Supervised learning in medical image registration

Sokooti, H.

3

3D Convolutional Neural Networks Image Registration Based on Efficient Supervised Learning from Artificial Deformations

This chapter was adapted from:

H Sokooti, B de Vos, F Berendsen, M Ghafoorian, S Yousefi, BP Lelieveldt, I Išgum, and M Staring. **3D Convolutional Neural Networks Image Registration Based on Efficient Supervised Learning from Artificial Deformations**, *arXiv preprint arXiv:1908.10235*, 2019.

Abstract

We propose a supervised nonrigid image registration method, trained using artificial displacement vector fields (DVF), for which we propose and compare three network architectures. The artificial DVFs allow training in a fully supervised and voxel-wise dense manner, but without the cost usually associated with the creation of densely labeled data. We propose a scheme to artificially generate DVFs, and for chest CT registration augment these with simulated respiratory motion. The proposed architectures are embedded in a multi-stage approach, to increase the capture range of the proposed networks in order to more accurately predict larger displacements. The proposed method, RegNet, is evaluated on multiple chest CT scans studies and achieved a target registration error of 2.32 ± 5.33 mm and 1.86 ± 2.12 mm on SPREAD and DIR-Lab-4DCT studies, respectively. The average inference time of RegNet with two stages is about 2.2 s.

3.1 Introduction

Image registration is the process of aligning images and has many applications in medical image analysis. Generally, image registration casts to an optimization problem of minimizing a predefined handcrafted intensity-based dissimilarity metric over a transformation model. Both the dissimilarity metric and the transformation model need to be selected and tuned in order to achieve high quality registration performance. This task is time-consuming and there is no guarantee that the selected dissimilarity model fits with new images.

General learning-based techniques have been used in several registration papers. Guetter et al. [53] incorporated a prior learned joint intensity distribution to perform a nonrigid registration. Jiang et al. [54] selected and fused a large number of features instead of using only one similarity metric. Hu et al. [41] leveraged regression forests to predict an initial DVF. In terms of predicting registration accuracy Muenzing et al. [39] casted this task to a classification problem and extracted several local intensity-based features, which are fed to a two-stage classifier. Sokooti et al. [55, 33] extracted some intensity-based and registration-based features, then by using regression forests estimated the local registration error.

In recent years, CNNs have also been utilized in the context of image registration. Miao et al. [42] used CNNs for rigid-body transformations. Yang et al. [43] trained a CNN to predict the initial momentum of a 3D LDDMM registration. Cao et al. [56] generated a multi-scale similarity map and utilized it to predict the DVF. Simonovsky et al. [57] proposed a CNN-based similarity metric for multi-modal registration. Their training samples were a set of aligned images as the positive cases and a set of manually deformed images as the negative cases.

In the unsupervised deep learning approaches, de Vos et al [22, 23] for the first time used normalized cross correlation (NCC) of the fixed and moving image as a loss function. Later Balakrishnan et al. [24] and Ferrante et al. [58] used the same loss to train their network. Mahapatra et al. [25] combined NCC with other similarity metrics such as the Dice overlap metric over the labeled images. Elmahdy et al. [8] utilized an adversarial training based on the segmentation maps in addition to the NCC loss. Sheikhjafari et al. [59] and Dalca et al. [60] employed the mean squared intensity difference, which was applied to mono-modal image registration. Hu et al. [61] proposed a loss function that calculates cross entropy over the smoothed segmentation maps, which was applied to multi-modal images. A drawback to use conventional similarity metrics is that these similarity metrics are not perfect and might not fit in all images.

In the supervised approaches, for the first time Sokooti et al. [17] generated artificial DVFs with different frequencies to train a CNN architecture. Rohé et al. [62]

proposed to build reference DVFs which were obtained by performing registration over segmented regions of interest. Fan et al. [63] proposed a ground truth based on the GAN network. The implicit ground truth is assigned using the negative cases derived from the generator network while the positive cases are synthetically made by perturbing the original images. Eppenhof et al. [19] constructed a small set of images by applying a random DVF. In the model-based methods Uzunova et al. [64] proposed statistical appearance models to be used for data augmentation. Hu et al. [65] utilized biomechanical simulations to regularize their network.

In several articles, reinforcement learning is used [66, 67, 68]. An artificial agent is trained by making a statistical deformation model from training data. However, this approach is still iterative and might be slow at inference time.

Conventionally, in quality assessment of registration, manually selected landmarks or manually segmented regions are used. However, utilizing them as a gold standard in training has some drawbacks. With manually segmented regions, several measurements like Dice and mean surface distance can be calculated, but there is no direct correlation between Dice and the true DVF in all voxels of the image. The drawback of using landmarks as a gold standard [55, 33] is that the numbers of landmarks usually is not enough to estimate a continuous gold standard DVF for the whole image.

In this paper, instead of using a transformation model, we directly predict the displacement vector field (DVF). The convolutional neural network (CNN) implicitly learns the dissimilarity metric. The current paper is a large extension of the work first presented in Sokooti et al. [17]. We present more ways to construct sufficiently realistic synthetic DVFs. The network design is greatly enhanced by increasing the capture range in order to more accurately predict larger DVFs. A multi-stage approach is also proposed to overcome this issue. The evaluation is performed on the SPREAD study as well as on the public DIR-Lab study. The proposed method is capable to be trained on any datasets without needing any manual ground truth.

3.2 Methods

3.2.1 System overview

A block diagram of the proposed system is given in Fig. 3.1. The inputs of the system are a fixed image I_F and a moving image I_M. Similar to the conventional registration methods, a multi-stage approach is employed. The registration blocks RegNet^4 and RegNet^2 perform on the down-scaled images with a factor of 4 and 2, respectively. The inputs of the final registration block RegNet^1 are original resolution images. The output of the system is a predicted DVF of transforming the moving image to the fixed image which is defined as $\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{T_{s1}}\big(\boldsymbol{T_{s2}}\big(\boldsymbol{T_{s4}}(\boldsymbol{x})\big)\big)$.

