Data augmentation of 3D brain environment using Deep Convolutional Refined Auto-Encoding Alpha GAN

Alice Segato*, Valentina Corbetta†, Marco Di Marzo‡, Luca Pozzi§ and Elena De Momi¶
Politecnico di Milano
Piazza Leonardo da Vinci 32, Milan, Italy
* Email: alice.segato@polimi.it
† Email: valentina.corbetta@mail.polimi.it
‡ Email: marco.dimarzo@mail.polimi.it
§ Email: luca6.pozzi@mail.polimi.it
¶ Email: elena.demomi@polimi.it

***Abstract*—Learning-based methods represent the state of the art in path planning problems. Their performance, however, depend on the number of medical images available for the training. Generative Adversarial Networks (GANs) are unsupervised neural networks that can be exploited to synthesize realistic images avoiding the dependency from the original data. In this paper, we propose an innovative type of GAN, Deep Convolutional Refined Auto-Encoding Alpha GAN, able to successfully generate 3D brain Magnetic Resonance Imaging (MRI) data from random vectors by learning the data distribution. We combined a Variational Auto-Encoder GAN with a Code Discriminator to solve the common mode collapse problem and reduce the image blurriness. Finally, we inserted a Refiner in series with the Generator Network in order to smooth the shapes of the images and generate more realistic samples. A qualitative comparison between the generated images and the real ones has been used to test our model's quality. With the use of three indexes, namely the Multi-Scale Structural Similarity Metric, the Maximum Mean Discrepancy and the Intersection over Union, we also performed a quantitative analysis. The final results suggest that our model can be a suitable solution to overcome the shortage of medical images needed for learning-based methods.**

I. Introduction

Recently developed prototypes of steerable needles represent a breakthrough in keyhole neurosurgery as they can reach targets behind sensitive or impenetrable areas [1].

Different path planning approaches (combinatorial, sampling-based, potential field, levels set methods) can be applied [2]. Classical methods are affected by a trade-off between completeness and efficiency, a limitation that recent learning-based (LB) methods try to overcome.

LB methods for path planning leverage the power of Deep Learning (DL) algorithms to learn how to take actions in order to find the best trajectory. The problem in working with DL algorithms is that their successful training is conditioned to the availability of a large number of data, which in our case consist of medical images. An issue that even the proliferation of publicly accessible data-sets cannot solve.

While classical data augmentation techniques (e.g. flip, rotation) are highly dependent on the original data, Generative Adversarial Networks (GANs) [3] are unsupervised neural networks that can be exploited to synthesize realistic images. Data augmentation using GANs has proven itself to be efficient in training classification models, such as [4]. However, currently available architectures do not allow for the generation of MRI volumes with a sufficient resolution to train a deep-learning model for path planning [5], [6]. The fundamental concept behind a GAN is the one of adversarial training of two networks: a Generator and a Discriminator. The Generator is trained to create new samples in order to fool the discriminator, which evaluates the authenticity of the images. This family of networks has been widely exploited in the medical field, being successfully trained either to perform image-to-image translation (denoising, reconstruction, cross-modality synthesis) as well as noise-to-image translation (unconditional synthesis) [7].

This work is oriented towards this last scope, taking it to the three-dimensional domain. The goal is the generation of 3D Magnetic Resonance Imaging (MRI) images of the brain to train path planning agents for a steerable surgical needle. Such an aim is chased through a Deep Convolutional Refined (DCR) α-GAN (DCR-α-GAN). The accomplishment of our aim would pave the way to the synthesis of brains suffering of specific pathological conditions. This, in turn, would tackle problems such as the shortage of these samples in public dataset and class imbalance when training a LB path planner on both healthy and pathological brains.

The outline of this paper is the following: in section II, we discuss the state of the art. In Section III, we present the proposed methodology, while we report the experimental results in Section IV. Finally, Section VI draws the conclusions and gives insights on future work.

