Diffeomorphic Unet for Lung CT scans Deformable Registration

Shahine Bouabid Simon Lajouanie Valentin Tran
CentraleSupelec
firstname.lastname@student.ecp.fr

Abstract

Medical imaging is a vital component of a large number of applications such as medical diagnosis but also in the area of planning or therapeutical procedures evaluation. Since information is scattered in different instances, a proper integration of distinct images is often required. One fundamental step in this journey is referred to as image registration which consists in finding a mapping from one image to another. Recent developments in this field have proven the efficiency deep learning approaches to compute such mapping. We propose in this work an attempt to use deep registration architectures based on the Luna dataset of lung CT scans and segmentation.

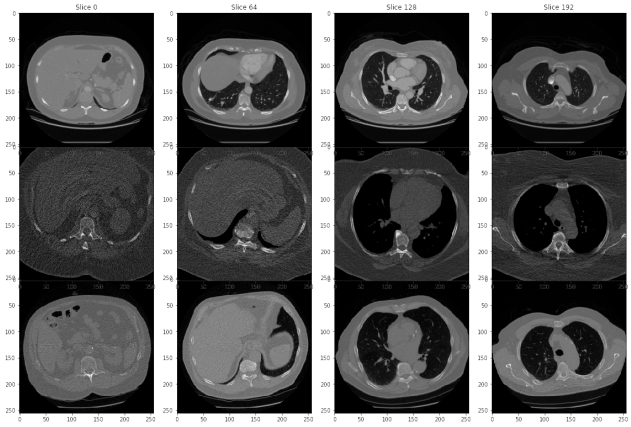

Figure 1. Example of lung slices from the CT scan Luna dataset

1. Introduction

Medical images are widely used for diagnosis, disease monitoring, treatment planning. Whether for matching images of the same patient at different times or for large scale studies, being able to match a given image to another is a critical task in radiology. An example of the use of modalities registration can be found in radiotherapy treatment planning.

These images usually refers to 3D volumes acquired by tomographic modalities such as computed tomography (CT) which makes use of X-ray measurement from different angles or magnetic resonance imaging (MRI) which rely on protons excitation by magnetic fields.

Recent advances in deep learning have proven neural networks to be a promising alternatives to classical optimization formulations of this problem as they offer a powerful end-to-end framework. Based on a source and a target scan, these networks are trained to compute an isomorphic geometrical mapping from the former to the latter. The use of convolutional neural networks (CNNs) have lately been providing robust results for medical image registration [8] with medical image acquisition modalities such as MRI. However, MRI imaging happens to be more expensive than CT scans in terms of utilities acquisition and cost per scan, and involves a longer (although not proven harmful) scanning time. Mastering CT-acquired volumes registration would hence be of great interest.

In this work, we attempt to tackle the CT scan registration for the Luna datset of lung scans[1] with convolutional architecture derived from state of the art literature in deep medical image registration to get a better understanding of the issues faced. In the meantime, we propose to include auxiliary data in registration to assess to what extent it can help improve registration performances.

2. Background

Registration is the determination of a geometrical transformation that aligns a view of an object (source) with corresponding points in another view of that object or even another object (target). By succeeding in mapping a volume onto another, registration enables cross-subject comparison regardless of the anatomical variability between patients. This is a key task in range of applications going from temporal evolution tracking of a disease to multimodal image fusion or inter-subject comparison across large populations.

Formally, registration can be explicitly stated as an energy optimization problem with regard to the transforma-

[1] https://luna16.grand-challenge.org/data/

tion. The latter energy usually decomposes into a similarity part, which answers for the closeness in terms of voxels intensity value between the target volume and the deformed source volume, and a regularization term which ensures are certain degree of smoothness of the obtained transformation. Given a source and target image pair (S, T), registration is formulated as the optimization problem :

$$\hat{\phi} = \arg\min_{\phi} \mathcal{L}(S, T, \phi) \quad (1)$$

$$= \arg\min_{\phi} \mathcal{L}_{\text{sim}}(T, \phi(S)) + \alpha R(\phi) \quad (2)$$

where $\mathcal{L}_{\text{sim}}$ denotes the refered to similarity measure between the target image and the registrated source image, R the regularization function on the transformation field and α the regularization weight. Commonly, similarity metrics are chosen amongst mean square error (MSE), cross-correlation [1] or mutual information [9]. ϕ for one, can either be specified as a displacement field [2] hence straightforwardly representing for each voxel a translation vector from its initial position to its registerd destination, or as the spatial integral of its velocity field $\nabla\phi$. The latter is particularly interesting as it preserves topology and ϕ's invertibility. Finally, the regularization function if often used to enforce a whished level of smoothness of the transformation by constraining its velocity field, i.e spatial gradient.