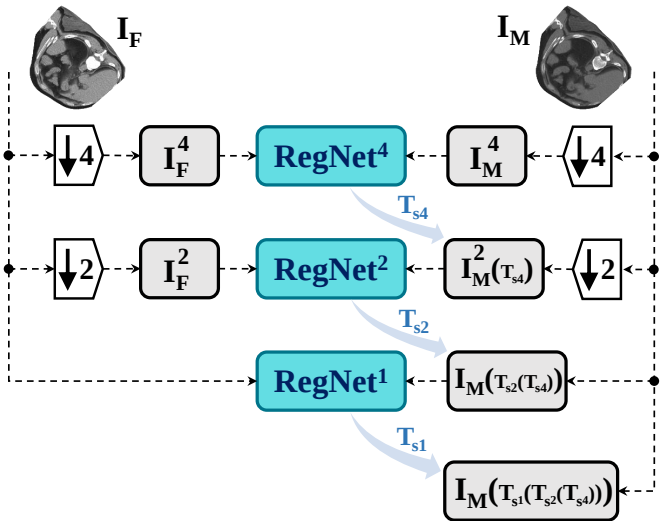


Figure 3.1: Block diagram of the proposed system. The initial inputs of the system are fixed and moving images down-scaled by a factor of four (↓4). Three RegNets process the input images over three stages (4, 2, 1) and generate the final output $\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{T}_{s1}\big(\boldsymbol{T}_{s2}\big(\boldsymbol{T}_{s4}(\boldsymbol{x})\big)\big)$.

3.2.2 Network architecture

We propose three network architectures for the RegNet design. The first two architectures are patch-based, and predict the DVF for a local neighborhood. These two networks are more complex and occupy a relatively large amount of GPU memory. The third architecture is based on a more simple U-Net design [69] with fewer network weights, and is capable of registering entire images (not patches), but down-scaled, within the memory limits of current GPUs. This last architecture is considered a candidate for the first resolution (RegNet^4), while the others are considered for the second and third resolution (RegNet^2 and RegNet^1). In Section 3.3 we compare these architectures and combinations thereof.

The networks have some settings in common. All convolutional layers use batch normalization [70] and ReLu activation [71], ,except for the last two layers of the U-Net design and the last three layers of the patch-based designs, where ELu activation is used to improve the regression accuracy. The last layer of all architectures does not use batch normalization nor an activation function. The Glorot uniform initializer [72] is used for all convolutional layers except for the trilinear upsampling, in which a fixed trilinear kernel is utilized. The three architectural designs are given in Fig. 2. The details are:

3.2.2.1 U-Net (U)

U-Net is one of the most common designs used in medial image segmentation The proposed modified design has an input size and output size of $125 \times 125 \times 125$ voxels. This architecture is only used for the sub-block RegNet^4, i.e. CNN-based registration is applied to down-scaled images with a factor of four. The proposed design is given in Fig. 3.2a. This relative simple design has 232,749 trainable parameters.

3.2.2.2 Multi-View (MV)

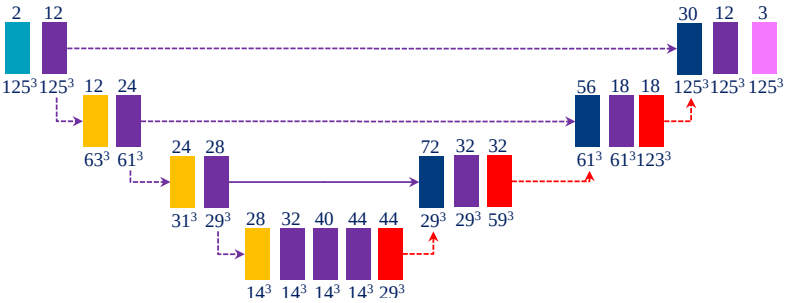
In this design, different scales are created by using conventional decimation by convolving the inputs with fixed B-spline kernels, which is similar to [17] and [46]. This design is relatively more memory efficient because of this multi-view approach. The proposed CNN architecture is visualized in Fig. 3.2b. The input of the network is a pair of 3D patches of size $105 \times 105 \times 105$ for the fixed and moving image. The network is then split into 3 pipelines: down-scaled with a factor of 4, a factor of 2, and the original resolution. In order to save memory, the original resolution and the down-scaled version with a factor 2 are cropped to $37 \times 37 \times 37$ and $67 \times 67 \times 67$, respectively. Decimation is done with the help of convolutions with a fixed B-spline kernel. In the down-scaled factor 2 pipeline, a stretched B-spline kernel with size $7 \times 7 \times 7$ is used. For down-scaling with a factor of 4, the B-spline kernel is stretched by a factor of 4, and has a size of $15 \times 15 \times 15$. Each pipeline continues with several convolutional layers with dilation of 1 or higher. The upsampling layers ensure spatial correspondence of all three pipelines. Finally, all pipelines are merged together followed by three more convolutional layers. The network gives three 3D outputs of size $21 \times 21 \times 21$ corresponding to the displacement in x, y and z direction. The total number of parameters in this design is 1,201,353.

3.2.2.3 U-Net-Advanced (Uadv)

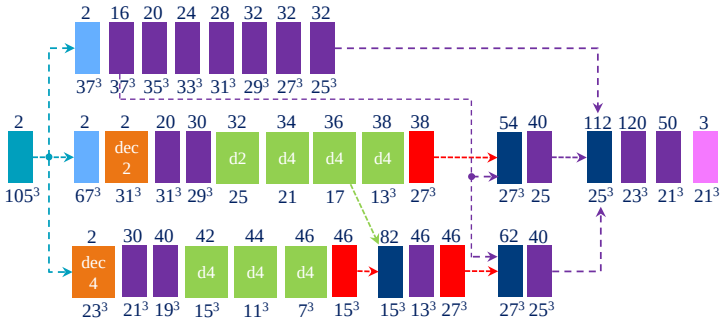
This proposed architecture is again a patch-based one but using a max-pooling technique instead of a decimation method. The global design is similar to the U-net architecture, but instead of simple shortcut connections, several convolutions are used for these connections. The proposed design is illustrated in Fig. 3.2c. The network starts with a convolutional layer to extract several low-level features from the images before any max-pooling. The size of the inputs and output are $101 \times 101 \times 101$ and $21 \times 21 \times 21$. The total number of parameters in this design is 1,420,701.

