. It brings a measurable stability and convergence to the classical GAN scheme, while keeping its generative quality and remaining simple and frugal in order to run on a lab PC. IGAN fosters the encoded latents to span the full prior space: this enables the exploitation of an enlarged and self-organised latent space in an unsupervised manner. An analysis of previously published articles sets the theoretical ground for the proposed algorithm. A qualitative demonstration of potential applications like self-supervision or multi-modal data translation is given on common image datasets including SAR (Synthetic Aperture Radar) and optical imagery. The code is available for download in Matlab file-exchange [1].

1.0 INTRODUCTION

1.1 Background on generative adversarial networks and variational auto-encoders

Generative Adversarial Networks (GANs) [2] and Variational Auto Encoders (VAEs) [3] are standard methods for learning generative models of complex data distributions. In their basic form, GANs tend to generate better quality data samples than VAEs, but suffer from stability and mode collapse problems [4]. GANs were not originally built to produce inference unlike VAEs. Inference is however crucial for numerous tasks that are much easier when processed in the latent space rather than in the data space: e.g. classification, clustering, disentanglement, interpolation or extrapolation, compression, domain and style transfer, to name a few [5] [6][7][8].

Let X_r be a real data distribution and Z_p a known prior model, e.g. a vector of Gaussians. Let $\tilde{X}_p$ be a data distribution generated from Z_p, and $\tilde{Z}_r$ a latent distribution estimated from X_r. E is defined as an Encoder that encodes data to latents (e.g. $\tilde{Z}_r = E(X_r)$), G a Generator that generates data from latents (e.g. $\tilde{X}_p = G(Z_p)$), and D a Discriminator with a single decision output.

A Vanilla GAN is composed by a Generator and a data Discriminator. It builds an adversarial game between real and generated (*fake*) data, using $D(X_r)$ and $D(\tilde{X}_p) = D\left(G(Z_p)\right)$ to minimize their Jensen Shannon Divergence [2]. Such architecture aligns the generated data distribution with the real one, but ignores latents inference.

A Vanilla VAE uses an Encoder and a Generator to minimize the two terms of the ELBOEvidence Lower Bound Optimization (ELBO) [3]. The first term minimizes the Kullback Liebler Divergence between the latents encoded from real data and the prior. The second one is a regularization term that minimizes the reconstruction error in the real data space $\varepsilon(X_r, G(E(X_r)))$. It aligns the posterior distribution of latents

encoded from real data with prior, and constraints the reconstruction data error to follow a given distribution (e.g. Gaussian if $\varepsilon = L_2$). However, the Euclidean distance yields blurry data reconstructions, partly because its data similarity judgment is not robust to minor data transformations such as translations.

1.2 Bridging the gap between GAN and VAE

Many solutions have been proposed to combine the best of the two worlds: VAEGAN [8], BIGAN [9], ALI
Many solutions have been proposed to combine the best of the two worlds: VAEGAN [9], BIGAN [10], ALI [11], ALICE [12], AGE [13], IntroVAE [14], BigBIGAN [15], AEGAN [16], DALI [17], LIA/GAN [18], ALAE [19]… Basically, all these methods bring a latent inference to the classical GAN, and compete to better align distributions in both latent and data spaces, hopefully for both posteriors and priors. Their relative performance is often measured by the ability of the algorithm to reconstruct the input data X_r from the auto-encoded regeneration of its inferred latent: $X_r \sim G(E(X_r))$. It is yet impossible to expect a perfect reconstruction of real data since the dimension space reduction from data to latents implies that some information has to be lost: details or outliers that are not statistically relevant to the training dataset need be filtered out of the generation process, and consequently out of the reconstruction. Other reconstruction choices have to be made depending on the input data types and the final application of the algorithm. Most methods, if not all, are focused on photographic optical images applications, and thus reconstruct images via a Euclidian distance alternative, for instance with a perceptual loss built on features extracted from the latest layers of an external pretrained classifier, as inspired by StyleGAN [20].