Given this outline, deep registration offers a solution based on the assumption that we can relevantly model the application computing transformation field out of a source-target pair by a neural network. Essentially, we can write $\phi = f_\theta(S, T)$ where f_θ denotes our neural network, and the problem stated in (2) becomes the one of the networks parameters optimization $\min_\theta \mathcal{L}(S, T, f_\theta(S, T))$.

Velocity based representation also exist [7], involving an 3D integration operation to recover the transformation out of the network's output. Given its convenient properties previously stated and the advice of [6], this is what we will focus on in this work.

3. Related work

Several recent papers have been addressing this problem with an unsupervised framework involving CNNs paired with a spatial transformation layer.

Two recent works [3], [7] propose to use a 3D-Convolutional Unet architecture to compute either displacement or velocity field which is then wrapped up to predict the registered image. Another paper [6] has been proposing to trade the unified Unet architecture for a network emphasizing the distinction between linear and deformable registration : encoding is done through multiresolution feature merging of the concatenated source-target pair and decoding is done, on the one hand, with squeeze excitation and classical convolutional cascade for the deformable part, on the other with a global average pooling for the rigid part. Splitting decoding in such a way allows to focus separately on global alignement consideration and more subtle non-linear transformation.

Both these architectures end up outputing either a displacement field ϕ or its spatial gradient, constrained with L_1 norm regularization, which is then wrapped up an adequate spatial transformation layer. Please note though that by enforcing the deformation's spatial gradient $\nabla\phi$ to have positive values via a sigmoid activation, [6] show we can elegantly retrieve the displacement field with a cumulative sum along each dimension and ensures the generation of smooth deformations.

However, current literature mainly focuses on MRI and 3D ultrasound image acquisition modalities. Others have already used dataset including CT scans [4] but only in the scope of supervised learning registration.

4. Methodology

In this section, we will first provide an overview of the Luna dataset and explain the preprocessing steps we had to realize, then motivate and detail our choice of architecture before finally describe how training was performed.

4.1. Data exploration

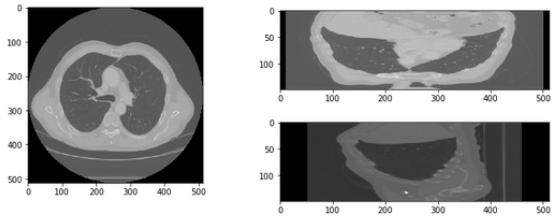

Figure 2. Example of lung slices along the 3 axis : z, x, y

Given the shape of the images, we will focus on the z axis for a better vizualisation of the preprocessing and the statistical exploration.

First of all, we tried to assess the disparity of our data. The first step was to evaluate the spectrum of sizes we were confronted to. Due to the resampling (See Section 4.2) we only have to deal with scans of the following shape : $Height * scale * scale$ (scale is a parameter we had to change due to memory limitations that varies between 64 and 256). The Picture 3 shows the (sparse) height repartition. We'll have to keep that in mind when trying to map a Scan to another. Basically, the height varies between 95 and 764 pixels.