3.2.3 Artificial generation of DVFs and images

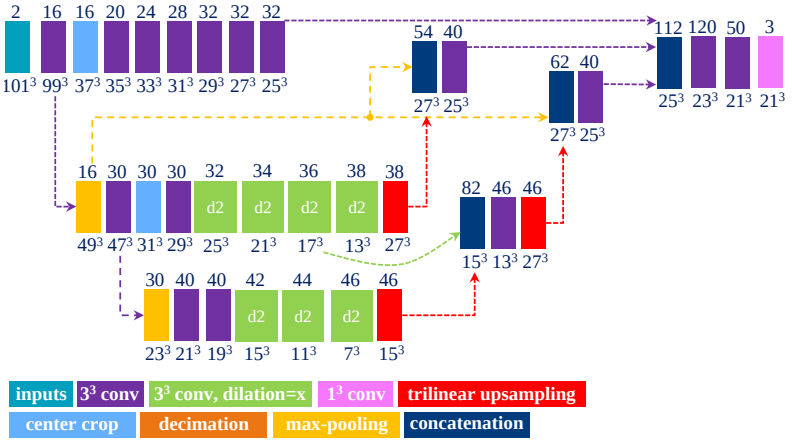
In order to train a CNN, a considerable number of ground truth DVFs are needed. We take a moving image I_M from the training set. The fixed image I_F is created artificially by generating a DVF, applying the DVF to the moving image resulting in I_F^{clean}, and adding artificial intensity models to finally obtain I_F.



(a) U-Net (U)



(b) Multi-view (MV)



(c) U-Net-advanced (Uadv)

Figure 3.2: RegNet designs: The inputs of the U-Net design are entire down-scaled images. However, in the Multi-view and U-Net-advanced architectures the output size is smaller than the input size and can be trained in a patch-based manner.

3.2.3.1 Artificial DVF

We propose to generate three categories of DVFs, to represent the range of displacements that can be seen in real images:

single frequency: The first category consists of DVFs having one or more local displacements of only one spatial frequency. They are generated as follows: Create an empty B-spline grid of control points with a spacing of s mm; Assign random values to the grid of control points and smooth it with a Gaussian kernel; Resample the B-spline grid to obtain the DVF; Normalize the DVF linearly to be in the range $[-\theta, +\theta]$ along each axis.

mixed frequency: In this category, two different spatial frequencies are mixed together as follows: Create a single frequency DVF similar to the previous category; Create a random binary mask and multiply it with the single frequency DVF. Finally, smooth the DVF with a Gaussian kernel with a standard deviation of σ_B. σ_B is chosen relatively small to generate a higher spatial frequency in comparison with the smooth filled region; By varying the σ_B value and s in the filled DVF, different spatial frequencies will be mixed together.

respiratory motion: We simulate respiratory motion with three components similar to [35] as follows: Expansion of the chest in the transversal plane with a maximum scaling factor of 1.12; Transition of the diaphragm in cranio-caudal direction with a maximum deformation of θ; Random deformation using the single frequency method. In order to locate the diaphragm, an automatically detected lung mask is used.

identity: This category comprises only identity DVFs. Later, when creating the artificially deformed image, intensity augmentations will be added to the deformed image. Thus, the network will be capable of detecting no motion, while the intensity values might have changed slightly.

3.2.3.2 Artificial intensity models

We propose two intensity models to be applied on the fixed images:

Sponge intensity model: By assuming mass preservation over the lung deformation, a dry sponge model [73] is added to deformed image:

$$\boldsymbol{I_F}(\boldsymbol{x}) = \boldsymbol{I_F^{\text{clean}}}(\boldsymbol{x})[\boldsymbol{J_T}(\boldsymbol{x})^{-1}], \tag{3.1}$$

where $\boldsymbol{J}$ denotes the determinant of the Jacobian of the transformation.

Gaussian noise: A Gaussian noise with a standard deviation of $\sigma_N = 5$ is added to the deformed image in order to achieve more accurate simulation of real images.

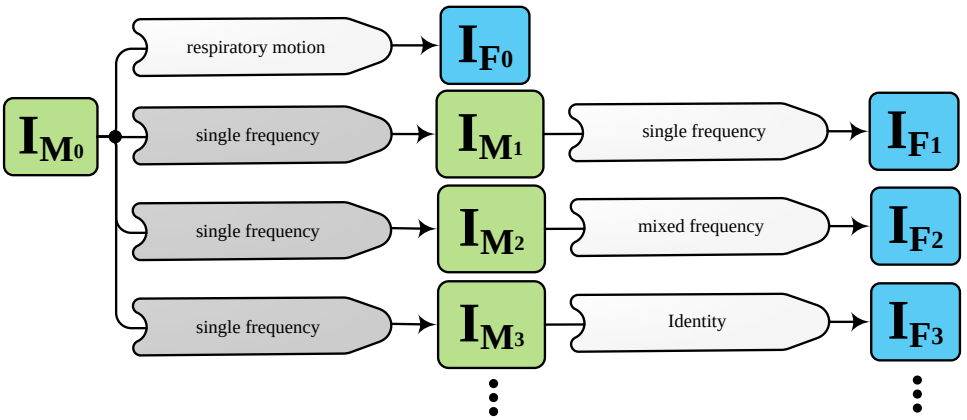
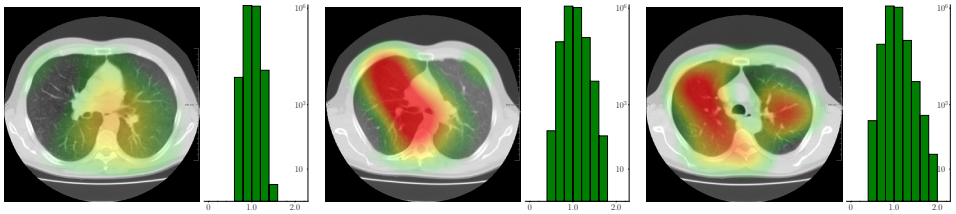


Figure 3.3: The generation of training pairs from a single input image I_{M0}. The input image is deformed slightly using the single frequency category, with the "lowest" settings (see Table 3.1), to generate moving images I_{Mi}. These are then each deformed and post-processed multiple times using all categories to generate fixed images I_{Fi}.

