Journal of Zhejiang University-SCIENCE C (Computers & Electronics)
ISSN 1869-1951 (Print); ISSN 1869-196X (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Multi-affine registration using local polynomial expansion*

Yuan-jun WANG[†1,2], Gunnar FARNEBÄCK[2], Carl-Fredrik WESTIN[†‡2]

([1]*Digital Medical Research Center, Shanghai Medical School, Fudan University, Shanghai 200032, China*)

([2]*Lab of Mathematics in Imaging, Brigham and Women's Hospital, Harvard Medical School, Boston 02115, MA, USA*)

[†]E-mail: yjwang@bwh.harvard.edu; westin@bwh.harvard.edu

Received Oct. 30, 2009; Revision accepted May 28, 2010; Crosschecked June 7, 2010

Abstract: In this paper, we present a non-linear (multi-affine) registration algorithm based on a local polynomial expansion model. We generalize previous work using a quadratic polynomial expansion model. Local affine models are estimated using this generalized model analytically and iteratively, and combined to a deformable registration algorithm. Experiments show that the affine parameter calculations derived from this quadratic model are more accurate than using a linear model. Experiments further indicate that the multi-affine deformable registration method can handle complex non-linear deformation fields necessary for deformable registration, and a faster convergent rate is verified from our comparison experiment.

Key words: Deformable registration, Polynomial expansion, Least squares, Multi-affine, Normalized convolution
doi:10.1631/jzus.C0910658 **Document code:** A **CLC number:** TP391.4

1 Introduction

Image registration is an important step in medical image analysis enabling comparisons of the same brain region across a population of subjects. In the work presented in this paper, we extend an affine registration method based on a polynomial expansion (Farnebäck and Westin, 2006) to a multi-affine model, thus creating a method that can perform deformable (non-linear) registration. An advantage of using a technique that is locally affine, compared to a more high-dimensional approach, is that it restricts the dimensionality of possible deformation locally, reducing the potential of creating unwanted non-realistic deformations. More details concerning medical image registration techniques can be found in Antoine Maintz and Viergever (1998), Hill *et al.* (2001), and Holden (2008).

A well-known non-linear registration model is the Demons' algorithm developed by Thirion (1998) and Pennec *et al.* (1999), in which a multi-scale and iteration process is used to achieve a good quality, and here we have also used these techniques. A popular metric for image registration is mutual information (Wells *et al.*, 1996; Pluim *et al.*, 2004). Mutual information has the advantage of being able to handle registration of different modalities. In addition, normalized cross-correlation metric has also been applied to both affine and deformable registration problems (Yuan *et al.*, 2006; Andronache *et al.*, 2008; Gan *et al.*, 2008; Schnabel *et al.*, 2008; Rehman *et al.*, 2009).

In this paper, we use an alternative approach and generalize previous work on registration based on polynomial expansion using a quadratic model. The idea of local polynomial expansion is that each image neighborhood can be approximated locally with a set of low-dimensional polynomial functions.

It turns out that local displacement between a source and a target image data can be analytically encoded by the polynomial coefficients (Farnebäck and Westin, 2006). An advantage of this analytic approach, compared to a numerical method, such as gradient-based optimization to find the transla-

‡ Corresponding author

* Project supported by the joint PhD Program of the China Scholarship Council (CSC), the US National Institutes of Health (NIH) (Nos. R01MH074794 and P41RR013218), and the National Natural Science Foundation of China (No. 60972102)

tion field, is that the convergence is in general much faster with an analytic approach. This paper also extends the affine method to a non-linear method using a multi-affine approach, where spatially localized affine models are estimated and combined in order to handle the more complex non-linear deformation fields necessary for deformable registration.

2 Local polynomial expansion

The idea behind polynomial expansion is to approximate the image's intensity in a neighborhood of each voxel with a set of polynomial functions. Westin (1994), Farnebäck (2002), and Farnebäck and Westin (2006) proposed a framework of polynomial expansion for registration, and they used a linear and quadratic polynomial expansion model for image registration:

$$f(\boldsymbol{x}) \sim \boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}^{\mathrm{T}}\boldsymbol{x} + c, \tag{1}$$

where $\boldsymbol{A}$ is a symmetric matrix, $\boldsymbol{b}$ is a vector, and c is a scalar.