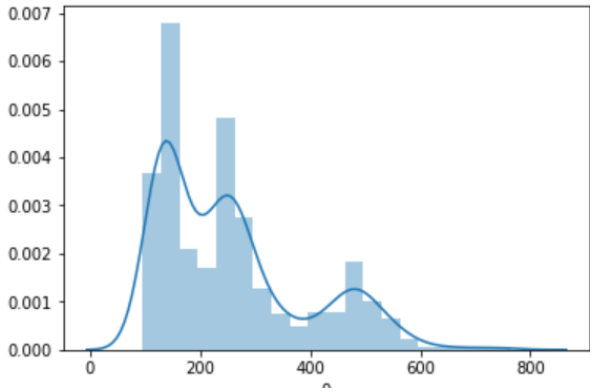

Figure 3. Height repartition for the whole dataset

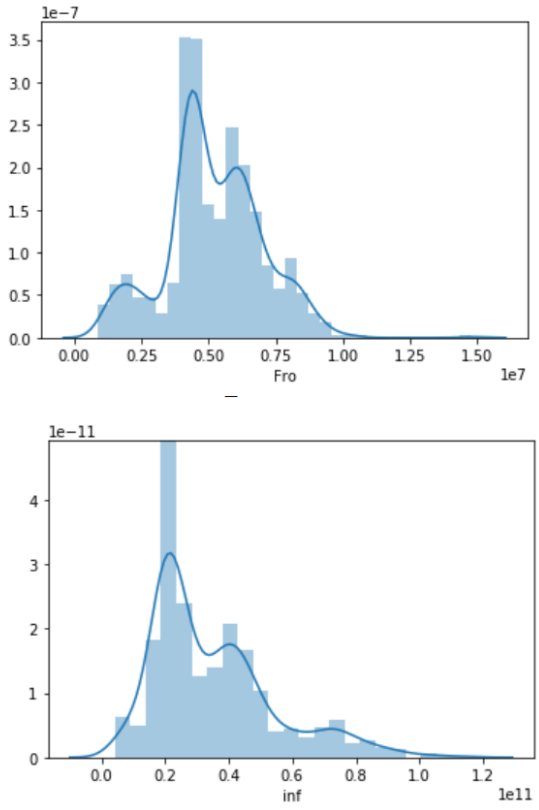

Figure 4. Froebenius and Inf norms repartition for the whole dataset

As a result of this broad distribution of heights, the intensities of the scans are quite different from one another. As seen in Figure 4, both norms result in a similar distribution. For both norms, there is a factor 10 between the smallest and highest norms. Figure 5 shows a slice of the scans with the minimum and maximum norm.

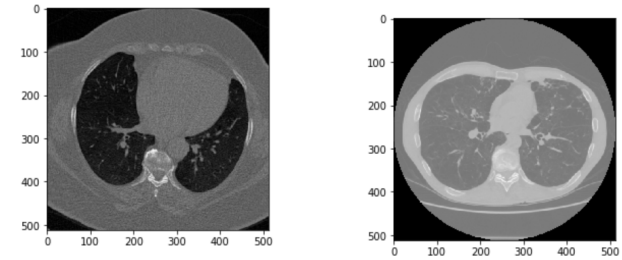

Figure 5. Slices of the most and least intense scans

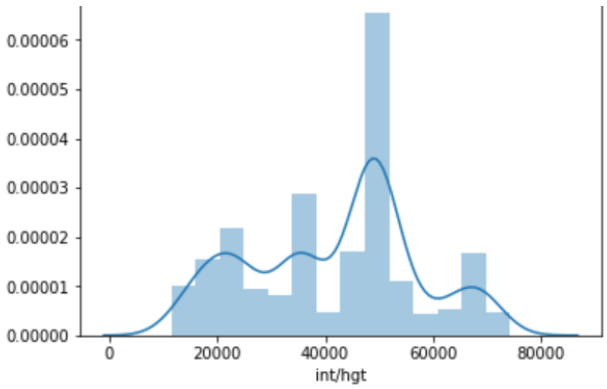

Figure 6. Repartition of the Scans' Norm per height

This analysis could lead us to believe that height is directly linked to the norm and thus, a simple rescaling would allow us to get rid of this issue. As seen in Figure 6, this phenomena is not that simple. It may be caused by a factor inherent to the scanner that took the CT Scans. We believe that normalizing the Scans as they are wouldn't result in a better accuracy. However, finding a more elaborate rule could lead to better results

4.2. Data preprocessing

The processing pipeline is contained in the `LungsLoader` class. It contains several loading function to retrieve a single scan or a generator over the dataset. Because the initialization is specific to the LUNA dataset, the exploitation of the class for another dataset would require to make it abstract and implement a child class for each dataset. The pipeline is composed of 3 steps:

- Data retrieval
- Scan loading
- Resampling, rescaling and clipping

Data retrieval This step retrieves the scans available in the `data` folder. The retrieval process would need to vary