II. State of the Art

In recent years, data augmentation exploiting GANs has become more and more common, given its promising results. In particular, in the context of MRI, it has been prominent

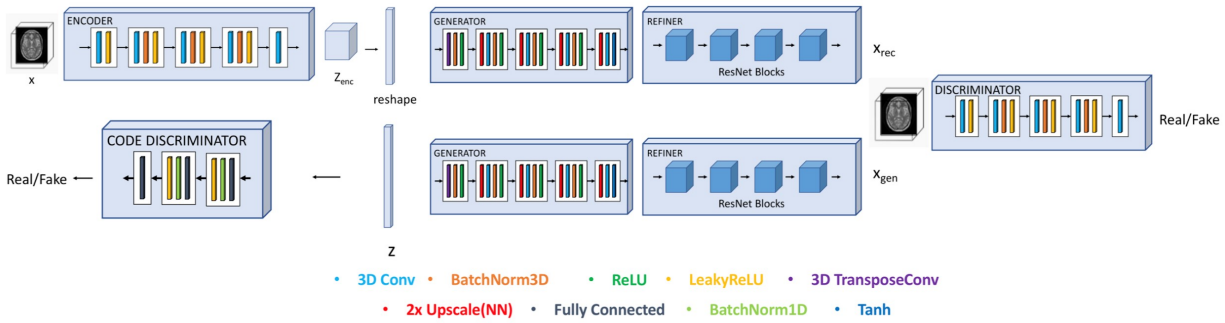

Fig. 1. Architecture of our proposed model. x is the real input 3D volume, z_{enc} is the output from the encoded image, x_{rec} and x_{gen} are respectively the output of the Refiner fed with the output of the Generator fed with z_{enc} and the output of the Refiner fed with the output of the Generator fed with z, which is the random noise vector obtained from a normal distribution.

the use of image transfer techniques, for instance, to synthesize X-ray Computed Tomography (CT) images from MRI samples [8]. Data augmentation through GANs could become critical in clinical practice, as in many countries medical data are considered sensitive and are protected by restrictive protection laws. Therefore, GANs may avoid the direct usage of actual data, which could be used instead to validate the results from GANs. Indeed, the generation of 3D MRI volumes has made significant progress, in particular with the work of [5]: the authors compared different available architectures and proposed a new model, namely an auto-encoder GAN. This architecture showed encouraging results both in terms of diversity of the generated data and of resemblance of the real samples. We introduce to this approach an additional component, a Refiner network, which allows the synthesis of smoother images, thus providing a closer resemblance to the real data.

III. Proposed Implementation

Our network architecture is an adaptation of the α-GAN [5] to the problem of 3D instance generation. To train our model, we used the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset [9]. From the ADNI control normal group, 997 T1 structural brain images were taken. In the images present in the ADNI dataset, non-brain areas are removed using the recon-all function from the software FreeSurfer. A few pre-processing operations were applied to the images before feeding them to our GAN: first, planes with all-zero values were removed, then, volumes were resized to $64 \times 64 \times 64$ (from $256 \times 256 \times 256$). Our network combines Variational Auto-Encoder (VAE) GANs, with an additional Code Discriminator (CD) and Refiner.

A. Model architecture

The complete architecture of our network can be seen in detail in Figure 1. The Encoder consists of 5 3D Convolutional layers with $4 \times 4 \times 4$ filters. Moreover, Batch Normalization, which is a technique to improve the speed, performance and stability of artificial networks, is present after each layer, except for the first and last ones, in which is absent to maintain the originality of the input and the output. As activation function, we use LeakyReLu, a variant of the Rectified Linear Units, has output 0 if the input is less than 0, and raw output otherwise. If the input is greater than 0, the output is equal to the input; it is non-linear and has the advantage of not having any backpropagation errors unlike the sigmoid function; also for larger Neural Networks, the speed of building models based off on ReLU is very fast. The presence of the Encoder opposes to the mode collapse [10], which is a common problem when training GANs and happens when the Generator collapses, which produces limited varieties of samples. It can occur when the Generator spots one image able to fool the Discriminator: from then on, the generated images become very similar, resulting in a limited variety of output samples. The VAE solves this problem by mapping all the available training samples to the same latent space.