Only a few methods consider the latent space on a par level with the data space, while it is *de facto* where most of the post-training data processing will be applied and most useful applications drawn. Yet, if we follow Rabelais's famous advice to "*break the bone and suck out the substantific marrow*" [*Gargantua*], the reconstruction of prior latents should even prevail on real data reconstruction to assure a bijective inversion between the Generator and the Encoder on the prior space, i.e. to form an optimal <u>latent</u> auto-encoder. In ALI [11], authors already suggested that an Encoder could use such a reconstruction while playing the conventional GAN scheme. Nevertheless, ALI, its twin BIGAN [10] (Bidirectional Generative Adversarial Networks), or its later extension BigBIGAN [15] used a large Discriminator embedding both images and latents whose aim is to equalize the joint probabilities $p(X_r, E(X_r)) = p(G(Z_p), Z_p)$, but with no explicit reconstruction term. Unfortunately, this lack of deterministic point-wise matching between X_r and $G(E(X_r))$ was shown to produce poor data reconstructions. Recently, DALI [17] proposed to guide the classical GAN adversarial game with a reconstruction term exclusively computed in the latent space to benefit from an easier reconstruction error model than in the data space. Then, two similar techniques (LIA/GAN [18] and ALAE [19]) introduced an intermediate latent space reconstruction, with a complementary latent auto-encoder linked to the prior latent space. AEGAN [16] presents a clever revisit of the famous unpaired image-to-image translator CycleGAN [21], where the latent space takes the place of one of the target image spaces. The problem is then seen as a double GAN in both image and latent spaces (but not embedded). A cycle consistency loss is built on the reconstruction of auto-encoded latents and images to assure the data-to-latent pairing that needs to be learned.

This analysis suggests that:

- A latent GAN counterpart is mandatory to allow the encoded latents distribution to span the full extension of prior space and explore all realistic possibilities for generating data.

- The latent and data GANs should be embedded within a single Discriminator to emphasize the joint probabilities equalization. This embedding needs to be done on an intermediate space, as already required by their dimension difference.

- A cycle consistency is needed for data-to-latent pairing and requires at least the auto-encoded latents reconstruction.

With IGAN (Inferent Generative Adversarial Networks), I propose to aggregate this framework with an original entangled and embedded adversarial game in both latent and image space. The architecture is kept fairly light with a low computation and memory load in order to run on a lab PC with a single GPU.

2.0 IGAN: INFERENT GENERATIVE ADVERSARIAL NETWORKS

2.1 IGAN architecture

IGAN has been built with the intention to give an equal importance to both data and latent spaces. Since the adversarial game between fake $\hat{X}_p$ and real data X_r is responsible for the high quality of data generation, the latent generation/inference should also benefit from its own adversarial game between *fake* encoded latents of real data $\tilde{Z}_r$ and *real* prior model Z_p. Previous works like DALI [17] and BigBIGAN [15] pointed on the importance to equalize the joint probabilities $p(X_r, E(X_r)) = p(G(Z_p), Z_p)$. We claim that it is even better to equalize the following joint probabilities: $p(X_r, Z_p) = p(G(Z_p), E(X_r))$. This modified condition enables us to rewrite the coupled adversarial game in an embedded space where whatever (X_r, Z_p) couple represents a "True data-latent couple", and $(G(Z_p), E(X_r))$ stands for the "False/generated data-latent couple" to play the adversarial game with. The overall IGAN architecture is presented in the following Figure 1.
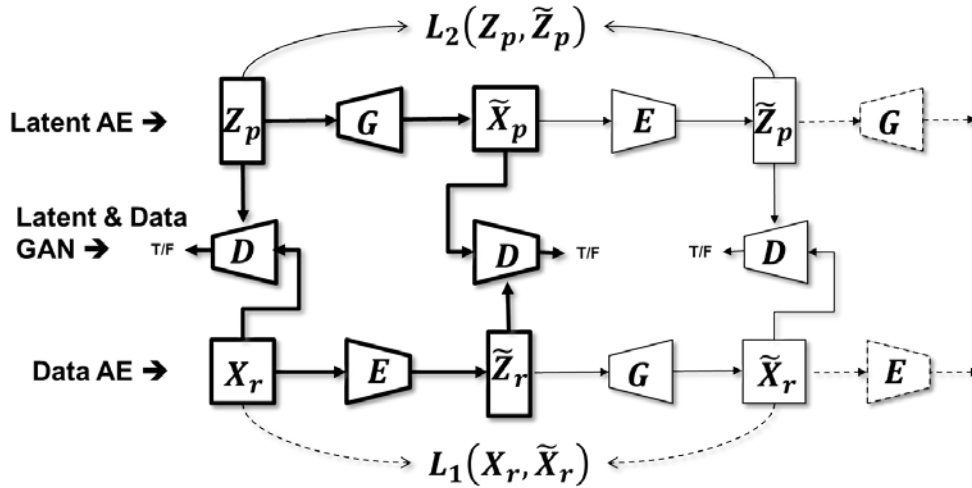