This expansion can also be written using a matrix $\boldsymbol{B}$ containing the basis functions. For simplicity, we will consider only quadratic polynomials. The number of basis functions required for quadratic polynomials depends on the dimensionality of the signal:

$$\begin{aligned} 6 \quad &\text{in 2D}: \quad 1, x, y, x^2, y^2, xy; \\ 10 \quad &\text{in 3D}: \quad 1, x, y, z, x^2, y^2, z^2, xy, xz, yz. \end{aligned}$$

In 2D, then

$$\boldsymbol{B} = \begin{pmatrix} | & | & | & | & | & | \\ 1 & x & y & x^2 & y^2 & xy \\ | & | & | & | & | & | \end{pmatrix},$$

and

$$f(\boldsymbol{x}) \sim \boldsymbol{r}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{r},$$

where the vector $\boldsymbol{r} = [r_1, r_2, ..., r_6]^{\mathrm{T}}$ contains the coordinates corresponding to the respective basis functions in $\boldsymbol{B}$. These quadratic polynomial expansion coordinates (also called coefficients) can be estimated using a weighted least squares method to solve for the polynomial coefficients in each image neighborhood (Westin, 1994; Farnebäck, 2002),

$$\boldsymbol{r}_{\mathrm{opt}} = \arg\min_{\boldsymbol{r}} \sum_{\boldsymbol{x}} w(\boldsymbol{x}) ||\boldsymbol{B}(\boldsymbol{x})\boldsymbol{r} - f(\boldsymbol{x})||^2, \tag{2}$$

where w is a Gaussian spatial weighting function.

3 Translation estimation

Following the description in Farnebäck and Westin (2006), to derive the expression for the translation, we first consider a polynomial expansion of the signal, f_{fixed}, and a version of the signal that has been globally translated, f_{moving}, by the vector $\boldsymbol{d}$.

3.1 Translation estimation from first-order polynomial expansion

We start with the simpler first-order polynomial (linear) expansion model, and then in the next section we will derive the expression for the quadratic case.

With the linear polynomial expansion model, the target f_{fixed} and source images f_{moving} can be locally described by

$$\begin{aligned} f_{\mathrm{fixed}}(\boldsymbol{x}) &\sim \boldsymbol{b}_{\mathrm{f}}^{\mathrm{T}}\boldsymbol{x} + c_{\mathrm{f}}, \\ f_{\mathrm{moving}}(\boldsymbol{x}) &\sim \boldsymbol{b}_{\mathrm{m}}^{\mathrm{T}}\boldsymbol{x} + c_{\mathrm{m}}. \end{aligned}$$

Suppose that $f_{\mathrm{moving}}(\boldsymbol{x})$ has been translated by the vector $\boldsymbol{d}$ in relation to the fixed image, i.e., $f_{\mathrm{moving}}(\boldsymbol{x}) = f_{\mathrm{fixed}}(\boldsymbol{x} - \boldsymbol{d})$. Inserting the expressions for the local expansion then gives

$$\begin{aligned} f_{\mathrm{moving}}(\boldsymbol{x}) &= f_{\mathrm{fixed}}(\boldsymbol{x} - \boldsymbol{d}) = \boldsymbol{b}_{\mathrm{f}}^{\mathrm{T}}(\boldsymbol{x} - \boldsymbol{d}) + c_{\mathrm{f}} \\ &= \boldsymbol{b}_{\mathrm{f}}^{\mathrm{T}}\boldsymbol{x} + c_{\mathrm{f}} - \boldsymbol{b}_{\mathrm{f}}^{\mathrm{T}}\boldsymbol{d} = \boldsymbol{b}_{\mathrm{m}}^{\mathrm{T}}\boldsymbol{x} + c_{\mathrm{m}}. \end{aligned}$$

Equating the coefficients in linear polynomial yields:

$$\begin{cases} \boldsymbol{b}_{\mathrm{m}} = \boldsymbol{b}_{\mathrm{f}}, \\ c_{\mathrm{m}} = c_{\mathrm{f}} - \boldsymbol{b}_{\mathrm{f}}^{\mathrm{T}}\boldsymbol{d}. \end{cases} \tag{3}$$

Inserting $\triangle c(\boldsymbol{x}) = c_{\mathrm{f}}(\boldsymbol{x}) - c_{\mathrm{m}}(\boldsymbol{x})$, $\boldsymbol{b}(\boldsymbol{x}) = (\boldsymbol{b}_{\mathrm{f}}(\boldsymbol{x}) + \boldsymbol{b}_{\mathrm{m}}(\boldsymbol{x}))/2$ in Eq. (3) gives

$$\boldsymbol{b}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{d} = \triangle c(\boldsymbol{x}). \tag{4}$$