The network's Discriminator has a structure similar to the one of the VAE and has the function explained in Section I.

The Generator consists of 5 layers. First, we apply the resize convolution to limit the number of parameters and checkerboard artefacts. Instead of transpose convolution layers, we use conventional nearest neighbour upscale, which is the simplest and fastest implementation of image scaling technique, before convolution layers with 3×3 filters. BatchNorm and ReLU layers are added after each convolution, except for the last one, to maintain training stability. The last layer has hyperbolic tangent (Tanh) as the activation function.

Our CD network consists of three fully-connected layers. Similar to the Discriminator, LeakyReLU and BatchNorm layers are placed between each fully-connected layer. The CD is trained to distinguish between latent vectors coming from the VAE and the random ones given as input to the Generator. This adversarial process makes the probability distributions of the two latent vectors to match, reducing the image blurriness that characterizes the VAE outputs.

The architecture of the Refiner consists in four ResNet blocks [11] and is depicted in detail in Figure 2. In traditional neural networks, each layer feeds into the next layer. In a network with residual blocks, each layer feeds into the next layer and directly into the layers about 2–3 hops away. The presence of skip connections reduces the vanishing gradient

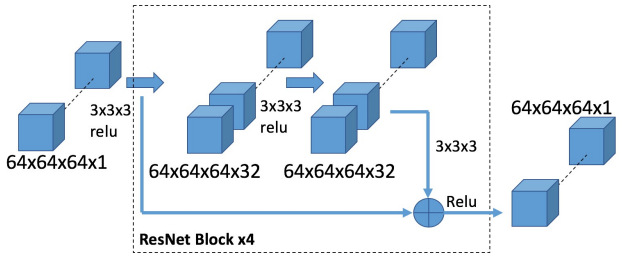

Fig. 2. Detailed architecture of the Refiner network, which consists of 4 ResNet blocks.

problem since it allows to skip the training of some layers. It smooths the shapes of the image and allows the generation of more realistic images. Our Refiner is similar to the one present in SimGAN [12]. Due to memory constraints, the number of channels has been decreased from 64 to 32.

B. Training

In the training of our network, we considered the VAE and the Generator as one network; therefore we sum the respective loss functions, and we optimize first the VAE, then the Generator, the Discriminator and lastly the CD. The optimization speed of the Generator is far slower, so the Generator is updated twice per iteration. Then, we train the Refiner separately, loading the weights of the previously trained components. To perform the training we used an NVIDIA Titan X GPU with 12GB.

C. Experimental Evaluation

The quantitative analysis considers three indexes, namely the Multi-Scale Structural Similarity Metric (MS-SSIM) [13], the Maximum Mean Discrepancy (MMD) [14] and the Intersection over Union (IoU) [15].

MS-SSIM measures the similarity between two images, and it is used to evaluate the diversity of the generated images. Differently from other methods, MS-SSIM considers phenomena that are crucial in human perception such as luminance and contrast.

Given the two images X and Y, their means μ_X and μ_Y are taken as estimates of the luminance of the images and combined to give the luminance comparison:

$$l(X,Y) = \frac{2\mu_X\mu_Y + C_1}{\mu_X^2 + \mu_Y^2 + C1}$$

The variances of the images, σ_X and σ_Y, account for their contrast and allow to compute the contrast comparison:

$$c(X,Y) = \frac{2\sigma_X\sigma_Y + C_2}{\sigma_X^2 + \sigma_Y^2 + C_2}$$

Finally, a structural similarity term is obtained looking at the X and Y covariance, σ_{XY}:

$$s(X,Y) = \frac{\sigma_{XY} + C_3}{\sigma_X\sigma_Y + C_3}$$

The terms C_1, C_2 and C_3 in the above expressions are constants depending on the range of the pixel values. The single-scale Structural Similarity Metric (SSIM) is computed as the product of the three terms, which relative importance can be fixed by as many exponents:

$$\text{SSIM}(X,Y) = [l(X,Y)]^\alpha \cdot [c(X,Y)]\beta \cdot [s(X,Y)]^\gamma$$

In the multi-scale SSIM, the contrast and the structural comparison are computed on iteratively downsampled versions of the two images, while the luminance term is computed only at the very last iteration, hence MS-SSIM is given by the following expression:

$$\text{MS-SSIM}(X,Y) = [l_J(X,Y)]^\alpha \cdot \prod_{j=1}^{J} [c_j(X,Y)]^\beta [s_j(X,Y)]^\gamma$$

