# Pulmonary CT Registration Through Supervised Learning With Convolutional Neural Networks

Koen A. J. Eppenhof and Josien P. W. Pluim, *Fellow, IEEE*

***Abstract*— Deformable image registration can be time consuming and often needs extensive parameterization to perform well on a specific application. We present a deformable registration method based on a 3-D convolutional neural network, together with a framework for training such a network. The network directly learns transformations between pairs of 3-D images. The network is trained on synthetic random transformations which are applied to a small set of representative images for the desired application. Training, therefore, does not require manually annotated ground truth information on the deformation. The framework for the generation of transformations for training uses a sequence of multiple transformations at different scales that are applied to the image. This way, complex transformations with large displacements can be modeled without folding or tearing images. The methodology is demonstrated on public data sets of inhale–exhale lung CT image pairs which come with landmarks for evaluation of the registration quality. We show that a small training set can be used to train the network, while still allowing generalization to a separate pulmonary CT data set containing data from a different patient group, acquired using a different scanner and scan protocol. This approach results in an accurate and very fast deformable registration method, without a requirement for parameterization at test time or manually annotated data for training.**

***Index Terms*— Deformable image registration, pulmonary CT images, convolutional neural networks, machine learning.**

## I. Introduction

A LARGE class of deformable image registration problems is solved using algorithms that maximize an image similarity function defined on the space of transformation parameters [1]. Optimization-based methods usually result in very good registration accuracy, but suffer from being computationally expensive, especially for complex transformations and high resolution images. This results in these algorithms having long runtimes, which makes them unfit for clinical applications in which realtime registration is desired, such as image-guided surgery and radiation treatment where fast registration methods can contribute to better correction for the patient's movement. Optimization based methods require different parameterizations of the registration algorithm for different applications (e.g. anatomy, modality, patient population), for example the choice of similarity metric and optimization method. These registration parameters need to be tuned manually, and it can take many experiments to determine parameters that result in a robust registration for the desired application. In medical image registration, the similarity function is not necessarily convex and can have many local maxima at points where the transformation causes similar – but not the same – structures to overlap.

To address these issues, we propose a fast deformable image registration framework based on a convolutional neural network (CNN). Although the computational expense of training the network is large, the use of the network for new images only requires one forward-pass through the network, which can be performed in much less time than the more conventional optimization-based methods. Because the network is trained to perform registration for a specific population, it is not necessary to manually optimize any registration parameters. Rather than optimizing a similarity metric, the network is explicitly trained to reduce registration errors.

### A. Related Work

Deep learning methods have proven very successful in a variety of tasks in medical image analysis [2]. Applications of deep learning to deformable image registration have emerged recently. These methods vary in their objectives, ranging from aiding conventional registration methods, to estimating a transformation model or deformation field directly.

Methods that aid the optimization of the transformation for example learn similarity metrics for multimodal image registration, the initialization of the optimization, or an optimization update. Simonovsky *et al.* [3] created a supervised method that can learn similarity metrics from patches of already aligned multi-modal three-dimensional T1 and T2 brain MRI images. Wu *et al.* [4] used feature maps learned by an unsupervised auto-encoder as features for deformable registration of brain MRIs. Gutiérrez-Becker *et al.* [5] developed a supervised method that can predict optimization steps when optimizing rigid and deformable two-dimensional transformations. Their method was trained and applied on intravascular ultrasound

Manuscript received August 28, 2018; revised October 12, 2018; accepted October 21, 2018. Date of publication October 26, 2018; date of current version May 1, 2019. *(Corresponding author: Koen A. J. Eppenhof.)*

K. A. J. Eppenhof is with the Department of Biomedical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands (e-mail: k.a.j.eppenhof@tue.nl).

J. P. W. Pluim is with the Department of Biomedical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands, and also with the Image Sciences Institute, University Medical Center Utrecht, 3548 CX Utrecht, The Netherlands.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMI.2018.2878316

and histology images. Yang *et al.* [6] proposed a supervised method that estimates the momentum parameterization for LDDMM shooting from image patches, which was demonstrated for registration of brain MR images.

Methods that predict registration models directly can be divided into supervised and unsupervised methods. One of the earliest publications on estimating deformations directly from two input images using supervised CNNs attempted to learn optical flow in 2D natural images [7]. The authors generated their own ground truth training set by applying affine transformations to parts of images. In a similar way this has been applied to medical images, but for estimating deformable transformations using patch-based CNNs [8], [9]. Cao *et al.* [10] used a similar technique but informed the neural network with a patch similarity map. Rohé *et al.* [11] used a fully convolutional neural network to perform direct estimation of a Stationary Velocity Field transformation model, trained on previously registered cardiac images. Spatial transformer networks [12] have been used to learn transformation models for unsupervised image registration of 2D medical images by de Vos *et al.* [13]. They trained the network by backpropagating a similarity metric between the transformed moving image and the fixed image.

In this paper, we substantially extend the supervised deformable image registration approach we presented in [14], in which a convolutional neural network is used to perform fast image registration of 3D pulmonary CT images. This paper significantly improves the complexity of the transformations used in the training set. Compared to existing deep-learning based registration applications, pulmonary CT registration requires a fine-grained or local deformation field combined with relatively large displacements. For this kind of deformation we propose a training method that combines multiple random transformations to generate a large training set. The network is not trained to optimize similarity but to minimize the registration error directly. Furthermore, we show that the proposed network performs fast image registration compared to existing methods, without sacrificing registration accuracy.

## II. Materials

To train and test the convolutional neural network, we use two separate sets of publicly available thoracic computed tomography scans. The sets were acquired at different hospitals, using different scanners and protocols. Each instance in the set consists of a pair of inspiration/expiration scans of the same patient. During training, we use both the inspiration and expiration images of the CREATIS dataset and POPI model [15], [16]. This set consists of 4D CTs showing a full breathing cycle, acquired for the purpose of radiotherapy planning on a Philips 16-slice Brilliance Big Bore Oncology Configuration (Philips Medical Systems, Cleveland, Ohio, USA) gated by a respiratory surrogate signal form the Pneumo Chest pressure belt (Lafayette Instrument, Lafayette, Indiana, USA). From this set, we used seven pairs of images at inspiration and expiration to train the network. The images have voxel sizes ranging from $0.78 \times 0.78 \times 2.00$ to $1.17 \times 1.17 \times 2.00$ $mm^3$ and a $482 \times 360 \times 139$ to $512 \times 512 \times 187$ voxel dimension.