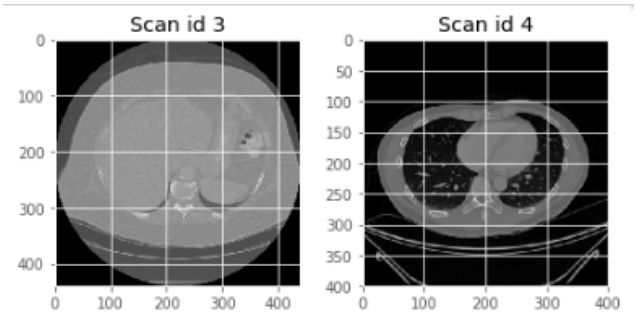

Figure 7. Example of different background layouts in scans

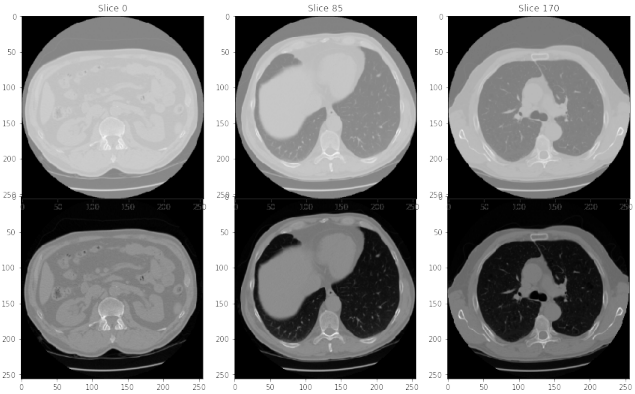

Figure 8. Example of clipping on a scan with 2 background values

for each dataset. A list of the id and location in storage of each scan is returned.

Scan loading This step consists in the conversion of the raw datafile into a python object. We chose to manipulate `numpy` arrays throughout the pipeline. We use the `SimpleITK` library to read the file and the proceed to store the numpy array and the spacing and origin information.

Resampling, rescaling The scans have different spacing, hence the need for us to resample the scans in order to have a dataset of consistently spaced scans. Therefore, we chose to resample all of the scans with a spacing of 1 along each axis. As the scans then have different dimensions, we need to rescale them to the same dimension along each axis in order to feed them to a neural network.

Clipping Finally, our dataset presents important discrepancies in the composition of the background. Some scans have highly different background layouts. For example in figure 2 we can see that the scan 4 has a uniform background whereas the scan 3 has two distinct background values that must surely correspond to outside the capture area and empty areas inside the capture area. For all the scans, we clipped the minimal value of the intensity captured to the value of the empty area inside the capture area, leading to uniform backgrounds throughout the dataset. The figure 3 presents the result of this preprocessing on different slices of a scan that presents two values for its background. As we can see, the expected operation is performed and all scans will present the same background.

4.3. Architecture

Our first approach here was to replicate the architecture proposed in [6] to attempt reproduction of the obtained performances but with lung CT scan. We hence implemented the whole network from scratch (except for the spatial transformation layer which was provided) only to eventually find out that due to the absence of pooling layers, we weren't able to fit it into memory for training. These memory limitations lead us to reconsider the autoencoding body architecture, and it was decided we would substitute it for a pooling network, namely the Unet architecture proposed in [3]. We were hence to discard at first the separation between deformable and linear registration for a unique registration network. Still, where [3] was trying to predict a displacement field with linear activation, we could stick with the velocity field approach with sigmoid activation presented in [6]. Such choice saves us the computation of large value range feature maps and ensures deformation smoothness.

Furthermore, we also wanted to assess to what extent providing the network with auxiliary information such as the scans segmentation masks would improve registration performances as suggested in [3]. We hence added an additional input to the spatial transformation layer so that we could also try to registrate the source segmentation mask against the target's one. A view of the overall architecture is depicted in Figure 4.3.

4.4. Problem simplification

During our first experiments, we ambitiously wanted to train the network to registrate against all possible pairs of scans among a training set consisting of 710 scans, thus more than $5 \cdot 10^5$ possible pairs !

As you probably would expect it, the obtained results were very disappointing and shed the light on difficulties for the network to cope with such a large spectrum of variability. Indeed, as explained above, the diversity range of the scans was too great for the network to properly registrate any kind of scan against any other. We weren't able to capture such a wide range of characteristics in a reasonable amount of time so that it would allow us to perform hyperparameters tuning.

Decision was hence taken to lower our expectation and work with a smaller training set of 98 scans. Additionally, instead of trying to registrate all possible pair of scans, it