The MMD is a distance-measure between distributions (P(X) and Q(Y)) defined as the squared distance between their embeddings in the reproducing kernel Hilbert space. In such a Hilbert space of functions, if two functions are close in the norm, then they are also pointwise close. MMD is computed as the squared distance between the embeddings of the distributions

$$\text{MMD}(P,Q) = ||\mu_X - \mu_Y||^2_{\mathcal{H}}$$

with a low score of MMD indicating closeness between the two distributions.

The IoU, also known as Jaccard index, is a statistic to evaluate the similarity of two sets (X and Y) as the ratio between the number of elements they have in common and the total number of elements:

$$\text{IoU}(X,Y) = \frac{|X \cap Y|}{|X \cup Y|}$$

In the present case, the real and the generated samples are compared at a voxel level, with high scores of IoU indicating closeness between the two distributions.

IV. Results

To perform our test, we retrained on our GPU the model from [5], to have a fair comparison with our samples. In Figure 3D slices along the three principal axes of a sample generated by the architecture of [5] are shown. Figure 3C depicts the samples synthesised by our architecture, which are compared with the real 3D MRI in Figure 3A and 3B. The qualitative comparison draws the attention to the network capability of producing realistic brain volumes, even though a difference in the level of detail can be spotted w.r.t the real samples. W.r.t the samples by [5], our images show a better capability to capture the details of the MRI volume.

For the quantitative evaluation, the similarity scores of 1000 pairs of generated images are computed and averaged to obtain an overall value of MS-SSIM$_{fake}$ = 0.9991 for our architecture and MS-SSIM$_{fake}$ = 0.6006 for the work by [5]. The MMD and IoU are evaluated comparing an image from the training set and a generated image. The procedure is repeated 100 times and the scores are then averaged. As a result, our architecture is characterized by MMD $= (0.2240 \pm 0.0008) \cdot 10^4$ and a IoU $= 0.6852 \pm 0.0024$; the model from [5] has an MMD $= (0.5932 \pm 0.0004) \cdot 10^4$ and a IoU $= 0.3668 \pm 0.0016$.

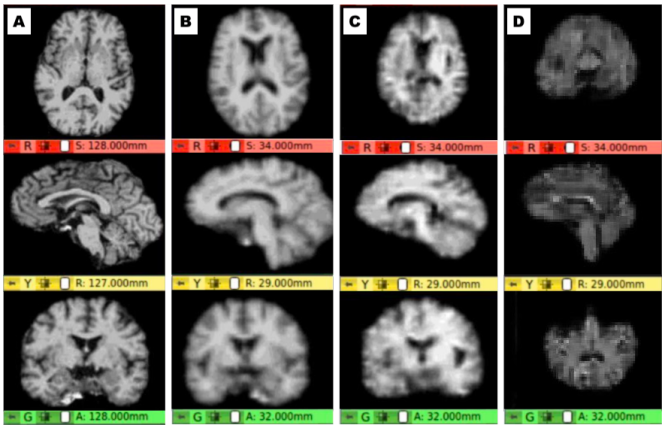

Fig. 3. (**A**) Real 3D MRI from the ADNI dataset control normal group. (**B**) Real 3D MRI with the same resolution of the fake images. (**C**) Fake 3D MRI synthesised by our trained model. (**D**) Fake 3D MRI synthesised by the model in [5]

V. Discussion

The gap in accuracy resulting from the qualitative comparison can be explained with the reduced dimensions of the output images, as it can be noticed comparing the fake MRI with a real downsampled one in Figure (3, centre).