To validate our methodology and test generalization to different data, we use the trained network to register expiration images (moving) to inspiration images (fixed) from a separate set of test data, the DIR-Lab dataset [17], [18]. This data set consists of ten pairs of inspiration/expiration thoracic CT images of patients treated for thoracic malignancies. The scans were acquired on a Discovery ST PET/CT scanner (GE Medical Systems, Waukesha, Wisconsin, USA), gated by using the respiratory signal from the Real-Time Position Management Respiratory Gatin System (Varian Medical Systems, Palo Alto, California, USA). The pairs of images come with expert-annotated corresponding landmarks, showing an initial misregistration error of 8.46 mm. Each pair of images has 300 landmarks expressed in voxel indices that have been annotated by a single observer, but are partly annotated by two additional observers to measure the inter-observer reproducibility which ranged from $0.70 \pm 1.01$ to $1.13 \pm 1.27$. The images have voxel sizes ranging from $0.97 \times 0.97 \times 2.5$ to $1.16 \times 1.16 \times 2.5$ $mm^3$ and voxel dimensions ranging from $256 \times 256 \times 94$ to $512 \times 512 \times 136$.

## III. Methods

Let $I_F : \Omega_F \rightarrow \mathbb{R}$ and $I_M : \Omega_M \rightarrow \mathbb{R}$ be two images sampled on their own $d$-dimensional domains $\Omega_F, \Omega_M \subset \mathbb{R}^d$. Registration aims to find the transformation $\mathbf{T} : \Omega_F \rightarrow \Omega_M : \mathbf{x} \mapsto \mathbf{x} + \mathbf{u}(\mathbf{x})$. In this paper we assume the images are pre-registered using an affine transformation $\mathbf{T}(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$. The task at hand is to estimate the remaining deformable transformation from the affinely registered images.

The vector field $\mathbf{u}(\mathbf{x})$ is estimated from two three-dimensional images using a convolutional neural network. The network's output is the three-dimensional displacement vector field $\mathbf{u}(\mathbf{x})$ represented as three maps $u_x(\mathbf{x})$, $u_y(\mathbf{x})$, $u_z(\mathbf{x})$ for the displacement in all three dimensions. These maps cover the full fixed image domain, with the network returning a displacement vector for every voxel.

### A. Data Preparation

The network accepts $128 \times 128 \times 128$ images in the current implementation, limited by the available GPU RAM required to keep the full network in memory. Therefore, all images are resized and cropped by removing the outer border of the images. We determine this border using lung masks, which were created by segmenting voxels with Hounsfield units below -250, resulting in a rough segmentation of voxels corresponding to low density, i.e. the lungs and the exterior of the patient. After setting the largest morphological component (the patient's exterior) to zero we obtain a rough segmentation of the lungs. Subsequently, the resulting images were resized to a $128 \times 128 \times 128$ resolution using fourth-order B-spline interpolation and the landmarks are translated to compensate for the border removal, and scaled to compensate for the resizing.

### B. End-to-End Training

The network is trained end-to-end: two full images are used as input, and the output is a displacement vector field for

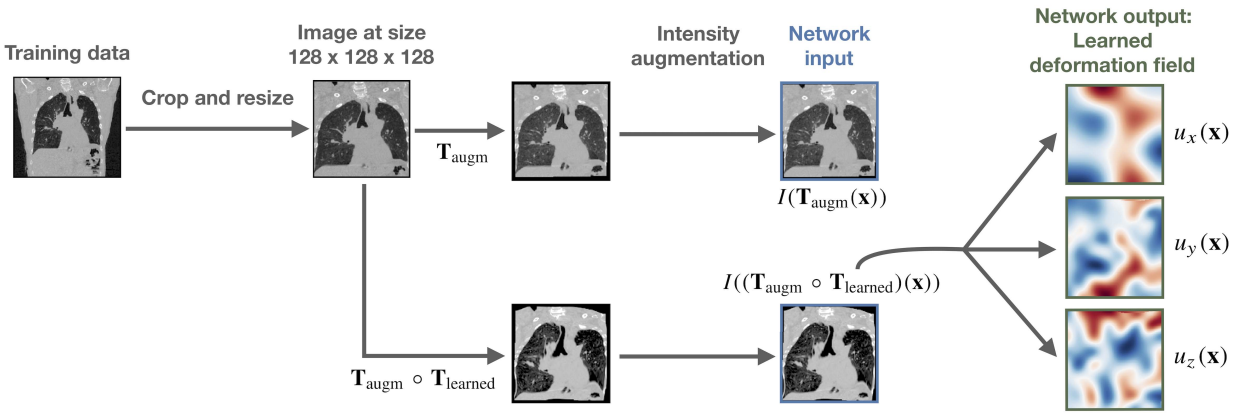


Fig. 1. Schematic explaining the on-the-fly training set construction. From the CREATIS data set, fourteen images are used for training the neural network. Each iteration of training an image is selected and randomly deformed using the augmentation transformation, which serves to augment the training set. This image is again deformed, using the learned transformation (deformation field shown in blue and red for positive and negative displacements).

the full image domain. This is in contrast with patch-based methods (e.g. [6], [8], [9]) that estimate the deformation from small patches from the pair of images. Another disadvantage of patch-based networks is that the displacements in the images can only be learned if they are significantly smaller than the patch, i.e. the displacement should 'fit' inside the patch but also leave enough context for the network to base its estimate on. In end-to-end architectures this restriction does not exist. A possible disadvantage to learning end-to-end is that the amount of data available for training is smaller compared to patch-based methods, which can be trained on many patches extracted from very few images. To compensate for this reduction in available training data, we use elaborate data augmentation techniques.

### C. Training Set Construction

A common issue in image registration is that ground truth transformations generally do not exist for pairs of clinical images. Therefore, we train the network on synthetically deformed clinical images for which we know the deformation field, as we have done in [8], [14], and [19]. In addition to this *learned transformation* $\mathbf{T}_{\text{learned}}$, we use deformable transformations to augment the training data set, which we call the *augmentation transformation* $\mathbf{T}_{\text{augm}}$ in the remainder of the paper. For each iteration of training, an image from the CREATIS data set is selected, to which a random augmentation transformation $\mathbf{T}_{\text{augm}}$ and the combination of a learned and augmentation transformation $\mathbf{T}_{\text{augm}} \circ \mathbf{T}_{\text{learned}}$ are applied. This results in two images $I(\mathbf{T}_{\text{augm}}(\mathbf{x}))$ and $I((\mathbf{T}_{\text{augm}} \circ \mathbf{T}_{\text{learned}})(\mathbf{x}))$, from which the network is trained to estimate the deformation field of $\mathbf{T}_{\text{learned}}$. These images simulate a moving and fixed image of a registration problem respectively, as is shown in Figure 1. The training process can be divided into the following steps:

*a) Augmentation transformations:* To increase the size of the training set we use deformable transformations to make variations on the training set's images. By doing this every iteration, the images in the CREATIS training set that we started with will be spatially transformed thousands of times during training. This aids the generalization of the network to new data. The deformable transformation is defined on a coarse B-spline grid of $2 \times 2 \times 2$ grid points, to which random displacements are assigned from a uniform distribution. These distributions have different ranges for each component of the displacement vectors, which are displayed in Table I. The augmentation transformation is also used to deform masks of the lungs that are used for computing the loss function (see Section III-D).

