VoxelMorph++: a convolutional neural network architecture for unsupervised CBCT to CT deformable image registration

Dingqian Liu, and *Jiwei* Liu*

School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing; 100083, China

Abstract. We use an unsupervised method based on the VoxelMorph architecture for Cone-beam computed tomography (CBCT) to CT deformable image registration (DIR), and propose VoxelMorph++, a new architecture for predicting the deformation vector field (DVF). The proposed architecture (1) overcomes the limitation that the optimal depth of encoder-decoder is unknown, by forming a nested structure where each feature with varying depth in the encoder path has a corresponding depth decoder; (2) fuses features of varying semantic scales more flexibly by redesigning skip connections. In the testing phase, we used ITK-SNAP software to semi-automatically segment the patients' lung regions as labels to solve the problem of expensive manual labelling. We evaluated these two architectures using lung region registration results from 10 patients' CBCT and CT images. After registration, the mean Dice score improved from 0.8556 to 0.9412 and 0.9430 for VoxelMorph and the proposed architecture, respectively. The results show that both architectures perform well in our dataset and the proposed architecture outperforms VoxelMorph in terms of registration accuracy.

Keywords: CBCT, CT, Unsupervised learning, Deformable image registration, Deep learning.

1 Introduction

CBCT and CT deformable image registration (DIR) is one of the key technologies for image-guided radiotherapy (IGRT) [1-3]. However, there are two challenges in practical applications. First, it's hard to obtain the ground truth transformation. Second, there are large artifacts in CBCT images. To overcome these two challenges, we chose an unsupervised MRI brain DIR method, which was proposed by Balakrishnan et al. [4], to achieve CBCT to CT DIR. This method predicts the deformation vector field (DVF) through a CNN architecture named 'VoxelMorph' and generates warped images with the help of the spatial transformer network (STN) [5]. The overall registration framework is shown in figure 1. The artifacts in CBCT can be regarded as a kind of noise, and the encoder-decoder architectures in

* Corresponding author: liujiwei@ustb.edu.cn

VoxelMorph are robust to noise. In addition, this method selects local cross-correlation (CC) as the main part of the loss function. The robustness of local CC to intensity variations [6] can also reduce the interference of artifacts.

However, for CBCT to CT DIR, the VoxelMorph architecture has two limitations. On the one hand, the depth of the current network architecture is not necessarily optimal. On the other hand, the U-Net [7] like structure of VoxelMorph makes it only aggregate the up-sampled output from the preceding node in the decoder path and the feature map with the same resolution in the encoder path. This restrictive pairwise fusion scheme may also not be optimal. Therefore, referring to the UNet++ [8], we propose a new architecture, VoxelMorph++, as shown in figure 2. The proposed architecture consists of VoxelMorph architectures of varying depths which share the same encoder so that it is convenient for the network to learn and adjust the weights of varying depth features in the encoder path according to different tasks. By redesigning architecture's long and short skip connections, the proposed architecture can integrate features of varying semantic scales more flexibly, while the gradient can be back-propagated from the deeper decoder to the shallower decoder during the training phase.

In the testing phase, to solve the problem of expensive time and labor cost of manually making labels, we use the 3D medical image segmentation function of ITK-SNAP [9] software to semi-automatically generate lung region's masks for all images in the test set, as the label used to calculate the Dice score. We evaluate VoxelMorph and VoxelMorph++ on our own dataset. The experimental results show that both architectures have high registration performance even under the interference of artifacts, and the performance of our proposed architecture is slightly better than the VoxelMorph.

2 Method

2.1 Overview of the registration framework

Similar to the framework in paper [4], our unsupervised CBCT to CT deformable image registration framework is shown in figure 1. We take the input CBCT image as the moving image M, and the CT image as the fixed image F, each voxel p of M and F is defined in the 3D spatial domain $\Omega \subset \mathbb{R}^3$.

By concatenating M and F into a 2-channel 3D image and sending it to the registration network which is a convolutional neural network (CNN) used to model the function $g_\theta(F, M) = \phi$. θ are learnable parameters of g, and ϕ is a registration field output by the network, which is used to deform the moving image M through the spatial transformation function, and obtain the warped image $M(\phi)$. In the training phase, the network finds the optimal parameter $\hat{\theta}$ by minimizing the loss function L.