The value of the MS-SSIM is close to 1, which indicates a not excellent diversity between generated images. This parameter is worse than the one of [5], therefore our architecture is characterised by a lower capability of generating diverse samples. On the other hand, the values of the MMD and the IoU suggest a better realism of the generated images by our model w.r.t the work by [5]. For these reasons, future works related to the network will focus on the improvement of the Encoder or any other block involved in the mode collapse phenomenon. The quantitative analysis shows two controversial aspects of the our model.

The obtained results are encouraging, but their low resolution ($64 \times 64 \times 64$) prevents them from being used to train a LB path planner. In fact, the lack of detail impedes the reproduction of small structures, such as vessels, with sufficient precision. This would make vessel avoidance impossible, missing one of the main goals in path planning.

VI. Conclusions and Future Work

In this paper, we have presented a GAN architecture able to reconstruct 3D brain environment using a Deep Convolutional Refined α GAN starting from random vectors. The fake 3D MRI synthesized by our model shows good realism and sufficient variety. The action of the refiner is effective in producing images that are similar to the real ones.

The diversity between generated MRIs is fundamental to consider the network as a countermeasure to the shortage of training data for ML applications. Together, these features point out our architecture as a suitable solution to provide the necessary amount of training samples to ML-based curvilinear trajectory planners for steerable needles.

A forthcoming step is the use of a super-resolution network to take the generated samples to $256 \times 256 \times 256$ pixels, improving the level of detail of the augmented data. As additional future works, we plan to train our network on diseased images, always taken from the ADNI dataset, to test its ability to generate this kind of samples. Moreover, we would like to try and use our GAN to perform style transfer from control to diseased images. This technique could solve the issue of class imbalance in classification problems by creating new samples of the class with the lowest number of instances [16].

Acknowledgment

This project has received funding from the European Union Horizon 2020 research and innovation program under grant agreement No 688279.

References

[1] M. Scali, T. P. Pusch, P. Breedveld, and D. Dodou, "Needle-like instruments for steering through solid organs: A review of the scientific and patent literature," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 231, no. 3, pp. 250–265, 2017.

[2] A. Gasparetto, P. Boscariol, A. Lanzutti, and R. Vidoni, "Path planning and trajectory planning algorithms: A general overview," in *Motion and operation planning of robotic systems*. Springer, 2015, pp. 3–27.

[3] 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.

[4] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," in *2018 international interdisciplinary PhD workshop (IIPhDW)*. IEEE, 2018, pp. 117–122.

[5] G. Kwon, C. Han, and D.-s. Kim, "Generation of 3d brain mri using auto-encoding generative adversarial networks," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2019, pp. 118–126.

[6] V. Sorin, Y. Barash, E. Konen, and E. Klang, "Creating artificial images for radiology applications using generative adversarial networks (gans)–a systematic review," *Academic Radiology*, 2020.

[7] X. Yi, E. Walia, and P. Babyn, "Generative adversarial network in medical imaging: A review," *Medical image analysis*, vol. 58, p. 101552, 2019.

[8] B. Kaiser and S. Albarqouni, "Mri to ct translation with gans," *arXiv preprint arXiv:1901.05259*, 2019.

[9] Alzheimer's disease neuroimaging initiative. [Online]. Available: http://adni.loni.usc.edu/

[10] M. Rosca, B. Lakshminarayanan, D. Warde-Farley, and S. Mohamed, "Variational approaches for auto-encoding generative adversarial networks," *arXiv preprint arXiv:1706.04987*, 2017.

[11] S. Targ, D. Almeida, and K. Lyman, "Resnet in resnet: Generalizing residual architectures," *arXiv preprint arXiv:1603.08029*, 2016.

[12] 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.

[13] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *International conference on machine learning*, 2017, pp. 2642–2651.

[14] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 723–773, 2012.

[15] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3d-r2n2: A unified approach for single and multi-view 3d object reconstruction," in *European conference on computer vision*. Springer, 2016, pp. 628–644.

[16] H. Cho, S. Lim, G. Choi, and H. Min, "Neural stain-style transfer learning using gan for histopathological images," *arXiv preprint arXiv:1710.08543*, 2017.