Figure 1: IGAN: Inferent and Generative Adversarial Networks architecture

Prior latent Z_p and real data X_r are both fed to a unique Discriminator D, concatenation being performed at a secondary latent stage in the latest layers of the Discriminator. This sets the reference for what will be used as the "True data-latent couple" discrimination. Prior latents are also turned into generated data $\tilde{X}_p$ through Generator G, while the real data are encoded into generated latents $\tilde{Z}_r$ by Encoder E. These "False/generated data-latent couples" will be driven through the same Discriminator as the "True data-latent couples" to feed the adversarial game. This seeds the following adversarial minimax optimization goal for the networks:

$$\min_{G,E} \max_{D} \left\{ \mathbb{E}_{X_r, Z_p} \left[D(X_r, Z_p) \right] - \mathbb{E}_{X_r, Z_p} \left[D\left(G(Z_p), E(X_r) \right) \right] \right\} \tag{1}$$

In Figure 1, one can easily spot the $(E \circ G)$ data auto-encoder and $(G \circ E)$ latent auto-encoder, respectively at the bottom and top lines. It is important to see that we will also be able to use the reconstruction of real data and prior latents to follow $p(G(E(X_r)), E(G(Z_p))) = p(X_r, Z_p)$ and, furthermore, the reconstruction

of generated data and latents encoded from real data with $p\big(G(E(G(Z_p)), E(G(E(X_r)))\big) = p(X_r, Z_p)$. Incorporating the corresponding additional adversarial games in Equation (2) will ensure that not only the reconstructed real and fake data, but also the reconstructed real and fake latents will follow the real data and prior latent statistics (respectively).

The proposed IGAN architecture is not as greedy as it may appear at first sight. It shares the same networks (G and D) as the classical GAN, and only requires the addition of a data Encoder E. Whereas AEGAN [16] was built on CycleGAN [21] grounds, IGAN may be related to the DiscoGAN [22] architecture with the latent space replacing one of the two image data spaces, and with the major difference that it uses a unique Discriminator instead of the two Discriminators traditionally used in such unpaired image-to-image translators (one per image side).

However, alignment of the latent and image distributions is not sufficient to guarantee the element wise latent or data reconstruction. A cycle consistency loss is then introduced to learn the data-to-latent pairing. Let *fake* data $\tilde{X}_p$ be re-encoded by E to $\tilde{Z}_p = E\big(G(Z_p)\big)$ as reconstructed/autoencoded latent priors. Thus, a regularization term $L_k\big(\tilde{Z}_p, Z_p\big)$ is added to the Encoder and Generator loss to ensure that the reconstruction errors of encoded latents follow a simple distribution (i.e. Gaussian when using L_2 norm). With a sigmoïd activated Discriminator, the losses to be minimized by the different networks translate into Equation. (2).

$$\begin{cases} Loss(D) = 3 * \mathbb{E}_{X_r, Z_p}\big[-\log(D(X_r, Z_p))\big] + \mathbb{E}_{X_r, Z_p}\big[-\log(1 - D(G(Z_p), E(X_r)))\big] \\ \qquad + \mathbb{E}_{X_r, Z_p}\big[-\log\big(1 - D(G(E(X_r)), E(G(Z_p)))\big)\big] + \mathbb{E}_{X_r, Z_p}\big[-\log\big(1 - D(G(E(G(Z_p))), E(G(E(X_r))))\big)\big] \\ Loss(E, G) = \mathbb{E}_{X_r, Z_p}\big[-\log\big(D(G(Z_p), E(X_r))\big)\big] + \mathbb{E}_{X_r, Z_p}\big[-\log\big(D(G(E(X_r)), E(G(Z_p)))\big)\big] \\ \qquad + \mathbb{E}_{X_r, Z_p}\big[-\log\big(D(G(E(G(Z_p))), E(G(E(X_r))))\big)\big] + \propto * \mathbb{E}_{Z_p}\Big[\big\|E(G(Z_p)) - Z_p\big\|^2\Big] \end{cases} \qquad (2)$$

Where $\propto$ is a hyper-parameter weighting. Additionally, a cycle consistent data reconstruction loss between X_r and $\tilde{X}_r = G\big(E(X_r)\big)$ may also be added to help the data-to-latent pairing process. But since the L_k norm is often suboptimal on raw data and may generate noisy reconstructions, an alternative is to minimize the real latent reconstruction error (e.g. $L_k\big(E(G(E(X_r))), E(X_r)\big)$, or the difference between some intermediate Encoder layers activations when fed with X_r and $\tilde{X}_r$.