TABLE I
PARAMETERS FOR THE B-SPLINE TRANSFORMATIONS IN THE TRAINING SET. THE GRID SIZE INDICATES THE NUMBER OF GRID POINTS IN EACH DIMENSION. THE DISPLACEMENTS ARE SAMPLED FROM UNIFORM DISTRIBUTIONS IN THE GIVEN RANGES

| T | Grid size | Grid point displacement ranges (voxels) $u_x$ | $u_y$ | $u_z$ |
|---|---|---|---|---|
| $\mathbf{T}_{\text{augm}}$ | $2 \times 2 \times 2$ | [−3.2, 3.2] | [−6.4, 6.4] | [−12.8, 12.8] |
| $\mathbf{T}^{\text{coarse}}_{\text{learned}}$ | $4 \times 4 \times 4$ | [−3.2, 3.2] | [−6.4, 6.4] | [−12.8, 12.8] |
| $\mathbf{T}^{\text{fine}}_{\text{learned}}$ | $8 \times 8 \times 8$ | [−3.2, 3.2] | [−3.2, 3.2] | [−3.2, 3.2] |

*b) Learned transformations:* To model large transformations such as the inspiration-to-expiration transformation, it is necessary to combine multiple transformations to prevent folding and tearing of the images. Large displacements are modeled on a coarse grid, and smaller displacements on fine grids. Combining them, both the large transformation that moves and scales the lungs in the axial dimensions, as well as the smaller transformations that register the lungs at a finer scale, can be modeled. The learned transformation therefore consists of a concatenation of a course and finer deformation, to form a realistic large deformation: $\mathbf{T}_{\text{learned}} = \mathbf{T}^{\text{coarse}}_{\text{learned}} \circ \mathbf{T}^{\text{fine}}_{\text{learned}}$. Both of these transformations are defined on B-spline grids with $4 \times 4 \times 4$ and $8 \times 8 \times 8$ grid points respectively. The displacements on these grids are again sampled from uniform distributions, with different ranges of displacements (Table I).

*c) Interpolation:* The transformations are applied to the images in such a way that each image is only interpolated once. Hence, for the simulated fixed image

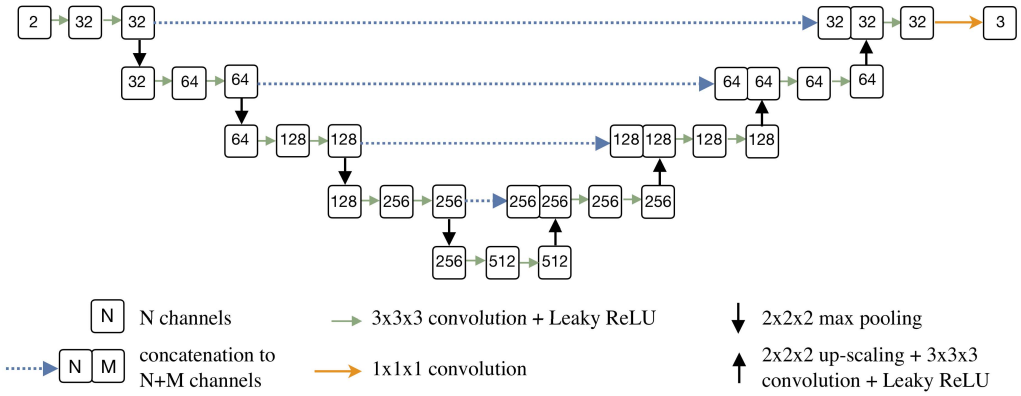


Fig. 2. The network architecture. The network takes two images as input, and outputs three maps: one for each vector field component.

$I((\mathbf{T}_{\text{augm}} \circ \mathbf{T}_{\text{learned}})(\mathbf{x}))$ the transformations are first concatenated, and then applied to the image. For all interpolation operations on images we use third-order B-spline interpolation. For the lung masks, nearest neighbor interpolation is used.

*d) Intensity-based data augmentation:* We also apply a gray-value transformation using the gamma transform $I(\mathbf{x}) \leftarrow I(\mathbf{x})^a$ where $a$ follows a uniform distribution between 0.5 and 1.5, to increase the range of gray-values the network can operate on and aid generalization to new data. These random gamma transforms are applied to both input images independently after interpolation.

### D. Training Procedure

The network is trained by minimizing the average of the $L_1$-norm of the difference between the network's estimate of the vector field $\hat{\mathbf{u}}$ and the true vector field $\mathbf{u}$. Because the relevant part of the transformation only takes place inside the lungs, we use a binary lung segmentation $M$ to mask the lungs during training. Masking the lungs is a common strategy in intensity-based lung registration, where it is used to mask the similarity metric. The augmentation transformation is applied to the mask, such that it masks the relevant region of the image pair. The loss is weighted by the transformed mask $M(\mathbf{T}_{\text{augm}}(\mathbf{x}))$ and defined as

$$L = \frac{\sum_{\mathbf{x}\in\Omega_{\mathbf{F}}} M(\mathbf{T}_{\text{augm}}(\mathbf{x})) \left|\mathbf{u}(\mathbf{x}) - \hat{\mathbf{u}}(\mathbf{x})\right|}{\sum_{\mathbf{x}\in\Omega_{\mathbf{F}}} M(\mathbf{T}_{\text{augm}}(\mathbf{x}))} \tag{1}$$

where $M(\mathbf{x})$ is 0 or 1. The lung masks were the same lung masks that we used for cropping and resizing the images in Section II. Note that the masks are only used to compute the loss during training. At test time the mask is not required for the registration of the lungs. The loss function (1) is optimized using the Adagrad optimizer [20], which decreases the learning rate as a function of training iteration by

$$\eta_{t,i} = \frac{\eta_0}{\sqrt{\sum_{\tau=1}^{t} g_{\tau,i}^2 + \epsilon}} \tag{2}$$

with $\eta_0 = 10^{-2}$, $g_{t,i}$ the gradient for weight $i$ at iteration $t$, and $\epsilon = 10^{-8}$ a constant for numerical stability.

