Affine Medical Image Registration with Coarse-to-Fine Vision Transformer (SUPPLEMENTARY MATERIAL)

A. Unsupervised and Semi-Supervised Learning

Figure 1 depicts the proposed unsupervised and semi-supervised training scheme of the Coarse-to-Fine Vision Transformer (C2FViT). The segmentation maps are only required in the training phrase under the semi-supervised training scheme.

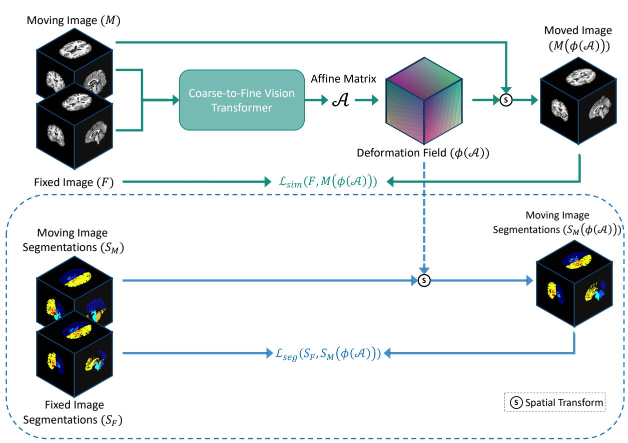

Figure 1. Schematic representation of the unsupervised and semi-supervised learning scheme in the Coarse-to-Fine Vision Transformer. The unsupervised and semi-supervised learning schemes are highlighted in green and blue colours, respectively.

B. Affine Transformations

The corresponding translation $\mathcal{T}$, rotation $\mathcal{R}$, scaling $\mathcal{S}$ and shearing $\mathcal{H}$ transformations derived by the geometric transformation parameters $t_x, t_y, t_z \in \boldsymbol{t}$, $r_x, r_y, r_z \in \boldsymbol{r}$, $s_x, s_y, s_z \in \boldsymbol{s}$ and $h_x, h_y, h_z \in \boldsymbol{h}$ are defined as follows:

$$\mathcal{T} = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}, \mathcal{R}_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(r_x) & \sin(r_x) & 0 \\ 0 & -\sin(r_x) & \cos(r_x) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathcal{S} = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \mathcal{R}_y = \begin{pmatrix} \cos(r_y) & 0 & -\sin(r_y) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(r_y) & 0 & \cos(r_y) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathcal{H} = \begin{pmatrix} 1 & h_{xy} & h_{xz} & 0 \\ 0 & 1 & h_{yz} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \mathcal{R}_z = \begin{pmatrix} \cos(r_z) & -\sin(r_z) & 0 & 0 \\ \sin(r_z) & \cos(r_z) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where rotation matrix $\mathcal{R}$ equals to $\mathcal{R}_x\mathcal{R}_y\mathcal{R}_z$.

C. Additional Implementation Details

Table 1 summarizes the configurations of C2FViT at each stage. Specifically, the input resolution, stride in the convolutional patch embedding, number of transformer encoders, embedding size of each patch embedding, embedding size of the convolutional feed-forward layer and number of heads for the multi-head self-attention module are listed in the table.

Stage	Input size	Stride	# Encoders	Hidden size	MLP size	Heads
Stage 1	32^3	2^3	4	256	512	2
Stage 2	64^3	4^3	4	256	512	2
Stage 3	128^3	8^3	4	256	512	2

Table 1. Model configurations of Coarse-to-Fine Vision Transformer at each stage.

D. Additional Qualitative Results

Figure 2 shows example MR slices obtained from the MNI152 template, OASIS and LPBA datasets. As shown in the figure, there are significant spatial and structural differences across scans as all scans are in native space, except for the MNI152 template. The comprehensive qualitative results of template-matching normalization and atlas-based registration tasks with the OASIS and LPBA dataset of the learning-based methods *without spatial initialization* are shown in figure 3.

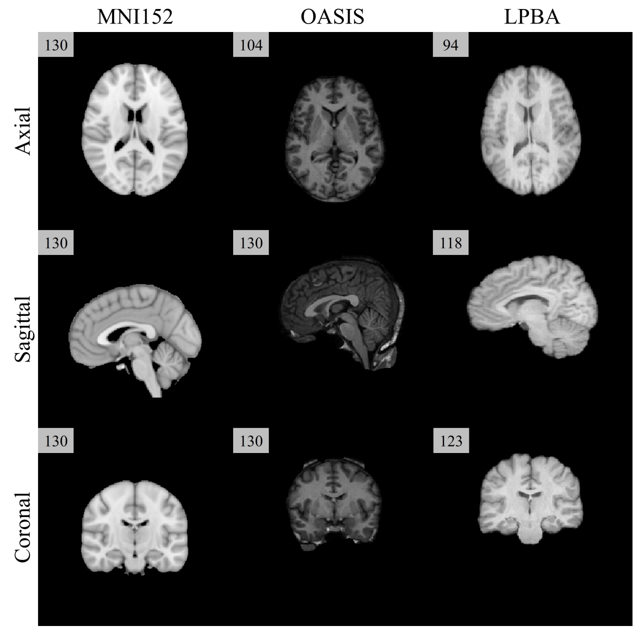

Figure 2. Example axial, sagittal and coronal slices obtained from the MNI152 template, OASIS and LPBA brain MRI datasets. The corresponding slice number of each slice is highlighted at the top-left corner.