Upon convergence, the Generator and Encoder become invertible relatively to each other when fed with latents complying with prior distribution, or data generated from prior. It is worth noting that the reconstruction of real data will still align with the real data distribution, but will ignore details that are not consistent with the somehow *compressed* distribution of real data learned by IGAN. Nevertheless, the simplified latent distribution extracted from real data will be very useful since it has been self-organised by IGAN to span a maximum of the full prior space. Therefore, *close* data in the data space (in the semantic sense) should be encoded to *close* latents (in L_2 sense), and *close* latents should generate *close* data.

2.2 Implementation choices

It is worth noting that the IGAN architecture do not impose a specific layer implementation for the neural networks, and may be tested with the ones previously suggested in Refs [9] to [19], using resnet nodes [15], attention layers, or even the latest transformers like in TransGAN [24].

Since applications will be shown Section 3 on image datasets with mid-size image dimensions (e.g. 64*64 pixels), a very classical DCGAN implementation [23] is chosen with [Stride 2 Convolution, BatchNorm, Relu] down-sampling blocks for the data Encoder E network, and [Stride 2 Transposed Convolution, BatchNorm, Relu activation] up-sampling blocks for the Generator G [1]. The Discriminator D is built with

the same down-sampling blocks as the Encoder for the data input branch; a secondary latent concatenation layer is added for the latent input branch, as well as a simple final Fully Connected layer with a single output.

With conventional GANs, it is often hard to estimate the convergence of the optimization, and stopping criteria remain unclear. With IGAN, multiple performance and convergence indicators are available:

- The Discriminator scores monitor the GAN balance within the latent-data generation process.

- The prior latent reconstruction error (e.g. $L_k\big(E(G(Z_p)), Z_p\big)$) measures the latent space consistency and Encoder/Generator bijective invertability.

- The *fake* data reconstruction error (e.g. $L_k\big(G(Z_p), G(E(G(Z_p)))\big)$) may also be used to test the generated data space consistency and Encoder/Generator bijective invertability.

- Real data reconstruction error (e.g. $L_k(X_r, G(E(X_r)))$) should be used cautiously since outliers or statistically non-coherent details will be ignored from reconstructed data. The real latent reconstruction error (e.g. $L_k\big(E(G(E(X_r))), E(X_r)\big)$) may be used as a substitute.

3.0 APPLICATIONS

3.1 Data and latent reconstruction

The AFRL SAMPLE database is a recently released collection of simulated Synthetic Aperture Radar (SAR) signatures of ground targets paired with MSTAR image chips [25]. The dataset contains ten targets observed at fixed incidences (15° to 17°) for 360° span of azimuth orientations. IGAN was trained on the real SAR image chips and the reconstruction of both trained and generated images is shown Figure 2. It should be noticed that this training dataset is very small (less than 6000 images), which may explain some punctual reconstruction discrepancies. However, IGAN managed to generate and reconstruct fairly plausible target signatures, including the variety of background levels, targets orientations and shadows.
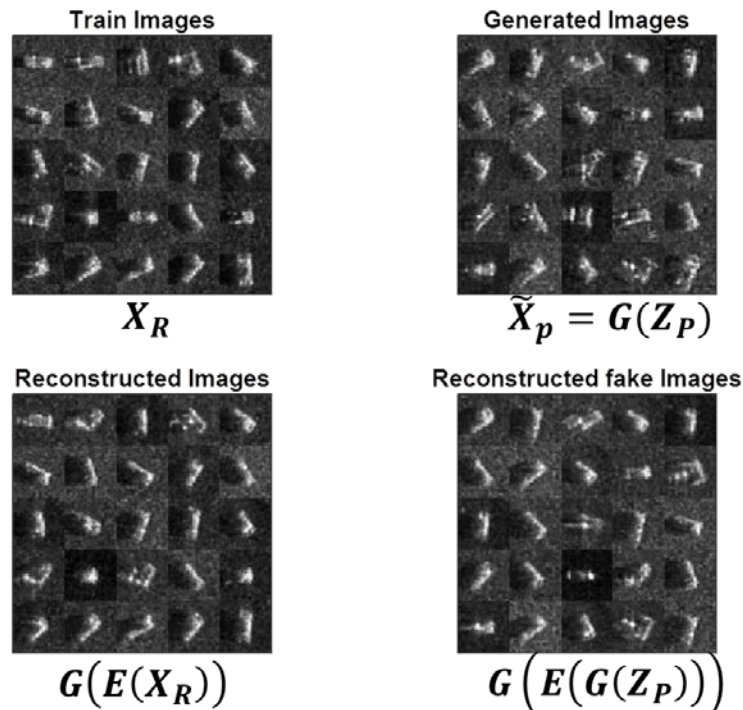