### E. Network Architecture

The end-to-end architecture used in this paper is the fully-convolutional neural network by Ronneberger *et al.* [21] and Çiçek *et al.* [22], and popularized as the U-net architecture which was first used for end-to-end learning of segmentations of 2D and 3D medical images. To adapt the 3D U-net architecture in [22] to the registration problem, the input layer was changed to have two channels, one for the fixed image and one for the moving image (Figure 2). The output layer was changed to have three channels, one for each vector field component. To enable real-valued estimates, the output layer has no activation function. In addition, we deepened the network by adding one more level of pooling, convolutions, and up-sampling. Lastly, all activation functions were changed to leaky rectified linear units parameterized as $\phi(x) = \max(x, 0.01\ x)$, which prevents dying neurons that occurred using the regular rectified linear units (ReLU) [23]. A batch size of one was used to limit the amount of memory required during training. Even with single-instance batches, it can be useful to use a form of batch normalization, which we applied to all convolutional layers. Instead of measuring the mean and standard deviations of layer inputs over a batch, we update the mean and standard deviations using an exponential running average, as proposed by Ioffe and Szegedy [24]. At iteration $t$ the mean and standard deviation are defined by $\bar{\mu}_t = \alpha * \bar{\mu}_{t-1} + (1-\alpha) * \hat{\mu}_{t-1}$ and $\bar{\tau}_t = \alpha * \bar{\tau}_{t-1} + (1-\alpha) * \hat{\tau}_{t-1}$ with $\tau_k = (\sigma_k^2 + \epsilon)^{-1}$, where $\bar{\mu}_t$ and $\bar{\tau}_t$ are the batch mean and inverse of the variance at iteration $t$, $\epsilon = 10^{-4}$ is a constant for numerical stability, and $\alpha$ was set to 0.1. In the current implementation the network accepts images of $128 \times 128 \times 128$, and outputs vector fields of the same size, resulting in about 1.3 billion weights for the entire network.

## IV. Experiments

### A. Elastix Registration Algorithm

We compare the network's performance on the ten DIR-Lab registration pairs with the state-of-the art Elastix image registration software [25]. We objectively determine the performance using the landmark sets included in the DIR-Lab data set. For the Elastix registration we used the Elastix parameters published by Staring *et al.* [26] which was a contender in the

TABLE II
TRE VALUES (mm, $\mu \pm \sigma$) FOR AFFINE REGISTRATION, ELASTIX' REGISTRATION, AND OUR METHOD, FOR EACH OF THE DIR-LAB IMAGES (N = 300 PER IMAGE), WITH COMPARISON TO FIVE EXISTING METHODS. THE LEFT-MOST COLUMNS SHOW TRE VALUES COMPUTED AFTER REGISTRATION OF THE ORIGINAL IMAGES. THE RIGHT-MOST COLUMNS SHOW TRE VALUES COMPUTED AFTER REGISTRATION OF THE CROPPED AND RESIZED IMAGES

| | | Evaluated on original image size | | | | | | Evaluated on $128 \times 128 \times 128$ size | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Set | Before registration | Schmidt-Richberg et al. [28] | Heinrich et al. [29] | Vandemeulebroucke et al. [30] | Delmon et al. [31] | Berendsen et al. [32] | Elastix [26] | Elastix [26] | Elastix without mask [26] | Network trained on CREATIS | Network trained on DIR-Lab[1] |
| 1 | 3.89±2.78 | 1.22±0.64 | 0.97±0.5 | 1.52±0.92 | 1.2±0.6 | 1.00±0.52 | 0.99±0.57 | 1.05±0.53 | 1.04±0.51 | 1.45±1.06 | – |
| 2 | 4.34±3.90 | 1.14±0.65 | 0.96±0.5 | 1.30±1.03 | 1.1±0.6 | 1.02±0.57 | 0.94±0.53 | 1.00±0.56 | 1.20±0.96 | 1.46±0.76 | 1.24±0.61 |
| 3 | 6.94±4.05 | 1.36±0.81 | 1.21±0.7 | 1.69±1.12 | 1.6±0.9 | 1.14±0.89 | 1.13±0.64 | 1.20±0.64 | 1.76±1.49 | 1.57±1.10 | – |
| 4 | 9.83±4.85 | 2.68±2.79 | 1.39±1.0 | 1.82±1.14 | 1.6±1.1 | 1.46±0.96 | 1.49±1.01 | 1.52±1.03 | 1.73±1.57 | 1.95±1.32 | 1.70±1.00 |
| 5 | 7.48±5.50 | 1.57±1.23 | 1.72±1.6 | 2.75±2.45 | 2.0±1.6 | 1.61±1.48 | 1.77±1.53 | 1.42±1.27 | 2.42±2.74 | 2.07±1.59 | – |
| 6 | 10.89±6.96 | 2.21±1.66 | 1.49±1.0 | 2.01±1.16 | 1.7±1.0 | 1.42±0.89 | 1.29±0.85 | 1.47±0.98 | 1.98±1.59 | 3.04±2.73 | – |
| 7 | 11.03±7.42 | 3.81±3.06 | 1.58±1.2 | 2.15±1.59 | 1.9±1.2 | 1.49±1.06 | 1.26±1.09 | 1.43±1.49 | 2.90±3.68 | 3.41±2.75 | – |
| 8 | 14.99±9.00 | 3.42±4.25 | 2.11±2.4 | 2.11±1.79 | 2.2±2.3 | 1.62±1.71 | 1.87±2.57 | 1.42±1.42 | 5.10±7.48 | 2.80±2.46 | – |
| 9 | 7.92±3.97 | 1.83±1.19 | 1.36±0.7 | 2.05±1.20 | 1.6±0.9 | 1.30±0.76 | 1.33±0.98 | 1.42±1.17 | 1.81±1.51 | 2.18±1.24 | 1.61±0.82 |
| 10 | 7.30±6.34 | 2.06±1.92 | 1.43±1.6 | 2.12±1.66 | 1.7±1.2 | 1.50±1.31 | 1.14±0.89 | 1.19±0.76 | 1.79±1.95 | 1.83±1.36 | – |
| All | 8.46±6.58 | 2.13±1.82 | 1.43±1.3 | 1.95±1.47 | 1.66±1.14 | 1.36±1.01 | 1.32±1.24 | 1.31±1.06 | 2.17±3.22 | 2.17±1.89 | 1.52±0.85 |

1) Only values for images not present in the training set shown. Average TRE for all images computed exclusively on these three images.