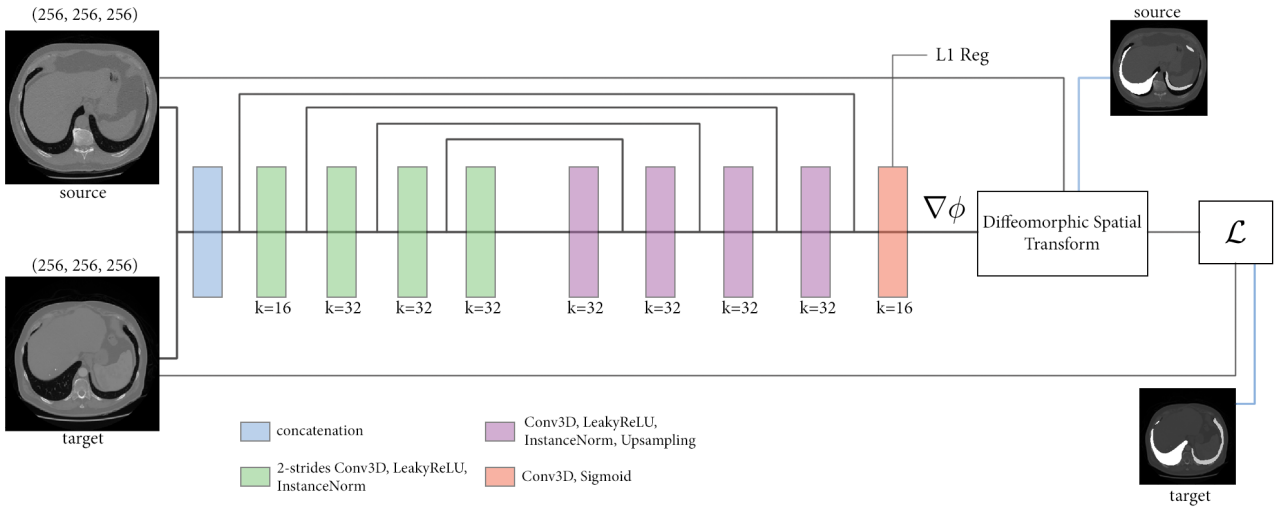

Figure 9. Overall Diffeomorphic Unet architecture as implemented

was decided to perform an atlas registration which basically consists in registering against a unique target. With such approach, as the computed transformation is to be isomorphic, we can actually theoritically still register any pair of scans. For example, let S a source object, T a target and A the registration atlas. Then, if we can compute ϕ_S and ϕ_T such that $A = \phi_S(S) = \phi_T(T)$, it comes that $T = \phi_T^{-1} \circ \phi_S(S)$.

Eventually, these simplifications brought down the number of possible source-target pairs to 98 for the training set, hence making our objective much more accessible. Although the task complexity level may have been greatly undermined, we believe the outcome of these experiments would still convey some significant insights regarding CT scan registration feasibility. In other words, if we manage to overfit this subset at the any registration-related task, then pursuing this work on a larger dataset would be worth it.

4.5. Training

The network was trained by minimizing the MSE between S and T image intensities along with a mean absolute error regularization term on the deformation flow gradient. Sticking with the formulation in (2)

$$\mathcal{L}(T, \phi(S)) = \|T - \phi(S)\|_2^2 \tag{3}$$

$$R(\phi) = \|\nabla\phi - \nabla\phi_{\mathrm{Id}}\|_1 \tag{4}$$

where $\nabla\phi_{\mathrm{Id}}$ denotes the spatial gradient of the identity deformation. As explained by [6], such regularization keeps the network away from non-smooth prediction which tend to get it stuck in a local minima.

Plus, if $\mathbb{S}$ (resp. $\mathbb{T}$) denotes the segmentation mask of S (resp. T), then if the computed registration field ϕ precisely captures anatomical correspondences, we would expect the overlap between $\mathbb{T}$ and $\phi(\mathbb{S})$ to be perfect. We translate this by using an additional dice regularization weighted by a coefficient β when trying to leverage segmentation data to assist registration. It is formulated as the negative dice score following :

$$\mathcal{L}_{\mathrm{seg}}(\mathbb{T}, \phi(\mathbb{S})) = -\frac{2|\mathbb{T} \cup \phi(\mathbb{S})|}{|\mathbb{T}| + |\mathbb{S}|} \tag{5}$$

The previous scores are naturally averaged on each batch when batch size is greater than 1, i.e for a batch $\{(S_1, T_1, \mathbb{S}_1, \mathbb{T}_1), \ldots, (S_k, T_k, \mathbb{S}_k, \mathbb{T}_k)\}$, the global loss formulation is given by :

$$\frac{1}{k}\sum_{i=1}^{k} \mathcal{L}(T_i, \phi(S_i)) + \alpha R(\phi) + \beta \mathcal{L}_{\mathrm{seg}}(\mathbb{T}_i, \phi(\mathbb{S}_i)) \tag{6}$$

However, due to memory limitation, we had to settle for a batch size of 1 and all scans were resampled to a $256 \times 256 \times 256$ grid and the autoencoder dimensioned to downsample to $32 \times 32 \times 32$ at its salient block.

We chose a conventional Adam optimizer with 10^{-3} initial learning rate and 10^{-6} decay and setup a loss-driven early stopping with 20 epochs patience. Regularization weights α and β were chosen in ranges such that they would not overwhelmingly contribute to the final loss.