Figure 2: SAMPLE dataset [24]: examples of SAR image chips used for IGAN training (up left), reconstruction of training images (down left), some images generated by IGAN (up right), and reconstruction of generated images (down right)

The filtering effect of non-statistically relevant details and outliers is more obvious when IGAN is trained on the ANIME-FACE dataset [26] (over 60000 images of manga faces): for instance glasses, hairbands or fingers are not taken into account in the generation or reconstruction process as expected from their relative scarcity among the training dataset images (cf. Figure 3).
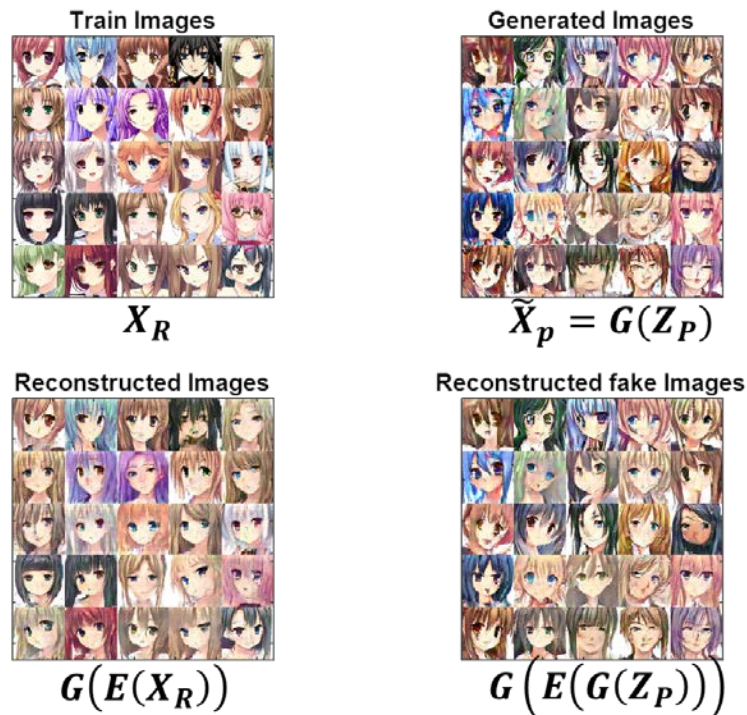
Figure 1: auto-encoded reconstructions of real and generated images with IGAN trained on ANIME-FACE dataset [26]