All figures that depict trunks in this paper show 2D transverse slices for visualization purposes only. All registration is done in 3D.

2.2 Proposed registration network architecture: VoxelMorph++

Based on the VoxelMorph-2 architecture which focuses more on registration accuracy among the two variants of VoxelMorph, and inspired by UNet++, we propose a new registration network architecture VoxelMorph++, as shown in figure 2. In this architecture, two separate convolutional layers (followed by Leaky ReLU activations) in the decoder stage are removed to save GPU memory. The parameter settings of each layer in the proposed network are the

same as VoxelMorph-2. All convolutional layers are 3D convolutions with a kernel size of $3\times3\times3$, and each LeakyReLU activation layer's negative slope parameter is 0.2.

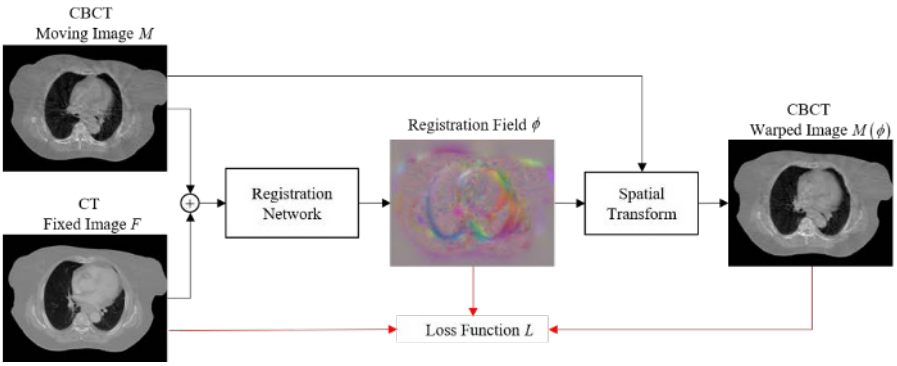

Fig. 1. Overview of the registration framework for CBCT to CT deformable image registration.

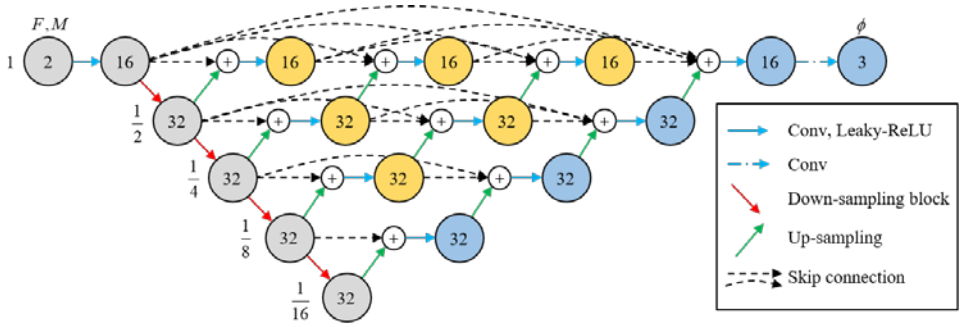

Fig. 2. Proposed architecture VoxelMorph++. Each circle represents a 3D volume. The number of channels is shown inside the circle. All 3D volumes in the same horizontal direction have the same spatial resolution, which is printed at the beginning of each row with respect to the input volume.

The input of network is a 2-channel 3D image concatenated by M and F. In our experiments, the input is concatenated by a pair of patches with the same corresponding positions in M and F. Each patch has a size of $160\times128\times z\left(z=64,128,160\right)$, where z is the z-axis of M and F. The size of the multi-channel 3D image is $160\times128\times z\times2$. The multi-channel input first goes through a convolutional layer, followed by Leaky ReLU activation, and then goes through a down-sampling block which consists of a $2\times2\times2$ max pooling layer with a stride of 2 and a convolutional layer followed by Leaky ReLU activation. In the down-sampling block, the size of the input is first reduced by half through the max pooling layer, and then the convolutional layer followed by Leaky ReLU activation is used to extract deeper features. We successively apply the down-sampling block for 4 times in the encoder stage to reduce the input size to $\frac{1}{16}$ of the original size and extract features of varying depths.