Training was performed on NVIDIA Tesla K40c GPUs and we started observing convergence after nearly 120 epochs which can last up to two days of training for networks involving segmentation registration. It was implemented using Keras [5] with Tensorflow backend. Retrospectively, although it provides a very handy framework to quickly setup deep learning sessions, it is too memory

greedy to scale with such complex architectures and we should have gone for another framework.

5. Evaluation and Results

5.1. Evaluation metrics

We evaluated dense ground truth registration with MSE standing for pixel intensity distance between the registered source and the target and Cross-Correlation, which gives intensity distribution and contrast insights about the registration result.

Plus, the latter two being ill-defined since different displacement fields can actually yield similar scores, we also measured anatomical segmentation overlap with a Dice score and the Haussdorf distance. The closeness between two anatomical partitioning measures by these scores yields a very good indicator of the registration quality.

5.2. Regularization Analysis

We first wanted to assess the impact of regularization weight α over registration field gradient without auxiliary input. We hence trained the diffeomorphic Unet for different regularization weights, Table 1 shows results with regard to the chosen evaluation metrics. Globally, lower regularization weights seemed to improve segmentation-based scores but worsening MSE and CC if chosen too low. Indeed, when constraining too weakly $\nabla\phi$, we would expect to obtained registration field to be less smooth and behave more widely while allowing a better overlap of the anatomical region due to its additional degrees of freedom. The opposite is also troublesome as a too strong regularization prevents the network from exploring great deformation spectrum by enforcing it to stick to the identity transformation.

	α	10^{-4} (150 epochs)	10^{-5} (125 epochs)	10^{-6} (90 epochs)
Train (98 scans)	MSE	0.274	0.239	0.269
	CC	0.658	0.689	0.546
	Dice	0.819	0.870	0.896
	Haussdorf	75.2	65.9	95.9
Test (33 scans)	MSE	0.264	0.223	0.263
	CC	0.670	0.711	0.560
	Dice	0.829	0.872	0.899
	Haussdorf	71.9	65.8	95.5

Table 1. Atlas registration average scores without segmentation. Numbers of epochs differ because of early stopping

In practice, this leads to more "realistic" looking output but quality of the obtained registration is penalized. And indeed, willing to verify these intuitions, we trained for a few epochs the network with a strong regularization weight ($\alpha = 10$) and without regularization. When training is over-regularized, network mimics the identity transformation modulo a couple artefacts while when we don't regularize it at all, the registered image looses any form of consistency with multiple self-crossing voxels.

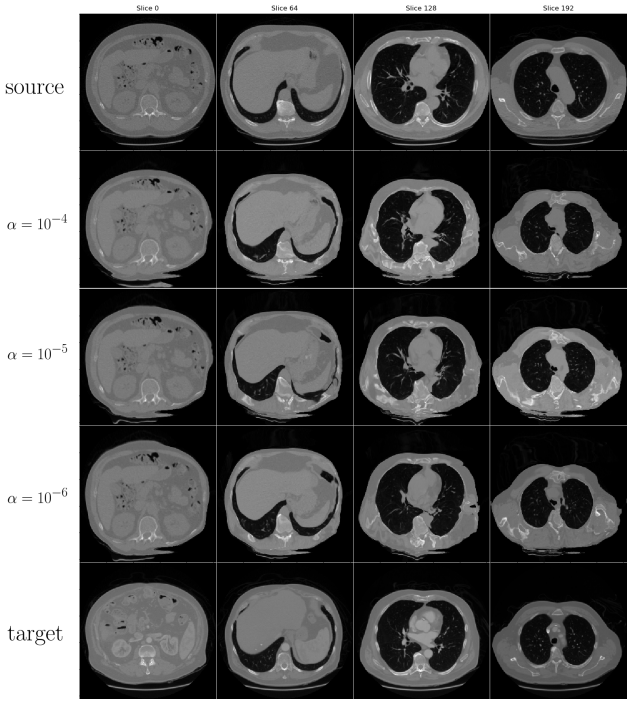

Figure 10. Example of CT lung slices registration for different values of α

Yet, we observe a pretty scattered score distribution underlying these average scores as depicted in Figure 11. This reveals that as it is, the networks struggles at generalizing its registering power to a certain category of scan.