EMPIRE registration challenge [27]. This algorithm consists of three stages: an affine transformation and two B-spline transformations. The affine stage is both used for the Elastix algorithm that we compare against and for the pre-registration that we apply before applying the network. Consequently, we compare Elastix' B-spline registrations with the network's registration result starting from the affine registration. We give a short description of each stage:

*e) Affine registration:* The affine transformation is optimized by maximizing the normalized correlation similarity metric using adaptive stochastic gradient descent [25]. The optimization is made more robust by using a multi-resolution approach that starts the registration at a lower-resolution version of $\frac{1}{16}$ the size of the original images and runs the optimization consecutively at fractions $\frac{1}{8}$, $\frac{1}{4}$, $\frac{1}{2}$, and 1 of the original resolution. At every resolution the optimization takes 1000 iterations. Every iteration the normalized correlation is computed for 2000 randomly sampled spatial coordinates.

*f) Coarse B-Spline:* The second stage starts from the affine registration's result and uses the same optimization techniques (optimizer, multi-resolution strategy, sampling, and number of iterations) to optimize the normalized correlation. The B-spline transformation is defined on a grid that has an isotropic grid spacing of 80 mm for the first two resolutions and is isotropically scaled down with factors of two for every next resolution to 10 mm.

*g) Finer B-Spline:* The third stage follows the same approach as the second stage, but only measures the metric inside the lung mask of the fixed image. The optimization runs for 2000 iterations at every resolution, and the multi-resolution pyramid in this actions of $\frac{1}{4}$, $\frac{1}{3}$, $\frac{1}{2}$, 1, and 1. The B-spline grid spacing starts from 80 mm and is isotropically scaled down with factors of two for every resolution to 10 mm for the final resolution. The implementation is equal to that of Staring *et al.* [26] with the exception of the sampler, which we changed to Elastix' random sparse mask sampler to reduce the time spent on finding enough samples within the lungs.

For every stage the moving image is interpolated using first-order B-spline interpolation during optimization, and third-order B-spline interpolation to sample the final image. Before registration, the images were resized to the $128 \times 128 \times 128$ input size for fair comparison to the network. Because the network does not require lung masks at test time, we also compared to running the Elastix pipeline described above without using lung masks.

### B. Landmark Errors

The network was trained for 96,700 iterations. During training, the snapshots of the weights were saved every 50 iterations. After training, we averaged the weights in the past 200 snapshots to regularize the final result. The network was applied to the affinely registered images of the DIR-Lab dataset. The affine transformation was applied to the moving (expiration) images, resulting in pairs $I_{\text{insp}}(\mathbf{x})$ and $I_{\text{exp}}(\mathbf{T}_{\text{affine}}(\mathbf{x}))$. The network was applied to this pair of images. The resulting displacement field and affine transformation were applied to the the expiration image, resulting in $I_{\text{exp}}(\mathbf{x} + \hat{\mathbf{u}}(\mathbf{T}_{\text{affine}}(\mathbf{x}) + \mathbf{x}))$. To evaluate we compute the target registration error for every fixed landmark $\mathbf{x}_F$, by finding the associated displacement for that point $\hat{\mathbf{u}}(\mathbf{x}_F)$ and measuring the $L_2$ norm of the difference with the true displacement $\mathbf{x}_M - \mathbf{x}_F$, i.e. $\text{TRE}(\mathbf{x}) = \|\mathbf{x}_F + \hat{\mathbf{u}}(\mathbf{x}_F) - \mathbf{x}_M\|$.

## V. RESULTS

### A. Quantitative Evaluation

The TRE values for our method, Elastix, and five methods from literature that also validate on the DIR-Lab data set [28]–[32] are shown Table II. Note that the methods in [28] and [30]–[32] have been specifically optimized for the sliding motion of the lungs against the ribs, which our method does not model explicitly.

Because our method requires resizing the images, we divide Table II into two parts: results from registrations on the original images are shown on the left, and results from registrations on the cropped and resized images are shown on the right. To show the influence of the resizing and cropping on optimization-based registration methods, we applied Elastix