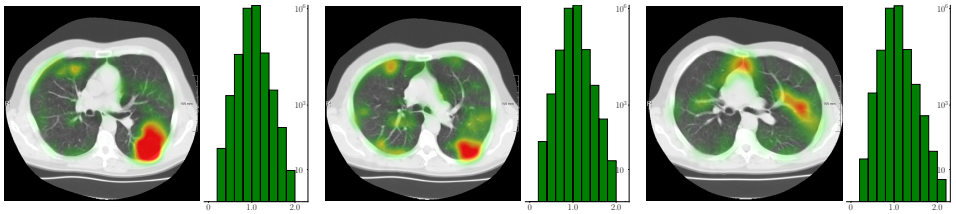
3.2.3.3 Extensive pair generation

For each single image in the training set, potentially a large number of artificial DVFs can be randomly generated. However, if this image is to be re-used for multiple DVFs, then for many training pairs we have the moving image unaltered. To tackle this problem, we also generate deformed versions of the original image (gray single frequency blocks in Fig. 3.3). A schematic design of utilizing artificial image pairs is depicted in Fig. 3.3. In this approach, the original image is only used once to generate the artificial image I_{F0}. Deformed versions of the original image I_{Mi} are used afterwards. Training pairs are thus $(I_{M0}, I_{F0}), (I_{M1}, I_{F1}), (I_{M2}, I_{F2}), ...$ The gray single frequency blocks in Fig. 3.3 have the same setting as single frequency "lowest" except that σ_N is set to 3 instead of 5. That is to avoid the accumulation of noise in the artificial images.

In total we generate 14 basis types of artificial DVFs: 5 single frequency, 4 mixed frequency, 4 respiratory motion and 1 identity. The precise settings of the parameters are available in Table 3.1 and examples are given in Fig. 3.4. The histograms of the Jacobians are also available in this figure. When the spatial frequency is increased, the Jacobian histograms will spread more, which shows that local relative volume changes are increased. The value of θ, the maximum artificial displacement along each axis, is chosen as 20, 15 and 7 for RegNet^4, RegNet^2 and RegNet^1, respectively.



(a) single freq. “lowest” (b) single freq. “intermediate” (c) single freq. “highest”



(d) mixed freq. “lowest” (e) mixed freq. “intermediate” (f) mixed freq. “high”

Figure 3.4: Examples of heat maps of generated artificial DVFs overlayed on the deformed images. We show three of the five spatial frequencies defined in Table 3.1. The histogram of the Jacobian determinant of each DVF is shown next to the sample image. As the spatial frequency increases, the histogram is more spread.

Table 3.1: DVFs with different spatial frequencies are obtained by varying the B-spline grid spacing s and the standard deviation of the Gaussian kernel σ_B. The maximum deformation along each axis θ only varies for each stage. When the spatial frequency is increased, the Jacobian histograms will spread more, which shows that local relative volume changes are increased (Fig. 3.4). S, M and R indicates single frequency, mixed frequency and respiratory motion.

Parameter	artificial DVF	lowest	low	intermediate	high	highest
	stage 1	3	7	7	7	7
θ (mm)	stage 2	5	15	15	15	15
	stage 4	7	20	20	20	20
	S^1	[50, 50, 50]	[45, 45, 45]	[35, 35, 35]	[25, 25, 25]	[20, 20, 20]
	S^2	[60, 60, 60]	[50, 50, 50]	[45, 45, 45]	[40, 40, 40]	[35, 35, 35]
	S^4	[80, 80, 80]	[70, 70, 70]	[60, 60, 60]	[50, 50, 50]	[45, 45, 45]
	M^1	[50, 50, 50]	[40, 40, 40]	[25, 25, 35]	[20, 20, 30]	
s (mm)	M^2	[60, 60, 60]	[50, 50, 40]	[40, 40, 80]	[35, 35, 80]	
	M^4	[80, 80, 80]	[60, 60, 60]	[50, 50, 50]	[45, 45, 60]	
	R^1	[50, 50, 50]	[45, 45, 45]	[35, 35, 35]	[25, 25, 25]	
	R^2	[60, 60, 60]	[50, 50, 50]	[45, 45, 45]	[40, 40, 40]	
	R^4	[80, 80, 80]	[70, 70, 70]	[60, 60, 60]	[50, 50, 50]	
	M^1	(5-10)	(5-10)	(5-10)	(5-10)	
σ_B	M^2	(7-12)	(7-12)	(7-12)	(7-12)	
	M^4	(10-15)	(10-15)	(10-15)	(10-15)	

3.2.4 Optimization

Optimization is done using the Adam optimizer with a learning rate of 0.001. The loss function consists of two parts. The first part is the Huber loss, which minimizes the difference between the ground truth $\boldsymbol{T}$ and the predicted DVF $\boldsymbol{T'}$ of the RegNet. The second part is a bending energy (BE) regularizer [6], which ensures smoothness of the displacement field:

$$\mathscr{C} = \text{Huber}\big(\boldsymbol{T}(\boldsymbol{x}), \boldsymbol{T'}(\boldsymbol{x})\big) + \gamma \cdot \text{BE}\big(\boldsymbol{T'}(\boldsymbol{x})\big), \tag{3.2}$$

where the Huber loss is defined as:

$$\text{Huber}(\boldsymbol{T}, \boldsymbol{T'}) = \begin{cases} (\boldsymbol{T} - \boldsymbol{T'})^2, & |\boldsymbol{T} - \boldsymbol{T'}| \leq 1, \\ |\boldsymbol{T} - \boldsymbol{T'}|, & |\boldsymbol{T} - \boldsymbol{T'}| > 1 \end{cases} \tag{3.3}$$

3.3 Experiments and results

3.3.1 Materials and ground truth

Three chest CT scan datasets are used in this study: The SPREAD [47], the DIR-Lab-4DCT [74] and the DIR-Lab-COPDgene dataset [75].

In the SPREAD database, 21 pairs of 3D chest CT images are available with a baseline and a follow-up image in each pair. The follow-up images are taken after 30 months. Both images are acquired in the inhale phase. Patients in this study are aged between 49 and 78 years old. The size of the images is approximately $446 \times 315 \times 129$ with a mean voxel size of $0.78 \times 0.78 \times 2.50$ mm. About 100 well-distributed corresponding landmarks were previously selected [73] semi-automatically on distinctive locations [48]. Two cases (12 and 19) are excluded because of the high uncertainty in the landmarks annotation [73].

In the DIR-Lab-COPDgene database, ten cases with severe breathing disorders are available in inhale and exhale phases. The average image size and the average voxel size are $512 \times 512 \times 120$ and $0.64 \times 0.64 \times 2.50$ mm, respectively. In each pair, 300 landmarks are annotated.

In the DIR-Lab-4DCT database ten cases are available. We use two phases of the available data: maximum inhalation and maximum exhalation. The size of the images is about $256 \times 256 \times 103$ with an average voxel size of $1.10 \times 1.10 \times 2.50$ mm.