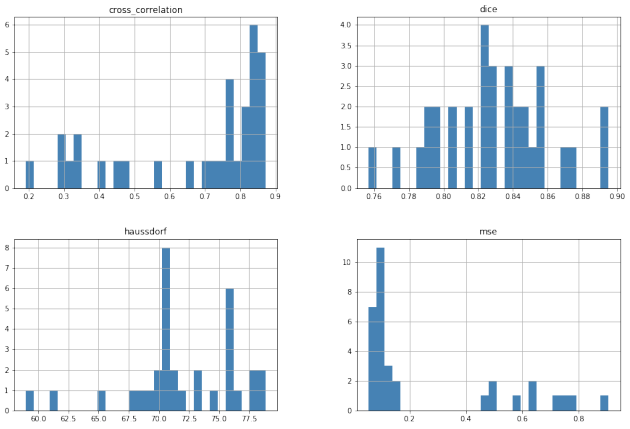

Figure 11. Scores distribution on testing set for $\alpha = 10^{-4}$

5.3. Comparing similarity measures

Although we settled to use the conventional MSE similarity loss, we were also interested in the outcome of a Cross-Correlation-driven training. As mentioned above, Cross-Correlation roughly compares intensity distributions, hence using it as a loss might help overcome the scattered intensity distribution issue pointed out by dataset exploration.

	$\mathcal{L}_{\text{sim}}$	MSE (125 epochs)	CC (140 epochs)
Train (98 scans)	MSE	0.239	0.267
	CC	0.689	0.644
	Dice	0.870	0.816
	Haussdorf	65.9	79.7
Test (33 scans)	MSE	0.223	0.251
	CC	0.711	0.664
	Dice	0.872	0.819
	Haussdorf	65.8	77.3

Table 2. Atlas registration scores for MSE and CC ($\alpha = 10^{-5}$)

Table 2 shows that not only we reach less perturbed overall registration scores for MSE with less epochs, but using MSE actually also yield better Cross-Correlation scores. Although we did not expect much from Cross-Correlation, we would have expected it to perform better at its own score.

These result underline that we better stick at first with MSE to measure similarity and that if used, Cross-Correlation should in the best case assist the MSE loss but not replace it. We insist on the fact that this is true for the specified architecure while it might actually work out better in another context such as [4].

5.4. Registration with segmentation data

In this section we present results when leveraging segmentation masks to assist registration and compare performances with previous models. Latter results can be found in table 3.

This is realized by feeding the network not only with a source volume and the atlas volume, but also with the source anatomical segmentation masks and the atlas' one. As the network is supposed to register source against atlas, it should henceforth be able the registrate properly their associated masks.

We observe that using segmentation significantly improves performances in terms of Dice score and Haussdorf distance, hence pointing out an enhanced ability to overlap anatomical regions. Furthermore, as depicted by figure 12, we observe scores distributions to be much tighter, contrasting with the previously scattered distribution.

This is actually a good surprise : although the obtained scores are not outstanding, involving segmentation data

	α, β	$10^{-4}, 10^{-7}$ (200 epochs)	$10^{-4}, 10^{-6}$ (160 epochs)	$10^{-5}, 10^{-7}$ (140 epochs)
Train (98 scans)	MSE	0.291	0.353	0.239
	CC	0.600	0.532	0.651
	Dice	0.951	0.944	0.954
	Haussdorf	71.6	64.0	65.0
Test (33 scans)	MSE	0.300	0.327	0.213
	CC	0.587	0.595	0.686
	Dice	0.950	0.955	0.958
	Haussdorf	70.5	62.4	61.0

Table 3. Atlas registration average scores with segmentation

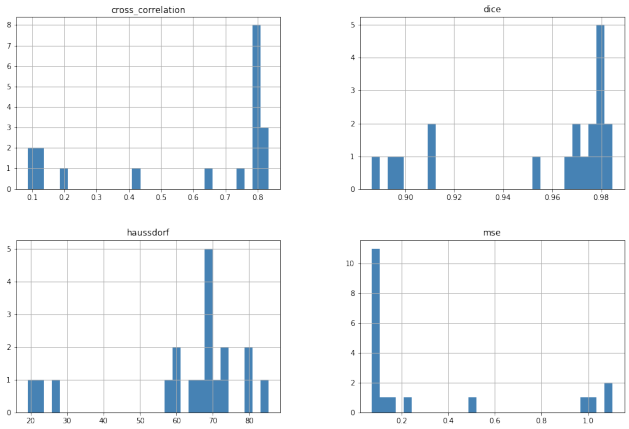

Figure 12. Scores distribution on testing set for $(\alpha, \beta) = (10^{-4}, 10^{-6})$

triggered an neat improvement in registration quality, bolstering the usage of the network's power. It seems fair to believe that orienting the optimizer in the right direction by constraining it on anatomical regions overlapping did benefit to our task.

$(\alpha, \beta) = (10^{-5}, 10^{-6})$ seems to be a good compromise, yet yielding best scores for the dense and segmentation based metrics. An extensive grid search would allow a better fine tuning of these weights. However, we can still observe on Figure 13 a couple of artefacts.