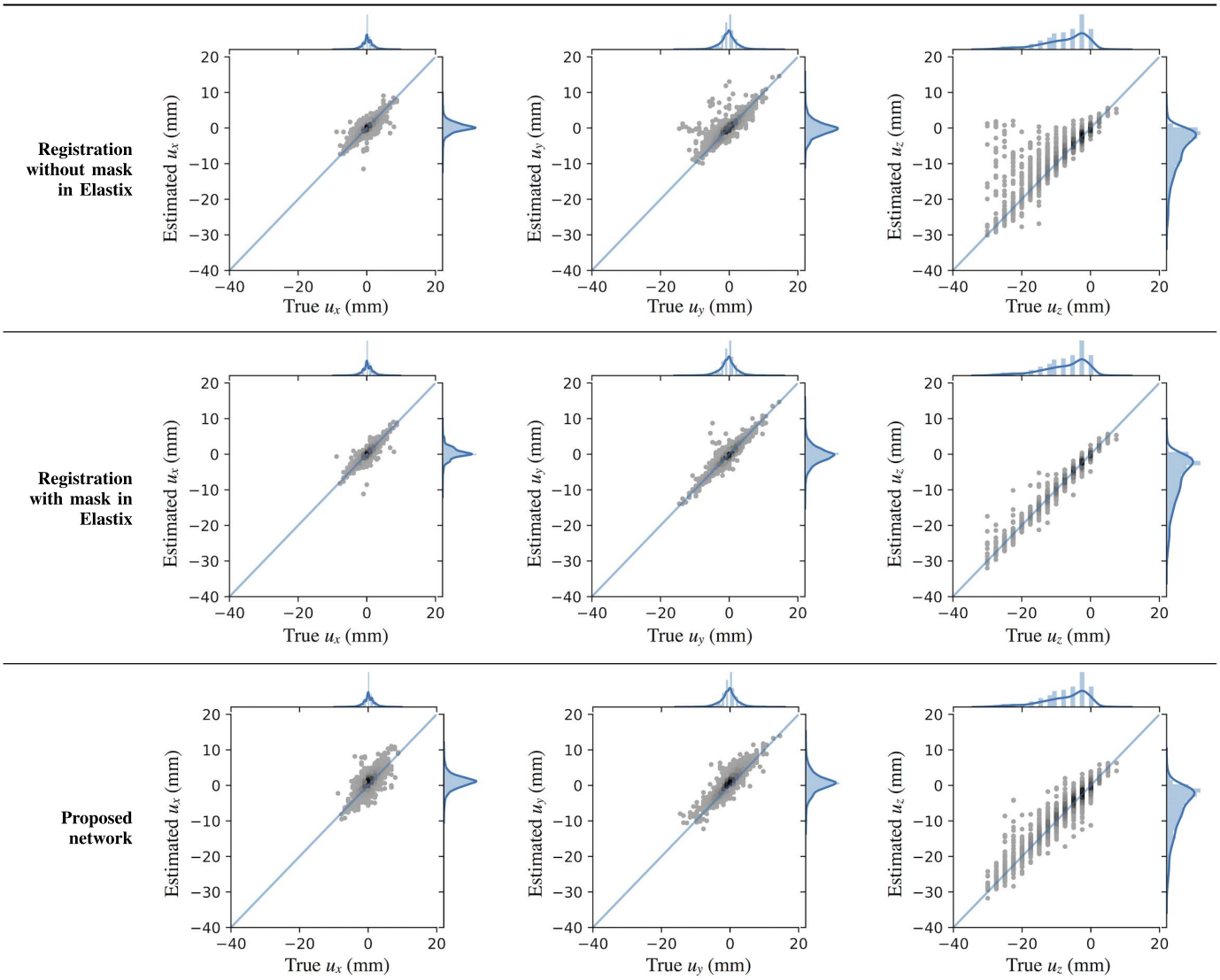


Fig. 3. Correlation plots for the estimated displacement against true displacement between the moving and fixed image for *x*-, *y*-, and *z*-direction for all three methods. Every point is one landmark (N = 3000). Darker colors indicate higher densities of points. The distributions for the true and estimated displacements are shown along the axes.

to both resolutions. In addition, we show the TRE values for running Elastix without the aid of a lung mask, because our method does not require a mask at test time. Our method results in similar TRE values on average ($2.17 \pm 1.89$ mm versus $2.17 \pm 3.22$ mm for Elastix), but the results are more consistent compared to running Elastix without a mask: the standard deviation for Elastix is substantially higher.

Adding the lung mask to Elastix improves the registration result. Resizing and cropping the images has no effect on Elastix' performance: on average the TRE values are similar for the resized and original images ($1.32 \pm 1.24$ mm vs. $1.31 \pm 1.06$ mm).

To show the effect of training on a more similar training set, we also trained the network on seven pairs of images of the DIR-Lab data set, keeping the training set the same size as in the original experiment. We use the other three DIR-Lab pairs (2, 4, and 9) as test set. Because the images in the training and testing sets are now from the same population and the same scanner, a better performance is expected. The results show that the performance improves marginally over training on the CREATIS data set. The average TRE for images 2, 4, and 9 is $1.86 \pm 1.17$ when training on CREATIS, versus $1.52 \pm 0.85$ for training on DIR-Lab.

Figure 3 shows the correlation plots for the displacement in *x*-, *y*-, and *z*-direction. The plots show that both Elastix and our network have high correlations with the ground truth displacements, and that our method has fewer outliers than Elastix without lung masks. Boxplots for the errors for these displacements as well as the TRE are shown in Figure 4.

### B. Registered Images and Folding

We computed the registered images and determined folding by examining values of the deformation field's Jacobian

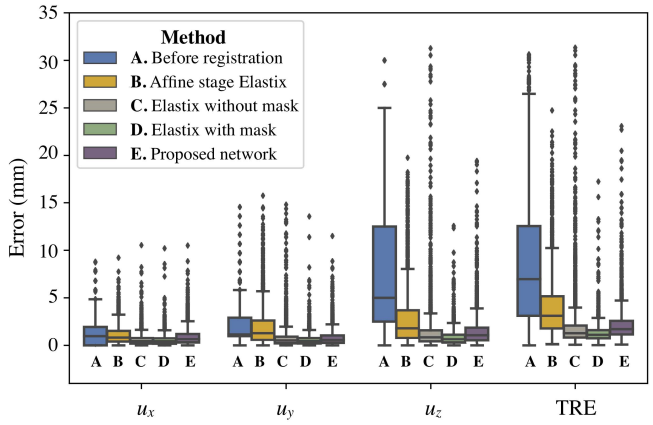


Fig. 4. Boxplots of absolute errors made by the affine registration, the Elastix registration, and our own method for *x*-, *y*-, and *z*-displacement as well as TRE. Diamonds signify outliers.

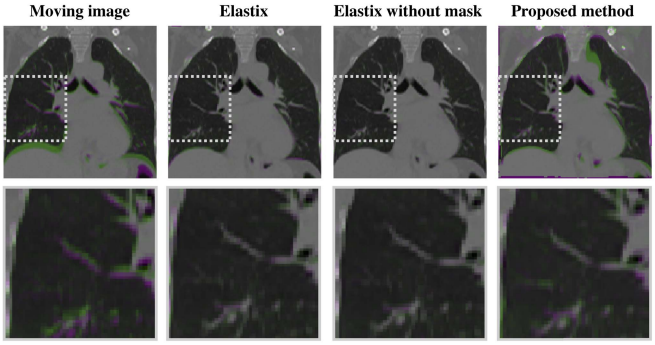


Fig. 5. Example of registration methods in overlap plots, with the (registered) moving image in pink and the fixed image in green. Note that the network only acts on the lung field, and that errors at the lung boundaries are expected.

determinant below zero. Registered images were interpolated using a third-order B-spline grid, with $128 \times 128 \times 128$ grid points. To give an indication of the registration results, we show the overlap between the fixed and registered image for the center frontal slice in overlap plots (Figure 5).

The Jacobian determinant was determined on the same B-spline grid. In these plots, each of the images is color coded: green for the registered image and purple for the fixed image. For 1.28% of voxel positions in the images the deformation field showed folding, measured as those voxels for which the deformation field has a Jacobian determinant smaller than zero. These voxels occur mostly close to the edge of the lungs where there is a transition between the lungs and surrounding tissue.

### C. Runtime

As an indication of speed, we measured the runtime of the network and Elastix for ten registrations. For Elastix, this was measured as the time to start the Elastix command with the two images and the mask as arguments, and obtain the transformation model. For the network we measured the time of one forward pass through the network. Both measurements were run on a system with an Intel Xeon CPU E5-2640 v4, 512 GB of memory and an Nvidia Titan XP graphics card with 12 GB of GPU memory. The neural network was implemented in Lasagne and Theano, and used the Nvidia CUDA and CUDNN toolboxes. For the network this resulted in $0.58 \pm 0.07$ seconds per registration of an image pair, while Elastix took $13.8 \pm 5.7$ minutes (Table III).