E. Details of ANTs and Elastix

The command and parameters we used for ANTs:

```
-d 3 -v 1 -t Affine[0.1]
-m MI[<Fixed>,<Moving>,1,32,Regular,0.1]
-c 200x200x200 -f 4x2x1 -s 2x1x0
-o <OutFileSpec>
```

The command and parameters we used for Elastix:

```
ef = sitk.ElastixImageFilter()
ef.SetFixedImage(sitk.ReadImage(<Fixed>))
ef.SetMovingImage(sitk.ReadImage(<Moving>))
pmap = sitk.GetDefaultParameterMap("affine")
ef.SetParameterMap(pmap)
ef.Execute()
```

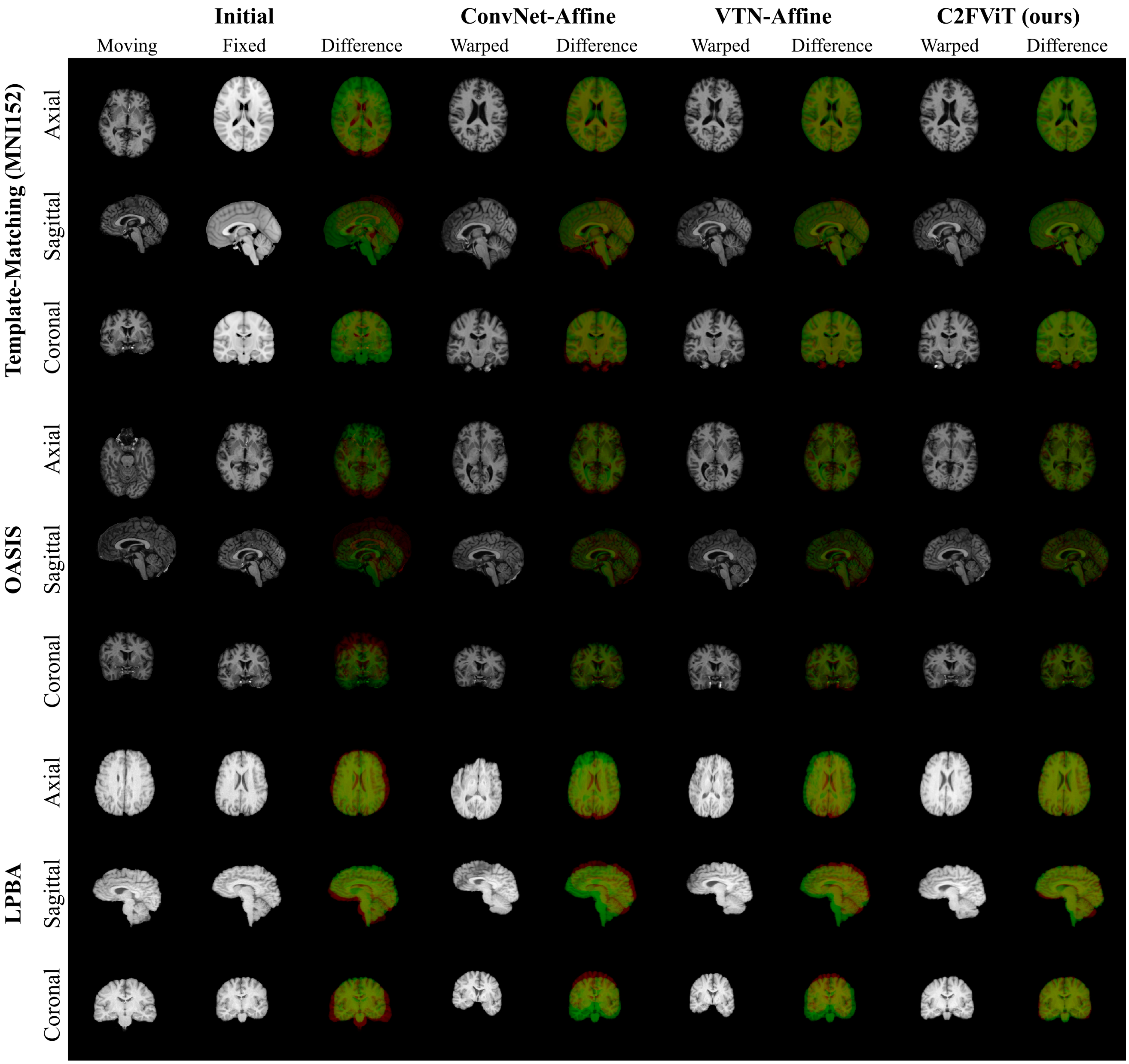

Figure 3. Example axial, sagittal and coronal MR slices obtained from the moving images, atlases (fixed images), resulting warped images for ConvNet-Affine, VTN-Affie and our method without center of mass initialization. For better visualization, we depict a difference map for each method, in which the colour maps of fixed and warped moving images are set to black-green and black-red, respectively, and overlay the resulting warped moving image to fixed image.