This gives an expression of the local displacement $\boldsymbol{d}$ for a neighborhood. Combining the local constraints over the image, a global displacement $\boldsymbol{d}$ can be estimated by minimizing the squared error:

$$\boldsymbol{d}_{\mathrm{opt}} = \arg\min_{\boldsymbol{d}} \sum_{\boldsymbol{x}} ||\boldsymbol{b}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{d} - \triangle c(\boldsymbol{x})||^2. \tag{5}$$

Solving for $\boldsymbol{d}_{\mathrm{opt}}$ then gives

$$\boldsymbol{d}_{\mathrm{opt}} = \Big(\sum \boldsymbol{b}(\boldsymbol{x})\boldsymbol{b}(\boldsymbol{x})^{\mathrm{T}}\Big)^{-1} \sum \Big(\boldsymbol{b}(\boldsymbol{x}) \triangle c(\boldsymbol{x})\Big). \tag{6}$$

3.2 Translation estimation from second-order polynomial expansion

For the second-order (quadratic) polynomial expansion model we can write the fixed and translated signal as

$$f_{\text{fixed}}(\boldsymbol{x}) = \boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{f}}\boldsymbol{x} + \boldsymbol{b}_{\mathrm{f}}^{\mathrm{T}}\boldsymbol{x} + c_{\mathrm{f}}, \tag{7}$$

and construct a new signal f_{moving} by a global translation $\boldsymbol{d}$:

$$\begin{aligned} f_{\text{moving}}(\boldsymbol{x}) &= f_{\text{fixed}}(\boldsymbol{x}-\boldsymbol{d}) \\ &= (\boldsymbol{x}-\boldsymbol{d})^{\mathrm{T}}\boldsymbol{A}_{\mathrm{f}}(\boldsymbol{x}-\boldsymbol{d}) + \boldsymbol{b}_{\mathrm{f}}^{\mathrm{T}}(\boldsymbol{x}-\boldsymbol{d}) + c_{\mathrm{f}} \\ &= \boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{f}}\boldsymbol{x} + (\boldsymbol{b}_{\mathrm{f}} - 2\boldsymbol{A}_{\mathrm{f}}\boldsymbol{d})^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{d}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{f}}\boldsymbol{d} \\ &\quad - \boldsymbol{b}_{\mathrm{f}}^{\mathrm{T}}\boldsymbol{d} + c_{\mathrm{f}} \\ &= \boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{m}}\boldsymbol{x} + \boldsymbol{b}_{\mathrm{m}}^{\mathrm{T}}\boldsymbol{x} + c_{\mathrm{m}}. \end{aligned} \tag{8}$$

Equating the quadratic, linear, and constant coefficients yields:

$$\boldsymbol{A}_{\mathrm{m}} = \boldsymbol{A}_{\mathrm{f}}, \tag{9}$$

$$\boldsymbol{b}_{\mathrm{m}} = \boldsymbol{b}_{\mathrm{f}} - 2\boldsymbol{A}_{\mathrm{f}}\boldsymbol{d}, \tag{10}$$

$$c_{\mathrm{m}} = \boldsymbol{d}^{\mathrm{T}}\boldsymbol{A}_{\mathrm{f}}\boldsymbol{d} - \boldsymbol{b}_{\mathrm{f}}^{\mathrm{T}}\boldsymbol{d} + c_{\mathrm{f}}. \tag{11}$$

Farnebäck and Westin (2006) used Eqs. (9) and (10) to derive an expression for the displacement. By setting $\boldsymbol{A}(\boldsymbol{x}) = (\boldsymbol{A}_{\mathrm{m}}(\boldsymbol{x}) + \boldsymbol{A}_{\mathrm{f}}(\boldsymbol{x}))/2$, and $\Delta\boldsymbol{b}(\boldsymbol{x}) = (\boldsymbol{b}_{\mathrm{f}}(\boldsymbol{x}) - \boldsymbol{b}_{\mathrm{m}}(\boldsymbol{x}))/2$,

$$\boldsymbol{A}(\boldsymbol{x})\boldsymbol{d} = \Delta\boldsymbol{b}(\boldsymbol{x}), \tag{12}$$

and $\boldsymbol{d}$ was computed by minimizing the squared error with the constraints over the whole image,