TABLE III
AVERAGE COMPUTATION TIMES ($\mu \pm \sigma$) FOR METHODS IN LITERATURE (WHEN REPORTED) AND OUR METHOD

| | Evaluated on original image size | | Evaluated on $128 \times 128 \times 128$ size | |
|---|---|---|---|---|
| | **Heinrich et al. [29]** | **Delmon et al. [31]** | **Elastix [26]** | **Proposed** |
| Total registration | 7.97 *min.* | 58 *min.* | – | – |
| Affine part | – | – | 2.7±1.1 *min.* | 2.7±1.1 *min.* |
| Deformable part | – | – | 13.8±5.7 *min.* | 0.58±0.07 *sec.* |

## VI. DISCUSSION

In this paper we have proposed a supervised registration algorithm based on a convolutional neural network, which learns from artificial deformations of a small set of images. No manual labeling of the deformation is required for training the network, as synthetic geometric transformations of the training images are generated during training. The network's end-to-end architecture allows the estimation of a deformable transformation for the full image domain, with a displacement vector for every voxel, in less than a second. This is substantially faster than comparable algorithms, which require many minutes.

We have validated this approach on pulmonary CT scans, and show that the network can perform accurate registration between expiration and inspiration images, given an affine pre-registration of the images. We show that a network trained on a small set of CT scans can generalize to a set of pulmonary CT images of a different patient group, acquired on a different scanner setup. Furthermore, this shows that the network can generalize from artificial transformations in the training to real deformations of the lungs at test time. Training on more similar images (i.e. same scanner, hospital, but different patient) can improve the results (Table II).

Based on the TRE values in Table II, the proposed method performs similar to the Elastix algorithm ($2.17 \pm 1.89$ mm versus $2.17 \pm 3.22$ mm for Elastix without mask). Without using lung masks, Elastix has a much higher standard deviation in the TRE values compared to our method, indicating a less consistent result than our method. When comparing the TRE values for our method with existing methods that explicitly model sliding motion [28], [30]–[32] we see that two of the existing methods perform better on average, but our method results in very similar TRE values to those of Schmidt-Richberg *et al.* [28]. It should be noted that the computation times for the methods (when they are reported) are slower than our method (Table III). Four of the methods [28], [31], [32] also require a lung mask, while our method only requires crude lung masks for the training set but does not require a lung mask at test time.

The current methodology has two benefits over optimization-based registration methods. First, it enables

near-instantaneous estimation of the deformation at test time, having done any time consuming optimization during training. This is an advantage that is clinically relevant, for example in radiation treatment when fast registration is necessary to register a pre-operative planning scan to an inter- or intra-fractional scan. Conventional methods usually take minutes to optimize a deformable transformation. Second, most registration algorithms require elaborate parameterization that is application specific, that are not required in the current methodology because the network learns from the images directly.

No expert-annotated data is required to train the network, because the training methodology uses artificial transformation applied to a small representative set of images to create the training set. This also means that the network can directly learn relevant image features for registration, requiring no choice for feature selection or similarity metric. Features that are useful for the estimation of the transformation are learned implicitly from the data. The fact that the network is not trained to optimize similarity between images means it cannot get stuck in local minima of a similarity metric. In similarity-based methods this can occur with repeating structures in the images, a common example being the rib cage. It is important to note that the network can generalize from the training set (the CREATIS set) to a completely separate testing set (the DIR-Lab set) created in a different hospital, using a different scanner. The experiment in which we trained on seven pairs of DIR-Lab images, and tested on the other three, shows only a small improvement in TRE compared to training on the CREATIS data set, which is further indication that the network can generalize well to other data.

Another important point is that realistic transformations need to be generated for training the network. This is potentially the part of the methodology requiring most attention when applying the methodology to other kinds of images. In lung registration, this is especially important because the deformation is complex. To model the lung deformation, a synthetic deformable transformation on a fine grid is required, but generating random displacements that are large enough will lead to unrealistic transformations. In this paper, a sequence of transformations was used because this provides the opportunity to construct transformations at a fine scale (a fine grid spacing) without improper (tearing and folding) transformations. The ranges of the transformations were chosen to be close to the expected displacements in the lungs, which are larger in the out-of-plane (superior to inferior) direction, compared to the in-plane displacements. Ideally, this kind of training set construction would require very little parameterization, but we found that a much better result can be obtained if these are chosen such that they reflect reality. In our experience, the composition of multiple transformations when synthesizing data contributes significantly to learning the complex deformation of the lungs compared to simpler transformations.

The network is trained on well-posed B-spline transformations, which means that the resulting deformation fields are relatively well-posed as well, without any additional constraint during training. The deformation fields did display folding close to the edge of the lungs, which is to be expected in a method that is optimized only inside the lungs. A point for future work is the inclusion of such a regularization constraint in the network's loss function, for example a bending energy penalty computed on the deformation field, or a regularization term that explicitly penalizes any sub-zero values of the deformation's Jacobian determinant.

The network's U-net architecture enables end-to-end estimation of the displacement field. The choice for this architecture stems from the fact that it allows many layers in the network while at the same time being memory efficient: the pooling layers give lower-resolution representations of the same information which saves a significant amount of memory. The pooling layers also force the network to use multi-resolution information during optimization. For the current implementation we chose an input-size of $128 \times 128 \times 128$ voxels, balancing the required computational resources and registration accuracy. Besides improvements in GPU hardware, options to solve this problem include scaling the images to a larger size closer to the original size and training and running the network on multiple patches at the expense of computational cost. Another disadvantage of patch-based methods is that the patches need to be large enough to capture the range of displacements, while leaving enough contextual information to estimate the displacements [14]. In addition, patch-based systems only use local information, which means that any anatomical or positional information is lost, while the proposed architecture is trained on a specific anatomy, namely the lungs, and can optimize parts of the network for specific anatomical features.

In this paper we have have applied the proposed method to pulmonary CT images. However, we consider this a generic method for supervised deformable image registration of monomodal image. The method can be adapted to other data by retraining the network. In the current implementation the network estimates the deformable part of the transformation, requiring an affine pre-registration to be able to perform well. In experiments we found the network unable to estimate large displacements, corresponding to those found in the affine stage. In most lung registration algorithms a multi-resolution strategy is applied, and in fact the Elastix algorithm that we have used for comparison uses both a multi-resolution strategy as well as a multi-stage approach in which affine and B-spline components are combined. A topic for future research is to explore a fully trainable registration algorithm that eliminates the need for pre-registration altogether, by using multiple stages or resolutions.

## VII. Conclusion

A supervised algorithm for pulmonary CT image registration has been presented. The algorithm is based on a convolutional neural network that is trained to estimate deformable transformations from a very small set of images that are artificially deformed. The artificial deformations that are applied to the images use transformations at multiple scales, which in our experience is crucial for successful learning of realistic transformations. Combined with affine pre-registration, the algorithm is shown accurately register images in the DIR-Lab data set of pulmonary CT images, even when trained

on a deviating set of pulmonary CT images. Major advantages of this algorithm are the very short time required to estimate the deformable transformation, and the fact that at test time no manual optimization of registration parameters is required.