Since the convolutional neural networks process the images in a voxel-based manner, all images are resampled to an isotropic voxel size of $1.0 \times 1.0 \times 1.0$ mm.

3.3.2 Evaluation measures

We use two measures to evaluate the performance of the proposed CNNs:

- **TRE:** The target registration error (TRE) defined as the mean Euclidean distance after registration between corresponding landmarks:

$$\text{TRE} = \frac{1}{n}\sum_{i=1}^{n} \| \boldsymbol{T}'(\boldsymbol{x}_{Fi}) + \boldsymbol{x}_{Fi} - \boldsymbol{x}_{Mi} \|_2, \tag{3.4}$$

  where $\boldsymbol{x}_F$ and $\boldsymbol{x}_M$ are the landmark locations on the fixed and moving images, respectively.

- **Jac:** The Determinant of the Jacobian of the predicted DVF is calculated in order to measure relative changes in local volume. A very large (Jac $\gg 1$) or very small (Jac $\ll 1$) or negative Jac (Jac < 0) can indicate poor registration quality. We report the percentage of negative Jacobian as well as the standard deviation of the Jacobian inside the lung masks.

All of the assessments are performed on the real images.

3.3.3 Experimental setup

3.3.3.1 Training data

In the SPREAD database, 10 patients (20 images) are used for training, 1 patient (2 images) is in the validation set and 8 patients remain for the test set. From the DIR-Lab-COPD database, the first 9 cases (18 images) are used for training, and the remaining case (2 images) is used in the validation set. The entire DIR-Lab-4DCT database is used as an independent test set. The validation set is mainly used for tuning the hyper-parameters and selecting the best network design. In all evaluations, images are multiplied with the lung masks.

To generate training pairs, we use the 14 basis types of artificial generations (see Section 3.2.3.3). For each of the three networks, from each original image we generate 70 (5×basis), 42 (3×basis) and 28 (2×basis) artificial pairs in the first stage (RegNet^4), the second stage (RegNet^2) and the third stage (RegNet^1), respectively. Here we generate more images for more coarse stages, as these images are smaller.

In the training phase of the patch-based networks (MV, Uadv), the batch size is 15. The number of patches per pair is 5, 20, and 50 for stage 4, 2, and 1, respectively. The patch size is 101^3 and 105^3 for the U-Net-advanced and Multi-view design. When choosing samples, several balancing criteria are considered based on the magnitude of DVFs of the patches. An equal number of samples are selected from the range [0, 1.5), [1.5, 8) and [8, 20) mm for stage 4. For stage 2 and 1 these bins are selected as [0, 1.5), [1.5, 4), [4, 15) mm and [0, 2), [2, 7) mm, respectively. Training is run for 30 semi-epochs. All methods are trained with an additional data augmentation step, by adding Gaussian noise to all patches on the fly.

3.3.3.2 Software

In order to efficiently implement the artificial deformation and training phase, we utilize two processes. The task of the first process is to create artificial DVFs and deformed images and write them to disk. The second process has a multithreading paradigm which loads the data from disk and also handles the network training on the GPU.

The CNNs are implemented in Tensorflow [76]. Artificial DVFs are generated with the help of SimpleITK [51]. The code is publicly available via github.com/hsokooti/RegNet.

3.3.3.3 `elastix`

We compare the proposed CNN-based registration methods with conventional image registration, using `elastix` [52]. We used the following settings: metric: mutual information, optimizer: adaptive stochastic gradient descent, transform: B-spline, number of resolutions: 3, number of iterations per resolution: 500. For the public DIR-Lab-4DCT data, more conventional and CNN-based methods are compared with RegNet in Section 3.3.4.2.

3.3.4 Experiments

3.3.4.1 Architecture selection

In order to inspect the performance of the different architectures, an evaluation is performed on all pairs in the training and validation sets, i.e. half of the SPREAD data and the entire DIR-Lab-COPDgene data. We utilize the single and mixed category plus identity transform for artificial generations. Please note that the networks are trained with artificial image pairs i.e. during training both the fixed and moving images are deformed versions of the original images. For this evaluation however, we used the original non-deformed pairs, which the network has not seen.

As a first experiment, we train and validate the networks on the original image resolution only, i.e. without any multi-stage pipeline: see MV^1 and $Uadv^1$ in Section 3.2. It can be seen that the TREs of these networks are on the high end for both studies. Please note that due to high intensity variation in the baseline and follow-up images in the DIR-Lab-COPDgene database, the overall results are relatively poor. We discuss this issue later in Section 3.4.

In a second experiment, we train and test the networks on the lowest image resolution only, again without any multi-stage pipeline: see U^4, MV^4 and $Uadv^4$ in Table 3.2. Note that on the SPREAD data, the performance improved with respect to registration on the original resolution. The main reason is that the lowest resolution training set has the maximum deformation θ of 20 mm, whereas the maximum deformation was set to 7 mm in the original resolution training set (see Table 3.1). On the DIR-Lab-COPDgene data, similar results were obtained except for U^4.

Table 3.2: Quantitative results on the training and validation sets. The target registration error (TRE) is reported, together with the percenage of folding and the standard deviation of the Jacobian inside the lung masks. The networks are trained using artificial deformations from the single and mixed category plus identity (see Section 3.2.3.1). U, Uadv and MV represent the U-Net, U-Net-advanced and Multi-view design (see Section 3.2.2). A Wilcoxon signed-rank test is performed between U^4-$Uadv^2$-$Uadv^1$ and others, where † indicates a statistically significant difference with $p < 0.05$.