Unlike VoxelMorph-2, which only applies upsampling to the deepest features in the encoder, we add decoders with the corresponding depth to each feature of varying depths extracted in the encoder, and restore these features to the original image size by upsampling. In this way, the proposed network architecture embeds VoxelMorph-2 architectures of varying depths which share the same encoder. We alternate between upsampling, concatenating skip connections and convolutions (followed by Leaky ReLU activations) for each decoder. And we use redesigning skip connections to concatenate the feature maps of

the encoder and different decoders at the same resolution. This design not only enables the shallower decoders to update parameters via back-propagation, but also provides additional features of varying semantic scales to the aggregation layers. The output of the final aggregation layer goes through an extra convolutional layer without Leaky ReLU activation to obtain ϕ, which in our experiments is of size $160\times128\times z\times3$. In the proposed architecture, we do not use deep supervision [10] which is not required.

2.3 Spatial transformation function

For the ϕ output by the registration network, we use a spatial transformation function based on the STN to obtain $M(\phi)$. For each voxel p in $M(\phi)$, we obtain it by applying trilinear interpolation to a (subpixel) voxel location $\phi(p)$ in M. That is, we perform:

$$M(\phi(p))=\sum_{q\in Z(\phi(p))}M(q)\prod_{d\in\{x,y,z\}}\left[1-\left|\phi_d(p)-q_d\right|\right], \tag{1}$$

Where $Z(\phi(p))$ is an 8-voxel cubic neighborhood of $\phi(p)$. Because the operations are differentiable almost everywhere, we can use back-propagation to train the registration network.

2.4 Loss function

Same as the method in paper [4], the loss function $\mathcal{L}$ of our method consists of two parts: L_{sim} which penalizes appearance differences, and L_{smooth} which penalizes local spatial variation in ϕ. The total equation of L is:

$$L(F,M,\phi)=L_{sim}(F,M(\phi))+\lambda L_{smooth}(\phi), \tag{2}$$

Where λ is the regularization parameter.

$$n=9$$

L_{sim} is defined as the negative local cross-correlation (CC) of $M(\phi)$ and F:

$$L_{sim}(F,M(\phi))=-CC(F,M(\phi))=-\sum_{p\in\Omega}\frac{\left\{\sum_{p_i}\left[F(p_i)-\widehat{F}(p)\right]\left[M(\phi(p_i))-\widehat{M}(\phi(p))\right]\right\}^2}{\left\{\sum_{p_i}\left[F(p_i)-\widehat{F}(p)\right]^2\right\}\left\{\sum_{p_i}\left[M(\phi(p_i))-\widehat{M}(\phi(p))\right]^2\right\}}, \tag{3}$$

Where $\widehat{F}(p)$ and $\widehat{M}(\phi(p))$ denote the local mean intensities of a n^3 volume centered on voxel p in images, p_i represents each voxel in the n^3 volume, and $n=9$ in our experiments. The robustness of local CC to intensity variations is beneficial to reduce the interference of artifacts during training. And a lower L_{sim} indicates a better alignment.

Smooth ϕ is encouraged by adding a regularizer L_{smooth} to the loss function L:

$$L_{smooth}(\phi) = \sum_{p\in\Omega} \left\| \nabla \phi(p) \right\|^2 . \tag{4}$$

The smoothness of ϕ is enforced with L_{smooth}. Finally, we write the complete loss as:

$$L(F, M, \phi) = -CC(F, M(\phi)) + \lambda \sum_{p\in\Omega} \left\| \nabla \phi(p) \right\|^2 . \tag{5}$$

3 Experiments

3.1 Data acquisition and preprocessing

The entire dataset was randomly selected from the clinical database, including whole-body planning CT (pCT) and CBCT scans from 34 patients scanned on different days. The parameters of pCT scans were 120 kVp, an in-plane resolution of range from 1.01 mm to 1.33 mm, and a slice thickness of 3.00 mm. The corresponding parameters of CBCT scans were 110 kVp, 0.51 mm to 0.91 mm, and 1.98 mm to 2.00 mm, respectively. Among those patients, 24 were used in the training set and 10 in the testing set. For training and testing, we resampled all CBCT and pCT scans to a $1.0\text{mm} \times 1.0\text{mm} \times 1.0\text{mm}$ grid to achieve consistent spatial resolution, and clipped their HU values to [-1000, 2000]. Then, we rigidly aligned all pCT scans and CBCT scans using AIRLab [11]. For each patient, we extracted a pair of CBCT and pCT slices with corresponding lung positions from scans. That is, there are 34 CBCT-CT image pairs in the total dataset. Finally, All CBCT and pCT slices were cropped or padded to dimensions $416 \times 320 \times 160$, except for two patients in the test set, which were $416 \times 320 \times 64$ and $416 \times 320 \times 128$, respectively.