$$\boldsymbol{d}_{\text{opt}} = \arg\min_{\boldsymbol{d}} \sum_{\boldsymbol{x}} \|\boldsymbol{A}(\boldsymbol{x})\boldsymbol{d} - \Delta\boldsymbol{b}(\boldsymbol{x})\|^2, \tag{13}$$

with the least squares solution for the displacement $\boldsymbol{d}_{\text{opt}}$,

$$\boldsymbol{d}_{\text{opt}} = \Big(\sum \boldsymbol{A}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{A}(\boldsymbol{x})\Big)^{-1} \sum \Big(\boldsymbol{A}(\boldsymbol{x})^{\mathrm{T}}\Delta\boldsymbol{b}(\boldsymbol{x})\Big). \tag{14}$$

In this paper we wish to investigate the estimation of $\boldsymbol{d}$ by incorporating the information contained in the additional constraint available by Eq. (11). It is difficult to directly derive an expression of $\boldsymbol{d}$ from this third constraint. However, by substituting the whole Eq. (10) into Eq. (11), we obtain

$$\begin{aligned} c_{\mathrm{m}} &= \boldsymbol{d}^{\mathrm{T}}(\boldsymbol{b}_{\mathrm{f}} - \boldsymbol{b}_{\mathrm{m}})/2 - \boldsymbol{b}_{\mathrm{f}}^{\mathrm{T}}\boldsymbol{d} + c_{\mathrm{f}} \\ &= (\boldsymbol{b}_{\mathrm{f}} - \boldsymbol{b}_{\mathrm{m}})^{\mathrm{T}}\boldsymbol{d}/2 - \boldsymbol{b}_{\mathrm{f}}^{\mathrm{T}}\boldsymbol{d} + c_{\mathrm{f}} \\ &= -(\boldsymbol{b}_{\mathrm{f}} + \boldsymbol{b}_{\mathrm{m}})^{\mathrm{T}}\boldsymbol{d}/2 + c_{\mathrm{f}}. \end{aligned}$$

By inserting $\triangle c(\boldsymbol{x}) = c_{\mathrm{f}}(\boldsymbol{x}) - c_{\mathrm{m}}(\boldsymbol{x})$ and $\boldsymbol{b}(\boldsymbol{x}) = (\boldsymbol{b}_{\mathrm{f}}(\boldsymbol{x}) + \boldsymbol{b}_{\mathrm{m}}(\boldsymbol{x}))/2$, we obtain again the constraints for the displacement field:

$$\boldsymbol{b}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{d}(\boldsymbol{x}) = \triangle c(\boldsymbol{x}), \tag{15}$$

with the least squares solution for $\boldsymbol{d}$,

$$\boldsymbol{d} = \Big(\sum \boldsymbol{b}(\boldsymbol{x})\boldsymbol{b}(\boldsymbol{x})^{\mathrm{T}}\Big)^{-1} \sum \Big(\boldsymbol{b}(\boldsymbol{x})\triangle c(\boldsymbol{x})\Big). \tag{16}$$

It is interesting to see that the third constraint (Eq. (11)) generates an expression for estimating the displacement vector $\boldsymbol{d}$ identical to the one obtained from the derivation using the first-order approximation (Eq. (6)). It should be noted, however, that when increasing the model from order one to two, the added second-order basis functions are orthogonal to the first-order ones, but not to the constant (0th order) basis function. This means that the first-order term $\boldsymbol{b}$ will be estimated in the same way by linear and quadratic models, but the constant term c, the coefficient for the 0th order basis function, will not.

In Farnebäck and Westin (2006), the estimation of the displacement vector $\boldsymbol{d}$ from the linear and the quadratic models were combined (Eqs. (4) and (14)):

$$\begin{aligned} \epsilon^2 = \sum_{\boldsymbol{x}} \Big(&\beta_1 ||\boldsymbol{A}(\boldsymbol{x})\boldsymbol{d} - \triangle\boldsymbol{b}(x)||^2 \\ &+\beta_2 ||\boldsymbol{b}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{d} - \triangle c(\boldsymbol{x})||^2\Big), \end{aligned} \tag{17}$$

$$\begin{aligned} \boldsymbol{G} &= \sum(\beta_1 \boldsymbol{A}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{A}(\boldsymbol{x}) + \beta_2 \boldsymbol{b}(\boldsymbol{x})\boldsymbol{b}(\boldsymbol{x})^{\mathrm{T}}), \\ \boldsymbol{h} &= \sum(\beta_1 \boldsymbol{A}(\boldsymbol{x})^{\mathrm{T}}\triangle\boldsymbol{b}(\boldsymbol{x}) + \beta_2 \boldsymbol{b}(\boldsymbol{x})\triangle c(\boldsymbol{x})), \\ \boldsymbol{d} &= \boldsymbol{G}^{-1}\boldsymbol{h}. \end{aligned} \tag{18}$$