Network	SPREAD (case 1-11)			DIR-Lab-COPDgene		
	TRE (mm)	%folding	std(Jac)	TRE (mm)	%folding	std(Jac)
MV^1	3.86±4.32†	0.23±0.18	0.23±0.05	9.28±5.83†	0.24±0.07	0.29±0.03
$Uadv^1$	3.80±4.15†	0.24±0.20	0.28±0.06	9.65±6.19†	0.32±0.13	0.35±0.05
U^4	2.71±1.59†	0.00±0.00	0.09±0.02	10.2±6.00†	0.00±0.00	0.10±0.01
MV^4	2.30±1.80†	0.00±0.00	0.11±0.02	8.27±5.44†	0.00±0.00	0.14±0.02
$Uadv^4$	2.29±1.89†	0.00±0.00	0.08±0.01	8.60±5.50†	0.00±0.00	0.12±0.01
MV^4-MV^2	1.70±1.31†	0.00±0.01	0.18±0.03	6.67±5.53†	0.01±0.01	0.28±0.05
U^4-MV^2	1.71±1.23†	0.00±0.00	0.15±0.03	8.94±6.95†	0.03±0.02	0.27±0.06
$Uadv^4$-$Uadv^2$	1.69±1.34†	0.00±0.00	0.12±0.02	6.96±5.89†	0.00±0.00	0.21±0.04
U^4-$Uadv^2$	1.68±1.15†	0.00±0.00	0.11±0.02	8.54±6.91†	0.00±0.00	0.20±0.04
MV^4-MV^2-MV^1	1.63±1.30†	0.05±0.07	0.22±0.04	6.35±5.74†	0.44±0.24	0.38±0.08
U^4-MV^2-MV^1	1.60±1.20	0.02±0.03	0.19±0.04	8.65±7.27†	0.49±0.27	0.38±0.09
$Uadv^4$-$Uadv^2$-$Uadv^1$	1.60±1.23	0.03±0.04	0.22±0.03	6.45±6.39†	0.29±0.20	0.37±0.10
U^4-$Uadv^2$-$Uadv^1$	1.57±1.15	0.02±0.02	0.20±0.03	8.07±7.65	0.39±0.24	0.38±0.10

In the next experiment, we utilized two stages at image resolutions downsampled with a factor of 4 and then 2. In all four tested architectural combinations, the TRE results are better than the single stage networks in both studies, which shows that adding a second stage can improve the performance of RegNet.

Finally, when the original resolution is added to form a three-stage network a small improvement is observed in both studies. By comparing the final TRE results, it can be seen that the performance of all four network combinations are similar. For the remainder of the experiments, we choose the combination (U^4-$Uadv^2$-$Uadv^1$) as it obtained slightly better results on the SPREAD database. A Wilcoxon signed-rank test is performed between U^4-$Uadv^2$-$Uadv^1$ and other combination in Table 3.2. A statistically significant difference (with $p < 0.05$) between U^4-$Uadv^2$-$Uadv^1$ and all single stage and two stages combination can be observed.

3.3.4.2 Independent test set experiments

Now that we have selected the best network combination, we applied the U^4-$Uadv^2$-$Uadv^1$ pipeline on the independent test set (without retraining): 8 cases of the SPREAD

Table 3.3: Quantitative results of the SPREAD study in the training set (case 1 to case 11) and in the test set (case 13 to case 21). This experiment is performed with the network combination U[4]-Uadv[2]-Uadv[1]. The target registration error (TRE) is reported, together with the percentage of folding and the standard deviation of the Jacobian inside the lung masks. S, M and R indicate single frequency, mixed frequency and respiratory motion, respectively (see Section 3.2.3.1). A Wilcoxon signed-rank test is performed between the B-spline registration and others. The symbol † indicates a significant difference between the average of TRE of B-spline registration and others, where † indicates a statistically significant difference with $p < 0.05$. The best method is shown in bold and the second best method is shown in green.

	`elastix`	`elastix` B-spline			RegNet				
pair	Affine TRE (mm)	TRE (mm)	%folding	std(Jac)	S TRE (mm)	S+M TRE (mm)	S+M+R TRE (mm)	S %folding	S std(Jac)
case 1	8.77±2.76†	2.13±2.12	0.00	0.19	1.87±1.68	**1.78±1.66**	1.92±1.71	0.20	0.26
case 2	7.41±2.99†	1.48±1.18	0.00	0.11	1.44±0.92	**1.37±0.88**	1.54±1.03	0.03	0.20
case 3	4.34±1.89†	**1.66±1.13**	0.00	0.09	1.78±1.18	1.79±1.21	1.81±1.19†	0.00	0.15
case 4	11.4±3.44†	1.79±1.43	0.00	0.15	**1.70±1.41**	1.70±1.29	1.99±1.74	0.07	0.21
case 5	6.47±2.07†	**1.08±0.62**	0.00	0.09	1.15±0.76	1.17±0.70	1.24±0.78†	0.00	0.15
case 6	8.22±2.37†	2.06±1.37	0.00	0.14	1.98±1.44	**1.90±1.27**	1.92±1.22	0.02	0.21
case 7	5.51±1.38†	**1.50±1.11**	0.00	0.10	1.70±1.07†	1.63±1.22	1.51±1.10	0.00	0.17
case 8	3.67±2.31†	1.70±1.23	0.00	0.14	1.74±1.02	1.63±0.94	**1.53±0.84**	0.01	0.20
case 9	4.93±1.61†	**1.28±0.72**	0.00	0.09	1.35±0.72	1.41±0.87	1.51±0.77†	0.02	0.16
case 10	6.22±2.27†	**1.33±1.11**	0.00	0.10	1.40±0.90	1.40±0.87	1.43±0.85†	0.02	0.18
case 11	5.93±2.20†	**1.40±1.10**	0.00	0.13	1.49±1.15	1.44±1.19	1.51±1.08	0.03	0.21
Total	6.62±3.17†	1.58±1.29	0.00±0.00	0.12±0.03	1.60±1.17†	**1.57±1.15**	1.63±1.19†	0.04±0.05	0.19±0.03
case 13	12.5±15.8†	7.94±16.0	0.00	0.13	**7.37±14.0**	7.76±15.2	8.28±15.4	0.71	0.36
case 14	8.99±2.40†	1.86±1.19	0.00	0.11	**1.71±1.18**	2.08±1.81	2.25±1.76†	0.06	0.25
case 15	3.17±1.32†	**1.20±0.82**	0.00	0.11	1.39±0.86	1.29±0.82	1.33±0.84	0.00	0.18
case 16	8.94±1.84†	**1.30±0.80**	0.00	0.09	1.54±0.96	1.78±0.98†	1.96±1.01†	0.00	0.19
case 17	13.4±4.73†	**1.76±0.73**	0.00	0.09	2.89±3.66†	2.30±1.70†	3.37±3.43†	0.38	0.27
case 18	7.85±2.89†	1.65±1.41	0.00	0.15	**1.40±0.86**	1.60±1.16	1.71±1.04	0.02	0.21
case 20	4.43±2.14†	**1.31±0.90**	0.00	0.11	1.41±1.00	1.50±1.05†	1.52±0.97†	0.14	0.22
case 21	6.48±2.03†	**1.26±1.35**	0.00	0.09	1.36±1.19	1.33±1.36	1.36±1.47†	0.01	0.17
Total	8.16±6.76†	**2.21±5.86**	0.00±0.00	0.11±0.02	2.32±5.33†	2.39±5.64†	2.65±5.82†	0.19±0.24	0.24±0.06

database and the complete DIR-Lab-4DCT database. The results are given in Tables 3.3 and 3.4.