3.2 Labels making based on ITK-SNAP software and dice score

Because the data used in this paper are clinical data, we evaluate the registration accuracy of network architectures on lung regions based on the Dice score, rather than landmarks. To achieve this, labels of the lung regions of all CBCT-CT image pairs in the test set are required, but it is arduous and time-consuming to make them manually. In order to overcome this challenge, we use the 3D medical image segmentation function of the ITK-SNAP [9] software to semi-automatically segment the lung regions of these images to make labels. The labels made in this way are shown in figure 3. Let M_l and F_l be the labels of the lung regions of M and F, respectively. Through the spatial transformation function, the warped label $M_l(\phi)$ is obtained by applying the same ϕ as M to M_l. We evaluate the registration performance of architectures by computing the Dice score between $M_l(\phi)$ and F_l:

$$Dice(M_l(\phi), F_l) = 2 * \frac{\left| M_l(\phi) \cap F_l \right|}{\left| M_l(\phi) \right| + \left| F_l \right|} . \tag{6}$$

In our experiments, the Dice score quantifies the degree of the lung regions' spatial overlap between $M(\phi)$ and F, with a value between 0 (no overlap) and 1 (complete overlap).

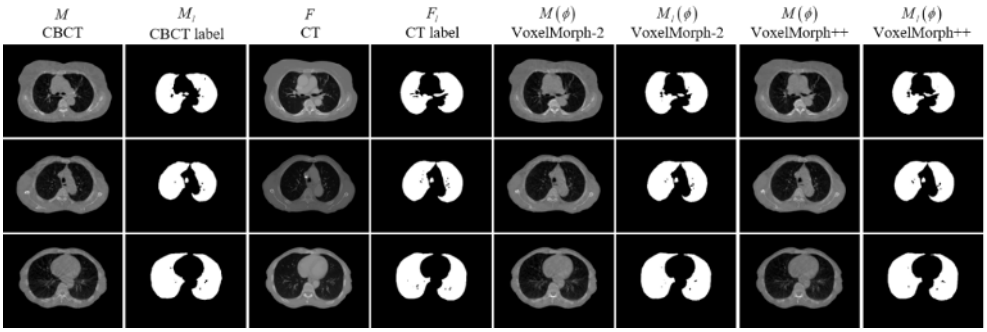

Fig. 3. Example CBCT-CT transverse slices extracted from input pairs (columns 1, 3), corresponding lung region labels made by ITK-SNAP (columns 2, 4), and resulting $M(\phi)$ and $M_l(\phi)$ for VoxelMorph-2 and VoxelMorph++ (the proposed architecture). Each line represents a different patient.

3.3 Implementation

We implemented the proposed architecture under the PyTorch deep learning framework. To demonstrate the effectiveness of our proposed improved architecture VoxelMorph++ compared to VoxelMorph-2, the same parameter settings were used for both architectures during training and testing. We used the Adam optimizer [12] with a 1×10^{-4} learning rate. The value of the hyperparameter λ in equation (2) was set to 1.5. Both architectures were trained and tested on an NVIDIA RTX 2080 Ti GPU with 11 GB of memory. The HU values of all input images were normalized to [0, 1]. The batch size was set to 1 and the training epoch was set to 60.

Due to hardware limitations, we implemented patch-based registration. We extracted overlapping patches in the x and y directions of the input images by the step size of 64. The size of each patch is $160\times128\times z$, where z is 160 for all images except two CBCT-CT image pairs in the test set, which have z of 64 and 128, respectively. And the size of the overlap between any two adjacent patches is $96\times64\times z$. This overlap ensures that continuous whole-image output $M(\phi)$ can be obtained. For the overlaps between output patches, only the voxel values in the volumes of $64\times48\times z$ were averaged, where the volumes were selected from the middle of these overlaps.

4 Results

Figure 3 shows the visual registration results of VoxelMorph-2 and the proposed architecture Voxel-Morph++ in our dataset. As seen, the registration effect of the lung regions between M and F is significantly improved after applying both architectures. We further quantify the registration results through Dice scores.