In this work we combine the two estimated displacement vectors derived directly from the second-order model (Eqs. (14) and (16)). It is interesting to see that the expressions are identical, except for

the fact that $\boldsymbol{b}$ and c are derived from different models. However, from the fact that $\boldsymbol{b}$ is the same in linear and quadratic models, the only difference in the expressions will be in the constant factor c. As we will see in the experiments, using the c from the quadratic model yields better registration results.

4 Affine registration based on local polynomial expansion

4.1 Estimation of affine registration parameters

Suppose the displacement of $\boldsymbol{x}$ is $\boldsymbol{d}(\boldsymbol{x}) = (d_1(\boldsymbol{x}), d_2(\boldsymbol{x}))^{\mathrm{T}}$, where $d_1(\boldsymbol{x}), d_2(\boldsymbol{x})$ are the components of $\boldsymbol{d}(\boldsymbol{x})$ in x, y directions. As for affine transform, it can be represented with six parameters:

$$\begin{aligned} d_1(\boldsymbol{x}) &= a_1 + a_2 x + a_3 y, \\ d_2(\boldsymbol{x}) &= a_4 + a_5 x + a_6 y. \end{aligned}$$

We define the basis function as $\boldsymbol{S}(\boldsymbol{x})$ and affine parameters as $\boldsymbol{P}$ corresponding to pixel $\boldsymbol{x}$. We have equation $\boldsymbol{d}(\boldsymbol{x}) = \boldsymbol{S}(\boldsymbol{x})\boldsymbol{P}$. Substituting this into Eq. (17), using the least squares method again, we obtain the solution to estimate affine registration parameters $\boldsymbol{P}$ for the whole image:

$$\boldsymbol{S}(\boldsymbol{x}) = \begin{pmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{pmatrix},$$

$$\boldsymbol{P} = [a_1, a_2, a_3, a_4, a_5, a_6]^{\mathrm{T}},$$

$$\begin{aligned} \epsilon^2 &= \sum_{\boldsymbol{x}} (\beta_1 ||\boldsymbol{A}(\boldsymbol{x})\boldsymbol{S}(\boldsymbol{x})\boldsymbol{P} - \triangle\boldsymbol{b}(\boldsymbol{x})||^2 \\ &\quad + \beta_2 ||\boldsymbol{b}(\boldsymbol{x})\boldsymbol{S}(\boldsymbol{x})\boldsymbol{P} - \triangle c(\boldsymbol{x})||^2), \\ \boldsymbol{G}' &= \sum_{\boldsymbol{x}} (\beta_1 \boldsymbol{S}(\boldsymbol{x})^{\mathrm{T}} \boldsymbol{A}(\boldsymbol{x})^{\mathrm{T}} \boldsymbol{A}(\boldsymbol{x}) \boldsymbol{S}(\boldsymbol{x}) \\ &\quad + \beta_2 \boldsymbol{S}(\boldsymbol{x})^{\mathrm{T}} \boldsymbol{b}(\boldsymbol{x}) \boldsymbol{b}(\boldsymbol{x})^{\mathrm{T}} \boldsymbol{S}(\boldsymbol{x})), \\ \boldsymbol{h}' &= \sum_{\boldsymbol{x}} (\beta_1 \boldsymbol{S}(\boldsymbol{x})^{\mathrm{T}} \boldsymbol{A}(\boldsymbol{x})^{\mathrm{T}} \triangle\boldsymbol{b}(\boldsymbol{x}) \\ &\quad + \beta_2 \boldsymbol{S}(\boldsymbol{x})^{\mathrm{T}} \boldsymbol{b}(\boldsymbol{x}) \triangle c(\boldsymbol{x})), \\ \boldsymbol{P} &= (\boldsymbol{G}')^{-1} \boldsymbol{h}'. \end{aligned} \tag{19}$$