For the SPREAD database, the TRE results with affine and B-spline registration are compared with three versions of RegNet trained using the category "S" (single frequency plus identity), "S+M" (single frequency and mixed frequency plus identity) and "S+M+R" (single frequency plus mixed frequency and respiratory motion plus identity). Since there is no respiratory motion in the SPREAD data, adding respiratory motion did not improve the performance of the registration. Adding mixed frequencies did not change the results considerably: there was a small improvement for the cases 1-11, and slightly larger TREs for the cases 13 to 21. The percentage of folding inside the lung masks for the RegNet trained using "S" is also available in Table 3.3, which reports that the percentage of negative Jacobian are small in most cases, especially, when the TRE after affine registration is not very large. A Wilcoxon signed-rank test is performed between the `elastix` B-spline and other results. It can be seen that in

Table 3.4: Quantitative results on the DIR-Lab-4DCT study. This experiment is performed with the network combination of U^4-Uadv^2-Uadv^1. The target registration error (TRE) is reported, together with the percentage of folding and the standard deviation of the Jacobian inside the lung masks. S, M and R indicate single frequency, mixed frequency and respiratory motion, respectively (see Section 3.2.3.1). The result of [77] is the average of all respiratory phases per each case. The best method is shown in bold and the second best method is shown in green.

	elastix Affine	elastix B-spline	[78]	[77]	Vos et al. [23]	Eppenhof et al. [79]	Eppenhof et al. [79]-DIR	RegNet				
								S	S+M	S+M+R	S+M+R	S+M+R
pair	TRE (mm)	TRE (mm)	TRE (mm)	TRE (mm)	TRE (mm)	TRE (mm)	TRE (mm)	TRE (mm)	TRE (mm)	TRE (mm)	%fold-ing	std(Jac)
case 01	3.02±2.13	1.21±0.71	**1.00±0.52**	1.20±0.60	1.27±1.16	1.45±1.06	-	1.09±0.51	1.12±0.54	1.13±0.51	0.00	0.10
case 02	3.76±3.20	1.39±1.27	**1.02±0.57**	1.19±0.63	1.20±1.12	1.46±0.76	1.24±0.61	1.08±0.89	1.06±0.57	1.08±0.55	0.00	0.12
case 03	5.92±3.64	2.44±2.11	**1.14±0.89**	1.67±0.90	1.48±1.26	1.57±1.10	-	1.23±0.69	1.23±0.75	1.33±0.73	0.00	0.14
case 04	9.01±4.53	2.16±2.16	**1.46±0.96**	2.53±2.01	2.09±1.93	1.95±1.32	1.70±1.00	1.47±0.95	1.62±1.09	1.57±0.99	0.00	0.18
case 05	3.95±2.85	3.02±3.22	1.61±1.48	2.06±1.56	1.95±2.10	2.07±1.59	-	**1.58±1.33**	1.60±1.33	1.62±1.30	0.00	0.14
case 06	10.7±6.80	3.33±3.30	**1.42±1.71**	2.90±1.70	5.16±7.09	3.04±2.73	-	4.56±7.06	4.95±6.91	2.75±2.91	0.03	0.25
case 07	11.1±7.43	6.16±6.33	**1.49±4.25**	3.60±2.99	3.05±3.01	3.41±2.75	-	6.10±7.10	5.00±6.35	2.34±2.32	0.03	0.24
case 08	12.0±6.59	9.36±9.30	**1.62±1.71**	5.29±5.52	6.48±5.37	2.80±2.46	-	6.54±8.51	6.18±7.01	3.29±4.32	0.01	0.22
case 09	7.89±3.83	3.31±2.74	**1.30±0.76**	2.38±1.46	2.10±1.66	2.18±1.24	1.61±0.82	2.02±2.25	1.84±1.93	1.86±1.47	0.00	0.17
case 10	6.87±6.12	2.72±3.43	**1.50±1.31**	2.13±1.88	2.09±2.24	1.83±1.36	-	2.82±4.93	2.44±3.85	1.63±1.29	0.00	0.19
Total	7.43±5.92	3.51±4.83	**1.36±1.01**	2.50±1.16	2.64±4.32	2.17±1.89	-	2.85±4.96	2.70±4.39	1.86±2.12	0.01±0.01	0.18±0.05

most cases there is no significant difference between B-spline registration and RegNet trained using "S" or trained using "S+M".

For the DIR-Lab-4DCT database, a comparison between RegNet and affine, B-spline (three resolutions), an advanced conventional registration method using sliding motion [78] and three other CNN-based methods [79, 23, 77] is available in Table 3.4. It can be seen that training with "S+M" improved performance slightly with respect to just "S". Adding the respiratory motion category improved performance substantially, as these are inhale-exhale pairs; this is predominantly caused by the patients where the TRE after affine registration was still quite large. An example visualization is also available in Fig. 3.5, showing that adding the respiratory motion category can align images better in the diaphragm region. The advanced conventional registration method that leverages sliding motion [78] is still better than RegNet. Note that RegNet was not trained on the DIR-Lab-4DCT data, similar to [79, 77]. However, Vos et al. [23] and Eppenhof et al. [79] DIR methods were trained on the same database but using cross-validation to report the results. Also note that the results reported in [77] are averaged over all phases of DIR-Lab-4DCT (T00 to T10), while the results of other CNN methods (including RegNet) are reported between the maximum inhale and maximum exhale phase (T00 and T50). These reported results are therefore likely somewhat better than the results for T00 and T50 only.

3.3.4.3 Inference

At inference time, the patch size can be enlarged depending on the available GPU memory. For the U-Net-advanced design, the inference time of an image of size 101^3 and 269^3 voxels, is 0.02 s and 2.4 s, respectively, on our TITAN Xp (12 GB). An image of size 273^3 voxels took about 2.1 s to process for the Multi-view design. For the U-Net design we used the downsized image (by a factor of 4) of size 125^3 which took 0.02 s to be processed.