## REFERENCES


[1] J. B. A. Maintz and M. A. Viergever, "A survey of medical image registration," *Med. Image Anal.*, vol. 2, no. 1, pp. 1–36, Mar. 1998.

[2] H. Greenspan, B. V. Ginneken, and R. M. Summers, "Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1153–1159, Mar. 2016.

[3] M. Simonovsky, B. Gutiérrez-Becker, D. Mateus, N. Navab, and N. Komodakis, "A deep metric for multimodal registration," in *Proc. MICCAI*, vol. 3, 2016, pp. 10–18.

[4] G. Wu, M. Kim, Q. Wang, B. C. Munsell, and D. Shen, "Scalable high-performance image registration framework by unsupervised deep feature representations learning," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 7, pp. 1505–1516, Jul. 2016.

[5] B. Gutiérrez-Becker, D. Mateus, L. Peter, and N. Navab, "Guiding multimodal registration with learned optimization updates," *Med. Image Anal.*, vol. 41, pp. 2–17, Oct. 2017.

[6] X. Yang, R. Kwitt, M. Styner, and M. Niethammer, "QuickSilver: Fast predictive image registration—A deep learning approach," *NeuroImage*, vol. 158, pp. 378–396, Sep. 2017.

[7] A. Dosovitskiy *et al.*, "FlowNet: Learning optical flow with convolutional networks," in *Proc. ICCV*, 2015, pp. 2758–2766.

[8] K. A. J. Eppenhof and J. P. W. Pluim, "Supervised local error estimation for nonlinear image registration using convolutional neural networks," *Proc. SPIE*, vol. 10133, pp. 101331U-1–101331U-6, Feb. 2017.

[9] H. Sokooti, B. D. de Vos, F. F. Berendsen, B. P. F. Lelieveldt, I. Išgum, and M. Staring, "Nonrigid image registration using multi-scale 3D convolutional neural networks," in *Proc. MICCAI*, vol. 1, 2017, pp. 232–239.

[10] X. Cao *et al.*, "Deformable image registration based on similarity-steered CNN regression," in *Proc. MICCAI*, vol. 1, 2017, pp. 300–308.

[11] M.-M. Rohé, M. Datar, T. Heimann, M. Sermesant, and X. Pennec, "SVF-Net: Learning deformable image registration using shape matching," in *Proc. MICCAI*, vol. 1, 2017, pp. 266–274.

[12] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Proc. NIPS*, 2015, pp. 2017–2025.

[13] B. D. de Vos, F. F. Berendsen, M. A. Viergever, M. Staring, and I. Išgum, "End-to-end unsupervised deformable image registration with a convolutional neural network," in *Proc. DLMIA Workshop Held Conjunct (MICCAI)*, 2017, pp. 204–212.

[14] K. A. J. Eppenhof and J. P. W. Pluim, "Error estimation of deformable image registration of pulmonary CT scans using convolutional neural networks," *J. Med. Imag.*, vol. 5, no. 2, p. 024003, 2018.

[15] J. Vandemeulebroucke, D. Sarrut, and P. Clarysse, "The POPI-model, a point-validated pixel-based breathing thorax model," in *Proc. ICCR*, 2007, pp. 195–199.

[16] J. Vandemeulebroucke, S. Rit, J. Kybic, P. Clarysse, and D. Sarrut, "Spatiotemporal motion estimation for respiratory-correlated imaging of the lungs," *Med. Phys.*, vol. 38, no. 1, pp. 166–178, 2011.

[17] R. Castillo *et al.*, "A framework for evaluation of deformable image registration spatial accuracy using large landmark point sets," *Phys. Med. Biol.*, vol. 54, no. 7, p. 1849, 2009.

[18] E. Castillo, R. Castillo, J. Martinez, M. Shenoy, and T. Guerrero, "Four-dimensional deformable image registration using trajectory modeling," *Phys. Med. Biol.*, vol. 55, no. 1, p. 305, 2010.

[19] K. A. J. Eppenhof, M. W. Lafarge, P. Moeskops, M. Veta, and J. P. W. Pluim, "Deformable image registration using convolutional neural networks," *Proc. SPIE*, vol. 10574, pp. 105740S-1–105740S-6, Mar. 2018.

[20] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, Feb. 2011.

[21] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, vol. 3, 2015, pp. 234–241.

[22] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: Learning dense volumetric segmentation from sparse annotation," in *Proc. MICCAI*, vol. 2, 2016, pp. 424–432.

[23] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. ICML*, 2013, pp. 1–3.

[24] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. ICML*, 2015, pp. 448–456.

[25] S. Klein, M. Staring, K. Murphy, M. A. Viergever, and J. P. W. Pluim, "elastix: A toolbox for intensity-based medical image registration," *IEEE Trans. Med. Imag.*, vol. 29, no. 1, pp. 196–205, Jan. 2010.

[26] M. Staring, S. Klein, J. H. C. Reiber, W. Niessen, and B. Stoel, "Pulmonary image registration with elastix using a standard intensity-based algorithm," in *Proc. Workshop Med. Image Anal. Clinic Grand Challenge (MICCAI)*, 2010, pp. 73–79.

[27] K. Murphy *et al.*, "Evaluation of registration methods on thoracic CT: The EMPIRE10 challenge," *IEEE Trans. Med. Imag.*, vol. 30, no. 11, pp. 1901–1920, Nov. 2011.

[28] A. Schmidt-Richberg, R. Werner, H. Handels, and J. Ehrhardt, "Estimation of slipping organ motion by registration with direction-dependent regularization," *Med. Image Anal.*, vol. 16, no. 1, pp. 150–159, 2012.

[29] M. P. Heinrich, M. Jenkinson, M. Brady, and J. A. Schnabel, "MRF-based deformable registration and ventilation estimation of lung CT," *IEEE Trans. Med. Imag.*, vol. 32, no. 7, pp. 1239–1248, Jul. 2013.

[30] J. Vandemeulebroucke, O. Bernard, S. Rit, J. Kybic, P. Clarysse, and D. Sarrut, "Automated segmentation of a motion mask to preserve sliding motion in deformable registration of thoracic CT," *Med. Phys.*, vol. 39, no. 2, pp. 1006–1015, 2012.

[31] V. Delmon, S. Rit, R. Pinho, and D. Sarrut, "Registration of sliding objects using direction dependent B-splines decomposition," *Phys. Med. Biol.*, vol. 58, no. 5, pp. 1303–1314, 2013.

[32] F. F. Berendsen, A. N. T. J. Kotte, M. A. Viergever, and J. P. W. Pluim, "Registration of organs with sliding interfaces and changing topologies," *Proc. SPIE*, vol. 9034, pp. 90340E-1–90340E-7, Mar. 2014.