4.2 Iterations and multi-scale

To reduce the error from model mismatch the algorithm is iterated a few times (typically 2–3 times is enough). For each iteration, the coefficients are updated using the previous estimated affine transformation according to the following equations:

$$\begin{aligned} f_{\mathrm{m}}(\boldsymbol{x}) &\sim \boldsymbol{x}^{\mathrm{T}} \boldsymbol{A}_{\mathrm{m}} \boldsymbol{x} + \boldsymbol{b}_{\mathrm{m}}^{\mathrm{T}} \boldsymbol{x} + c_{\mathrm{m}}, \\ f_{\mathrm{m}}(\boldsymbol{F}\boldsymbol{x}) &\sim \boldsymbol{x}^{\mathrm{T}} (\boldsymbol{F}^{\mathrm{T}} \boldsymbol{A}_{\mathrm{m}} \boldsymbol{F}) \boldsymbol{x} + (\boldsymbol{F}^{\mathrm{T}} \boldsymbol{b}_{\mathrm{m}})^{\mathrm{T}} \boldsymbol{x} + c_{\mathrm{m}}, \\ f_{\mathrm{m}}(\boldsymbol{x} + \boldsymbol{t}) &\sim \boldsymbol{x}^{\mathrm{T}} \boldsymbol{A}_{\mathrm{m}} \boldsymbol{x} + (\boldsymbol{b}_{\mathrm{m}} + 2\boldsymbol{A}_{\mathrm{m}} \boldsymbol{t})^{\mathrm{T}} \boldsymbol{x} \\ &\quad + (\boldsymbol{t}^{\mathrm{T}} \boldsymbol{A}_{\mathrm{m}} \boldsymbol{t} + \boldsymbol{b}_{\mathrm{m}}^{\mathrm{T}} \boldsymbol{t} + c_{\mathrm{m}}), \end{aligned}$$

$$\begin{aligned} \boldsymbol{A}_{\mathrm{m}}^{\boldsymbol{F}} &= \boldsymbol{F}^{\mathrm{T}} \boldsymbol{A}_{\mathrm{m}} \boldsymbol{F}, \\ \boldsymbol{b}_{\mathrm{m}}^{\boldsymbol{F}} &= \boldsymbol{F}^{\mathrm{T}} \boldsymbol{b}_{\mathrm{m}}, \\ c_{\mathrm{m}}^{\boldsymbol{F}} &= c_{\mathrm{m}}, \\ \boldsymbol{A}_{\mathrm{m}}^{\boldsymbol{t}} &= \boldsymbol{A}_{\mathrm{m}}, \\ \boldsymbol{b}_{\mathrm{m}}^{\boldsymbol{t}} &= \boldsymbol{b}_{\mathrm{m}} + 2\boldsymbol{A}_{\mathrm{m}} \boldsymbol{t}, \\ c_{\mathrm{m}}^{\boldsymbol{t}} &= \boldsymbol{t}^{\mathrm{T}} \boldsymbol{A}_{\mathrm{m}} \boldsymbol{t} + \boldsymbol{b}_{\mathrm{m}}^{\mathrm{T}} \boldsymbol{t} + c_{\mathrm{m}}. \end{aligned}$$

The same iteration scheme can also be used in each scale. By starting the registration at a low resolution, large deformations can be captured. By applying the estimated transform, re-estimating the residual transform at the next finer scale, and iterating this procedure until reaching the finest scale, large deformations can be estimated accurately in an efficient way. In our experiments, three scales are used, wherein the polynomial expansion is performed at every scale.

5 Multi-affine registration

Global affine registration is often sufficient only to align image data from the same subject. For aligning data from different subjects, global affine registration is normally used as a first step to remove translation and rotation, but then a deformable registration is required to align the corresponding anatomical features.

The analysis outlined in the previous sections on calculating the affine transform is performed globally over the image. If we perform the same analysis for several local regions, we obtain a description of the displacement field in terms of local affine transforms. The local regions are defined by a spatial mask, $\boldsymbol{m}_i$, and for each of them we thus obtain a displacement field $\boldsymbol{d}_i$. In this study we combine these local fields to a non-linear global deformation using a weighted

sum:

$$\boldsymbol{d}(\boldsymbol{x}) = \sum_{i=1}^{K} \frac{m_i(\boldsymbol{x})\boldsymbol{d}_i(\boldsymbol{x})}{\sum_{i=1}^{K} m_i(\boldsymbol{x})}, \tag{20}$$