3.4 Discussion

In this paper we have shown that training a CNN with sufficiently realistic artificially generated displacement fields, can yield accurate registration results even in real cases. We utilized some randomly generated deformations (single and mixed frequencies) and a more realistic one (respiratory motion). We observed that even training with randomly generated deformations in the SPREAD study, the obtained TRE was on par with the B-spline registration (see Table 3.3). Adding more realistic DVFs (respiratory motion) in the DIR-Lab 4DCT study, improved the TRE results from 2.70±4.39 mm ("S+M") to 1.86±2.12 mm ("S+M+R") as can be seen in Table 3.4. In the case that sufficient realism was not added to the training, for instance in the DIR-Lab-COPDgene study in Table 3.2, the results were sub-optimal. Note that this dataset is challenging for conventional methods also. Anatomical structures in the baseline and follow-up

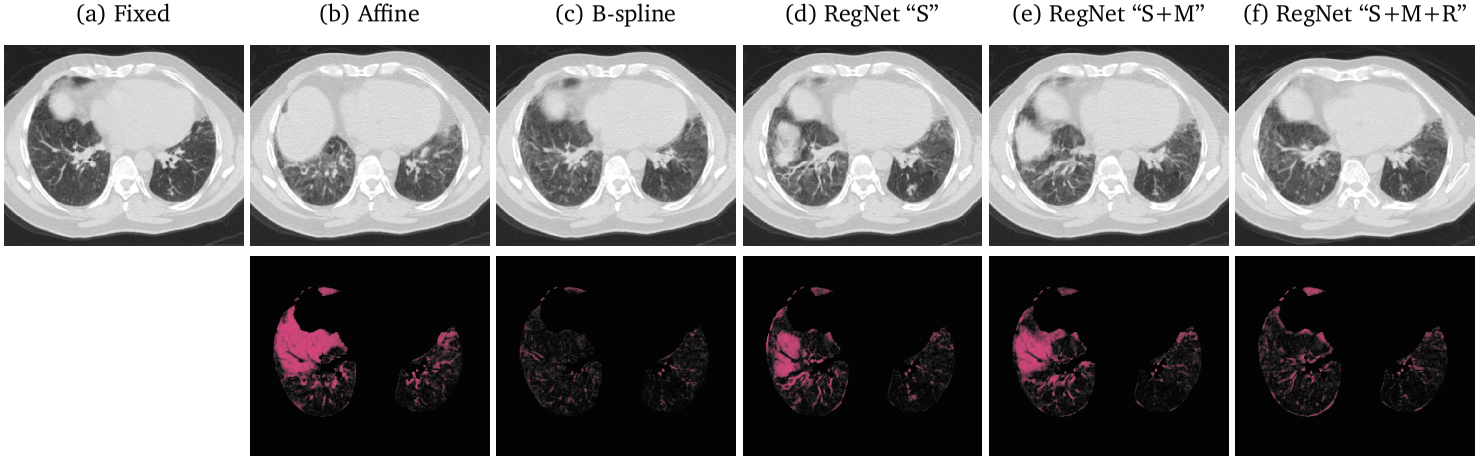


Figure 3.5: Example results (top row) and difference images (bottom row) from DIR-Lab-4DCT study.

images of this database are quite different and the proposed intensity simulations in Section 3.2.3.2 did not cover this issue. A solution may be the addition of random intensity occlusions to the deformed images. Another interesting research direction is to learn realistic appearance and deformations from a database.

One of the major challenges of CNN-based image registration is capturing large DVFs, especially for patch-based methods. Using the whole image as an input might be very time consuming in the training phase. Based on our experiments, the maximum deformation that can be detected in patch with size $101 \times 101 \times 101$ by the U-Net-advanced is approximately 7 mm (the value of θ in Table 3.1). By enlarging the DVFs the Huber loss increased substantially. Please note that the maximum deformation θ is along each axis so the magnitude of a maximum deformation is $2 \times \sqrt{3} \times \theta$. Adding the original resolution makes the pipeline slower. However, based on the results in Table 3.2 it can be concluded that using two stages (U^4-$Uadv^2$, TRE: 1.68±1.15) can achieve similar results in comparison with three stages (U^4-$Uadv^2$-$Uadv^1$, TRE: 1.57±1.15). Similar to conventional registration, the best method on the first stage is not always the best in combination with others. In Table 3.2, the single U^4 is worse than $Uadv^4$ and MV^4. Conversely, the combination of U^4-$Uadv^2$-$Uadv^1$ obtains the best results. All in all, the differences between the architectures are relatively small and the performance of RegNet is more influenced by the artificial images used in training phase.

The performance of RegNet is very close to the conventional registration. However, B-spline registration is the best in the SPREAD test set (case 13 to 21; 2.21 ± 5.86 vs 2.32 ± 5.33) and the method of Berendsen et al. [78] performed better in the DIR-Lab-4DCT database (1.36 ± 1.01 vs 1.86 ± 2.12). On the contrary, the inference time of CNN approaches are much faster than the conventional methods. Potentially, by increasing the training data and generalizing the artificial generation like sliding motion, the performance of the RegNet can be improved.

In the current implementation of RegNet, all images are resampled to an isotropic voxel size $1.0 \times 1.0 \times 1.0$ mm. If resampling is not intended, it might be possible to simply multiply the output of RegNet by the voxel size. However, this approach is not very accurate because the spatial frequency of different voxel size might not be covered by the training data. A more accurate solution could be to include additional input of the voxel size to the network.

In principle, the proposed network design potentially can be utilized to predict the registration quality. Several methods are suggested by conventional learning using handcrafted features [55, 39] and a preliminary result by [44].

The proposed method can be trained and evaluated on other image modalities like brain MRI images. Potentially, the same network design and artificial generation excluding respiratory motion can be utilized.

The artificial generation can be enhanced if rib segmentation is available. Then, it is possible to incorporate rigid deformation outside of the rib and nonrigid deformations inside the rib. The network potentially can learn the relation between organs and rigidity of the deformations. More realistic and complex simulation like sliding motion of lungs [78] can also be added to the training images as it had a positive effect for non-learning based methods.

3.5 Conclusion

We proposed a 3D multi-stage CNN framework for chest CT registration. For training the network, we proposed models to generate artificial DVFs, and intensity models, to easily generate large quantities of paired images with a known spatial relation. We showed via multiple chest CT databases that this way of artificial training is very effective, with good results on real data. On the public DIR-Lab-4DCT database, we achieved the best results among the CNN approaches.