3.2   Self-supervised learning properties

IGAN was trained on the MNIST digits database [28] and the t-distributed Stochastic Neighbour Embedding (t-SNE) statistical method [27] was used to visualize the inferred latents distribution for Train and Test images (cf. Figure 4). Note that this training was done with no supervision, i.e. without any use of the digits label or even the number of classes. Figure 4 results show that latents belonging to the same digit class form relatively compact clusters on train set, and that we remarkably find the same clusters on full test set (simple and hard cases mixed). Some clusters boundaries are even clearly separated from others. This highlights the nice self-organisation property of the latent space generated by IGAN, with close latents corresponding to semantically close data. A fully un-supervised classification and class cardinality estimation would of course require an automatic clustering post-processing. Nevertheless, this example shows that IGAN presents some competitive properties with semi-supervised methods like InfoGAN [6], disentangling ones like BetaVAE [7], or clustering others like ClusterGAN [5].
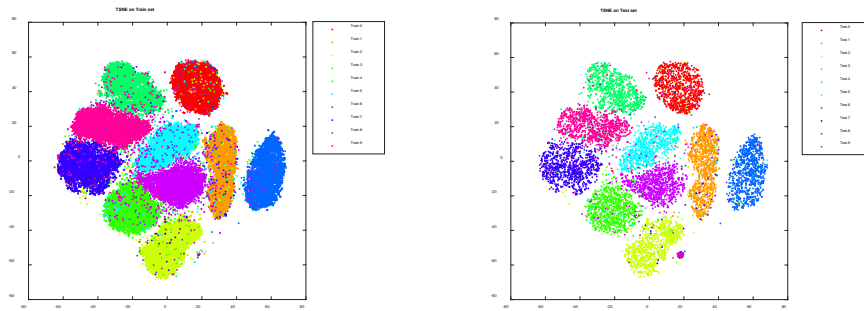
**Figure 2: IGAN trained on MNIST dataset [28]: t-SNE of latents generated on training set (left),
and on test set (right), both with post-training color visualisation of digits ground truth**

3.3 Latent space exploration and data transformations

The self-organization provided by IGAN enables an easy computation of some classical interpolations and arithmetic in the latent space. As an illustration, Figure 5 presents the bijective transformation of "serious brunettes into smiling blondes and vice-versa" with IGAN being trained on the CelebA dataset [29]. The hair colour and smile attribute tags were used to select the desired image/latent couples, compute the mean latents corresponding to either characteristic, and do the subtract/add arithmetic with the complementary mean latents to obtain the cross-characteristic images reconstructions. As expected, the transformation retains the facial personality and the head position of the original images, while filtering out the non-statistically relevant details/outliers (background, hat, scarf, arm, boyfriend, etc.).


**Figure 3: CelebA [29] images with "serious & brunette" characteristics (top left), "smiling &
blonde" (bottom left), IGAN reconstruction of "serious & brunette" changed into "smiling &
blonde" (top right), and IGAN reconstruction of opposite change (bottom right)**

3.4 Shared latent space and multi-domain transformations

Several publications have already explored the exploitation of a common or shared latent space, especially in the context of multi-modal image transformations, e.g. UNIT [30], MUNIT [31] or XGAN [32]. Most of the

proposed solutions are limited to a dual image domain translation, whereas the IGAN architecture is *de facto* well suited for multiple data domains change (not limited to images) on a single shared latent space. Moreover, IGAN does not require a simultaneous training of paired domains, as each domain keeps its own separated Encoder and Generator training. Since IGAN spans the entirety of the latent space for every domain, any data in one domain will be turned into a plausible data for any other domain. Figure 6 illustrates the reconstruction of domain A (ANIME-FACE [26]) and domain B (CelebA [29]) images through successive cross-domain translations $\{G_A(E_B(G_B(E_A(A))\}$ and $\{G_B(E_A(G_A(E_B(B))\}$. Perfect reconstructions are not expected for data going through so many networks, but it is noticeable that the main image characteristics remain clearly unchanged. Domains that share common semantic features should probably be merged into a single IGAN training to avoid unrelated latent correspondences between domains (e.g. non corresponding head orientations in both domains in Figure 6 example). Dedicated optical images manipulations often use separated latent spaces for style and content [33], and the IGAN framework could be advantageously adapted to this context in the future.
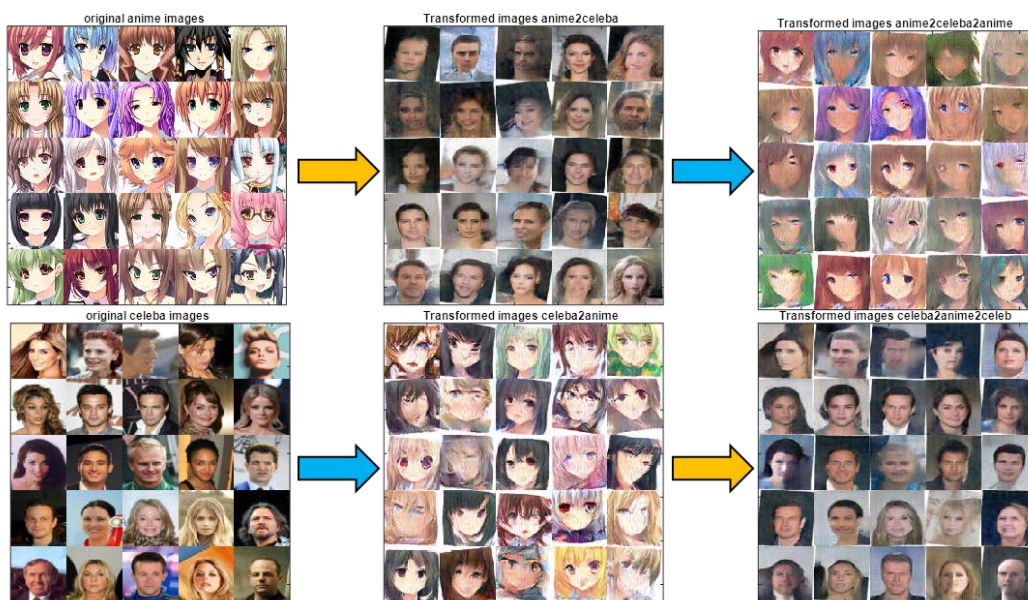
Figure 6: Reconstruction of ANIME-FACE [25] and CelebA [28] images through successive cross-domain (ANIME-FACE ←→ CelebA) translations