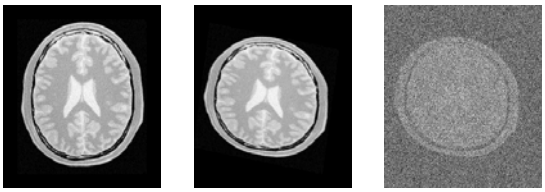
where the partial masks $\boldsymbol{m}_i$ used in this study are Gaussian functions:

$$m_i(\boldsymbol{x}, \sigma_i) = \frac{1}{\sqrt{2\pi}\sigma_i}\exp\left(-\frac{(\boldsymbol{x}-\boldsymbol{x}_i)^{\mathrm{T}}(\boldsymbol{x}-\boldsymbol{x}_i)}{2\sigma_i^2}\right),$$

where σ_i is the variance, and $\boldsymbol{x}_i$ is the spatial centers of the mask. Usually, we choose uniformly distributed grid points as the mask's centers, and the variance of the Gaussian mask varies from big to small.

6 Experiments

This section presents some experimental results of the above affine and multi-affine algorithms using synthesized and real data. For the evaluation of the experiments, we refer to methods described in Hellier *et al.* (2003), Noblet *et al.* (2006), and Gholipour *et al.* (2008). In the first experiment, we compared the affine registration algorithms based on first- and second-order polynomial expansion, together with our previous method, which combined the constraints from the first-order expansion model and one of the constraints from the second-order expansion model. In this experiment, we used a 2D brain magnetic resonance imaging (MRI) T1 image as the target (Fig. 1a). Then we created a group of 20 source images with known linear transformations, and we registered them back to the target using the three affine algorithms described in Section 4. These linear transforms were created according to the following rule: the scaling factor within $[0.90, 1.20]$, the other items of the transform matrix within $[-0.25, 0.25]$, and the translation item within $[-10, 10]$. To verify these algorithms' sensitivity to noise, we further created a group of source images by adding a Gaussian noise to the previously transformed sources. These linear transformations and Gaussian noise were created randomly. After that, we registered the deformed source images and deformed noisy source images back with linear, quadratic, and the combined algorithms, respectively. For each of these registrations, we estimated the linear registration transform and weighted its difference from the known transform that we added on the target using Frobenius' norm. For the registration of deformed sources, the average registration error of the quadratic affine algorithm was 5.81×10^{-3} with a standard deviation of 3.7682×10^{-4}, while the average error of the linear method was 5.85×10^{-3} with a standard deviation of 3.8738×10^{-4}, and the average error of the combined method was 5.88×10^{-3} with a standard deviation of 3.5568×10^{-4}. For the result of deformed noisy sources, the average error of the quadratic algorithm was 0.1011 with a standard deviation of 0.0030, the average error of the linear method was 0.1078 with a standard deviation of 0.0025, and the average error of the combined algorithm was 0.1450 with a standard deviation of 0.0066. The average CPU time of these three algorithms was: quadratic 1.32 s, linear 0.61 s, combined 1.38 s. The combined algorithm required two times polynomial expansion, so a longer time was required. The numbers of iterations were all 5. These results showed that the accuracy and sensitivity to noise of the quadratic model are a little better than those of the other two. The linear model was the fastest because it is the simplest one. All of these three algorithms converged rapidly.



(a) (b) (c)

Fig. 1 The target, one of the deformed source images, and one of the deformed noisy source images. (a) Target image; (b) Image obtained by applying a linear transform to the target with scaler within [0.90, 1.20], other linear items within [–0.25, 0.25], and the translation item within [–10, 10]; (c) Image obtained by adding a Gaussian noise to the deformed source image

In the second experiment, we verified the performance of the quadratic polynomial expansion algorithm using 3D test data. The target volume was a T1 brain volume, downloaded from test data (MNI152_T1_2mm_brain.nii). As in the previous experiment, the source volume was obtained by applying a group of known linear transforms $\boldsymbol{F}$ to the target. In this experiment, we obtained 20 trans-

formed source images using linear transforms; the linear transforms $\boldsymbol{T}$ were created according to: the scale factor $s_i \in [0.8, 1.3]$ $(i = 1, 2, 3)$, the factor $a_i \in [-0.3, 0.3]$ $(i = 1, 2, ..., 6)$, and the translation factor $b_i \in [-10, 10]$ pixels $(i = 1, 2, 3)$. The parameters were then recovered by registering the synthetic source image to the original dataset. One result of the registrations is shown in Fig. 2.