For the 10 CBCT-CT image pairs corresponding to 10 patients in the test set, the results of the single Dice scores and the average Dice scores before and after registration are shown in table 1. The proposed architecture achieves higher registration accuracy than VoxelMorph-2. And both architectures improve significantly on rigid alignment. In terms of single Dice scores, the proposed architecture outperforms VoxelMorph-2 in 7 of the 10 CBCT-CT image pairs, and is also comparable to VoxelMorph-2 in the remaining 3 image pairs. Moreover, the proposed architecture VoxelMorph++ performs slightly better than VoxelMorph-2 in terms of average Dice scores.

Table 1. Single and average Dice scores and runtime results for rigid alignment, VoxelMorph-2 and VoxelMorph++ (the proposed architecture).

Method	Single Dice scores for 10 CBCT-CT pairs in test set										Mean
	1	2	3	4	5	6	7	8	9	10	
Rigid alignment	0.7473	0.8007	0.8989	0.8577	0.8981	0.8675	0.7069	0.8864	0.9280	0.9644	0.8556± 0.0766
VoxelMorph-2	**0.8677**	0.9256	0.9702	0.9491	0.9652	0.9529	**0.8669**	0.9602	**0.9794**	0.9748	0.9412± 0.0397
VoxelMorph++	0.8661	**0.9366**	**0.9715**	**0.9525**	**0.9666**	**0.9565**	0.8617	**0.9644**	0.9785	**0.9753**	**0.9430**± 0.0412

5 Conclusion

In this study, we used an unsupervised method based on the VoxelMorph-2 architecture for CBCT to CT deformable image registration, aiming to overcome two challenges: ground truth acquisition and large artifacts. Inspired by UNet++, we proposed an improved architecture, named VoxelMorph++, for more accurate image registration. Through the nested structure and redesigned skip connections, the proposed architecture addressed two key challenges of VoxelMorph-2: the unknown optimal depth of the architecture and the unnecessary pairwise feature fusion strategy. In the testing phase, we used the ITK-SNAP software to semi-automatically segment the lung regions of the input images as labels, which avoided the expensive cost of manpower and time caused by making labels manually. The experimental results demonstrated that both architectures showed good registration performance in our dataset and the proposed architecture was slightly better than VoxelMorph-2 in terms of registration accuracy.

References

1. Park S and Plishker W 2017 Deformable registration of CT and cone-beam CT with local intensity matching (Physics in Medicine & Biology vol 62) pp 927-947
2. Zachiu C and De Senneville B D 2017 Non-rigid CT/CBCT to CBCT registration for online external beam radiotherapy guidance (Physics in Medicine & Biology vol 63) 015027
3. Cole A J and Veiga C 2018. Toward adaptive radiotherapy for lung patients: feasibility study on deforming planning CT to CBCT to assess the impact of anatomical changes on dosimetry (Physics in Medicine & Biology vol 63) 155014
4. Balakrishnan G and Zhao A 2018 An unsupervised learning model for deformable medical image registration (Proceedings of the IEEE conference on computer vision and pattern recognition) pp 9252-9260
5. Jaderberg M and Simonyan K 2015 Spatial transformer networks (Advances in neural information processing systems) pp 2017–2025
6. Avants B B and Epstein C L 2008 Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain (Medical image analysis vol 12) pp 26-41

7. Ronneberger O and Fischer P 2015 U-net: Convolutional networks for biomedical image segmentation (International Conference on Medical image computing and computer-assisted intervention) pp 234-241
8. Zhou Z and Siddiquee M M R 2019 Unet++: Redesigning skip connections to exploit multiscale features in image segmentation (IEEE transactions on medical imaging vol 39) pp 1856-1867
9. Yushkevich P A and Piven J 2006 User-guided 3D active contour segmentation of anatomical structures: significantly improved efficiency and reliability (Neuroimage vol 31) pp 1116-1128
10. Lee C Y and Xie S 2015 Deeply-supervised nets (Artificial intelligence and statistics) pp 562-570
11. Sandkühler R and Jud C 2018 AirLab: autograd image registration laboratory (arXiv preprint arXiv:1806.09907)
12. Kingma D P and Ba J 2014 Adam: A method for stochastic optimization (arXiv preprint arXiv:1412.6980)