6. Conclusion and future work

In this work, we demonstrate on a simple example that recent unsupervised 3D CNNs registrations methods can effectively be ported to CT scan registration. The modularity of these frameworks with regard to the chosen architecture, nature of the transformation, choice of similarity measure or regularization offers a wide range of possible approaches to be explored.

We also point out how leveraging anatomical regions segmentation data benefits to the network training, significantly helping it converge in a direction enhancing its reg-

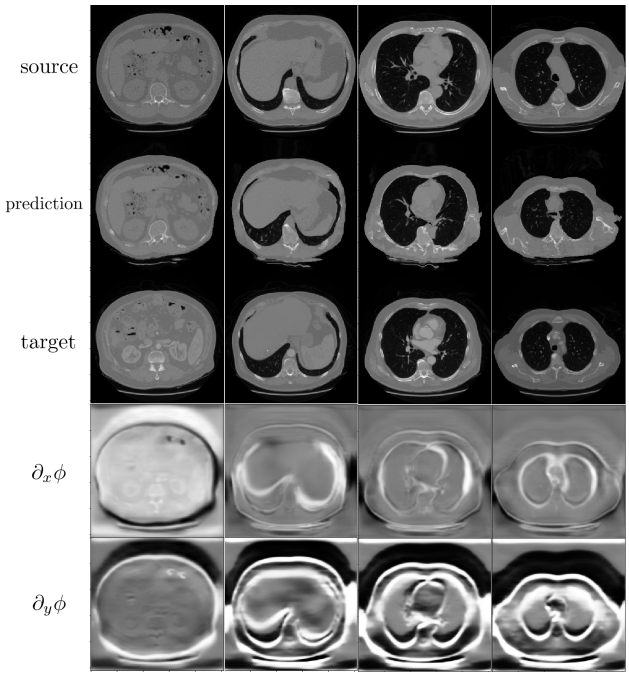

Figure 13. Network's output when trained with segmentation $(\alpha, \beta) = (10^{-5}, 10^{-t})$

istering performances.

We leave for future work to try adding the discarded affine registration decoding path to what extent dissociating global rigid alignment from deformable registration serves positively registration despite of complexity. Also, one of the main challenges at stake when it comes to CT scans resides in their memory greediness. Trying to reimplement these networks more efficiently would definitely benefit to this work, allowing to test more complex networks without being in spite of the scans quality

References

[1] B. Avants, C. Epstein, M. Grossman, and J. Gee. Symmetric diffeomorphic image registration with cross-correlation: Evaluating automated labeling of elderly and neurodegenerative brain. *Medical image analysis*, 12:26–41, 03 2008. 2

[2] R. Bajcsy and S. Kovai. Multiresolution elastic matching. *Computer Vision, Graphics, and Image Processing*, 46(1):1 – 21, 1989. 2

[3] G. Balakrishnan, A. Zhao, M. R. Sabuncu, J. V. Guttag, and A. V. Dalca. Voxelmorph: A learning framework for deformable medical image registration. *CoRR*, abs/1809.05231, 2018. 2, 4

[4] X. Cao, J. Yang, L. Wang, Z. Xue, Q. Wang, and D. Shen. Deep learning based inter-modality image registration supervised by intra-modality similarity. *CoRR*, abs/1804.10735, 2018. 2, 7

[5] F. Chollet et al. Keras. https://github.com/fchollet/keras, 2015. 5

[6] S. Christodoulidis, M. Sahasrabudhe, M. Vakalopoulou, G. Chassagnon, M.-P. Revel, S. Mougiakakou, and N. Paragios. Linear and Deformable Image Registration with 3D Convolutional Neural Networks. In *Reconstruction and Analysis of Moving Body Organs, 21th International Conference on Medical Image Computing and Computer Assisted Intervention 2018*, Grenada, Spain, Sept. 2018. 2, 4, 5

[7] A. V. Dalca, G. Balakrishnan, J. V. Guttag, and M. R. Sabuncu. Unsupervised learning for fast probabilistic diffeomorphic registration. *CoRR*, abs/1805.04605, 2018. 2

[8] M. Simonovsky, B. Gutiérrez-Becker, D. Mateus, N. Navab, and N. Komodakis. A Deep Metric for Multimodal Registration. In S. Ourselin, L. Joskowicz, M. R. Sabuncu, G. Unal, and W. Wells, editors, *19th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2016)*, volume 9902 of *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, pages 10–18, Athènes, Greece, Oct. 2016. Springer. 1

[9] P. . W. I. Viola. 1997. 2