4.0   CONCLUSIONS

IGAN builds a symmetric entangled adversarial game and a bijective auto-encoder in both data and latent spaces with multiple advantages:

- It balances, stabilizes the generation of data (and latents) and avoids mode collapse, while keeping the data generative quality. It provides measurable performance and convergence indicators.

- It spans generated latents on the full prior space, giving access to a wider generated space and thus to more variability of generated data.

- It provides an un-supervised self-organization of the latent space and enables the reconstruction of real and generated images and latents.

- It remains simple and frugal adding only an Encoder network to the Vanilla GAN.

This leads to numerous latent based applications for complex high dimensional data processing, such as

semantic attributes arithmetic and interpolation, self-supervised learning, or multi domain translations (some other examples are shown in [34]). For future work, a conditional version (CondIGAN) and fractionable latent space extensions are being considered. The original adversarial game developed for IGAN also suggests revisiting DiscoGAN and similar methods from a single Discriminator perspective using channel wise concatenated images from both domains as input.

6.0 REFERENCES

[1] Vignaud, L. (2022). Inferent and Generative Adversarial Networks (IGAN), MATLAB Central File Exchange. Retrieved July 1, 2022. https://fr.mathworks.com/matlabcentral/fileexchange/100923-inferent-and-generative-adversarial-networks-igan

[2] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S. et al. (2014). Generative adversarial nets. Advances in Neural Information Processing Systems 27 (NIPS 2014).

[3] Kingma D.P. and Welling M. (2013). Auto-Encoding variational Bayes (2013). https://arxiv.org/abs/1312.6114

[4] Kodali N., Abernethy J., Hays J., and Kira Z. (2017). On convergence and stability of GANs. arXiv:1705.07215

[5] Mukherjee, S., Asnani, H., Lin, E., and Kannan, S. (2018). Cluster-GAN: Latent space clustering in generative adversarial net-works. CoRR, abs/1809.03627.

[6] Chen, X. Duan, Y., Houthooft, R., Schulman, J., Sutskever, I. and Abbeel, P. (2016). Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: NIPS'16: Proceedings of the 30th International Conference on Neural Information Processing Systems, pp. 2180-2188.

[7] Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A. (2017). beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework, ICLR 2017 Conference.

[8] Tiu, E. (4 Feb 2020). Understanding latent space in machine learning. Towards Data Science. https://towardsdatascience.com/understanding-latent-space-in-machine-learning-de5a7c687d8d

[9] Boesen, A., Larsen, L., Sønderby, S.K., Larochelle, H., and Winther, O. (2016). Autoencoding beyond pixels using a learned similarity metric. In: Proceedings of the 33rd International Conference on International Conference on Machine Learning – Volume 48 (ICML'16), JMLR.org, pp. 1558-1566.

[10] Donahue J., Krähenbühl, P., and Darrell, T. (2017). Adversarial feature learning. ICLR 2017 Conference. ArXiv abs/1605.09782.

[11] Dumoulin, V., Belghazi, I, Poole, B., Mastropietro, O., Lamb, A., Arjovsky, M., et al. (2017). Adversarially learned inference. ICLR 2017 Conference. ArXiv abs/1606.00704.

[12] Li C., Liu H., Chen C., Pu Y., Chen L., Henao R., and Carin L. (2017). ALICE: Towards understanding adversarial learning for joint distribution matching. In: Advances in Neural Information Processing Systems (NIPS 2017).

[13] Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2018). It takes (only) two: Adversarial generator-encoder networks. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-

18), pp. 1250-1257.

[14]   Huang H., Li, Z., He, R., Sun, Z., and Tan, T. (2018). IntroVAE: Introspective Variational autoencoders for photographic image synthesis. In: Advances in Neural Information Processing Systems 31 (NeurIPS 2018).

[15]   Donahue, J. and Simonyan, K. (2019). Large scale adversarial representation learning. In: Advances in Neural Information Processing Systems 32 (NeurIPS 2019).

[16]   Lazarou, C. (2020). Autoencoding generative adversarial networks, ArXiv abs/2004.05472

[17]   Li, A.H., Wang, Y., Chen, C., and Gao, J. (2020). Decomposed adversarial learned inference. ArXiv abs/2004.10267 2020.

[18]   Zhu, J., Zhao, D., Zhang, B., and Zhou, B. (2019). Disentangled inference for GANs with latently invertible autoencoder. arXiv:1906.08090.

[19]   Pidhorskyi, S., Adjeroh, D.A., and Doretto, G. (2020). Adversarial latent autoencoders. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2020.

[20]   Karras T., Laine S., and Aila T. (2018). A style-based generator architecture for generative adversarial networks, arXiv:181204948.

[21]   Zhu, J.Y., park, T., Isola, P., and Efros, A.A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2242-2251.

[22]   Kim, T., Cha, M., Kim, H., Lee, J.K., and Kim, J. (2017). Learning to discover cross-domain relations with generative adversarial networks. International Conference on Machine Learning (ICML) 2017.

[23]   Radford, A., Metz, L., and Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. CoRR abs/1511.06434.

[24]   Jiang, Y., Chang, S., and Wang, Z. (2021). TransGAN: Two pure transformers can make one strong GAN, and that can scale up. 35th Conference on Neural Information Processing Systems (NeurIPS 2021).

[25]   Lewis, B., Scarnati, T., Sudkamp, E., Nehrbass, J., Rosencrantz, S., and Zelnio E. (2019). A SAR database for ATR development: Synthetic and Measured Paired Labeled Experiment (SAMPLE). Proceedings of SPIE Algorithms for Synthetic Aperture Radar XXVI; Orlando, USA 2019.

[26]   Mckinsey666 (2020). Anime-face-dataset. https://github.com/bchao1/Anime-Face-Dataset

[27]   Van der Maaten, L.J.P, and Hinton, G.E., (2008). Visualizing high-dimensional data using t-SNE. (2008). Journal of Machine Learning Research, 9, November, pp. 2579-2605.

[28]   LeCun, Y., Cortes, C., and Burges, C.J.C. (1999). The MNIST database of handwritten digits. Courant Institute, NYU, Google Labs, New York, Microsoft Research, Redmond.

[29]   Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. International Conference on Computer Vision (ICCV) 2015, pp. 3730-3738.

[30]   Liu, M.Y., Breuel, T., and Kautz, J. (2017). Unsupervised image-to-image translation networks.

International Conference on Neural Information Processing Systems (NIPS 2017).

[31]   Huang, X., Liu, M.Y., Belongie, S., and Kautz, J. (2018). Multimodal unsupervised image-to-image translation. ArXiv abs/1804.04732.

[32]   Royer A., Bousmalis, K., Gouws, S., Bertsch, F., Mosseri, I., Cole, F., and Murphy, K. (2020). XGAN: Unsupervised image-to-image translation for many-to-many mappings. ArXiv abs/1711.05139.

[33]   Abdal R., Qin, Y., and Wonka, P. (2019). Image2StyleGAN: How to embed images into the StyleGAN latent space?' 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 4431-4440.

[34] Vignaud, L. (2021). IGAN Discriminative & Generative architecture will "crack open the bone and suck out the substantific marrow" of your favorite dataset. https://www.linkedin.com/pulse/igan-discriminative-generative-architecture-crack-open-luc-vignaud/