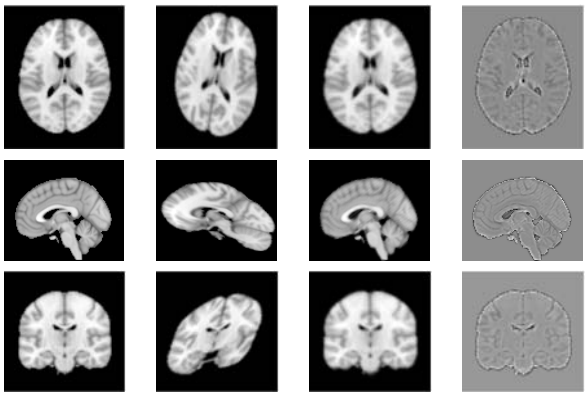


Fig. 2 Test example for 3D using the quadratic affine algorithm. Top row: the target, source, registration result and error images on the axial plane; middle row and bottom row images are the results on sagittal and coronal planes. The rightmost column shows the error as difference images, where the background gray corresponds to zero error. There are only minor errors close to the edges in the image, indicating a good alignment

The estimated transform is denoted as $\boldsymbol{F}'$. The computational time for this experiment, where the size of this data was $91 \times 91 \times 109$, was 88 s on a standard personal computer with a dual core 2.0 GHz CPU and 3 GB of RAM.

$$\boldsymbol{T} = \begin{pmatrix} s_1 & a_1 & a_2 & b_1 \\ a_3 & s_2 & a_4 & b_2 \\ a_5 & a_6 & s_3 & b_3 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

$$\boldsymbol{F} = \begin{pmatrix} 1.1 & -0.2 & -0.3 & 2.0 \\ 0.3 & 0.9 & -0.4 & 3.0 \\ -0.2 & -0.1 & 1.2 & 4.0 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{pmatrix},$$

$$\boldsymbol{F}' = \begin{pmatrix} 1.1004 & -0.2001 & -0.2998 & 2.0030 \\ 0.2999 & 0.9003 & -0.2998 & 3.0048 \\ -0.2000 & -0.1002 & 1.2000 & 4.0024 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{pmatrix}.$$

We used the Frobenius norm to weight the difference between a known transform and the estimated transform. For the error of this experiment, we used the average of the 20 times registrations' errors. Here the average error was 0.0050 and the standard deviation of them was 2.7385×10^{-5}. Furthermore, we used larger parameters of the linear transform to create the source image. We found that, when the translation was larger than 20, the a_i factors larger than 0.5, and the scaling factor larger than 1.5 or smaller than 0.7, the registration error will increase.

In the third experiment, we investigated the power of the multi-affine algorithm presented in Section 5, and compared the registration result with that obtained using the affine algorithm based on quadratic polynomial expansion. The target and source images in this experiment came from different subjects, while under the same scanning parameters setting. The 3D image data was T1-weighted MRI data. In the multi-affine registration process, the number of scales was 2, and the maximal number of iterations was 5 in each scale. The axial central slices of the target, source, and registration result with error images are shown in Fig. 3.

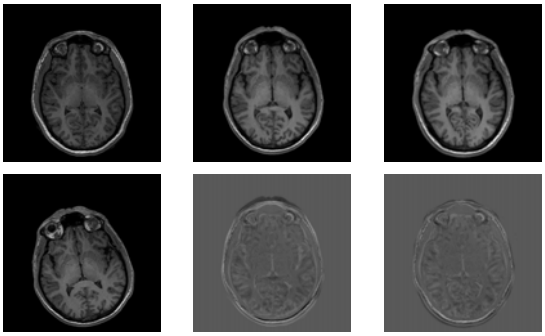


Fig. 3 Comparison of the multi-affine and affine algorithm. Left column: upper is the central slice of the target and lower is that of the source; middle column: upper is the registration result using the affine algorithm, and lower is the error between the affine registered source and the target; right column: upper is the result of the multi-affine algorithm, and lower is the error image

To assess the quality of the registration result, we compared the registration result to the result obtained by the Demons algorithm (Thirion, 1998). For the registration with Demons' algorithm, we chose two scales, and in each scale we iterated the registration process 5, 10, and 20 times. The registration

results are shown in Fig. 4. The upper images are the registration results and the lower images are the difference between the registered image and the target image. As the registration result was not satisfactory, we further did the registration with three scales, 5, 10, and 20 iterations. The results are displayed in Fig. 5. The multi-affine registration algorithm converged faster than Demons' because it needs fewer scales and fewer iterations to achieve a good registration result. In fact, for the multi-affine registration algorithm, two or three iterations are sufficient for obtaining a relatively good registration result.

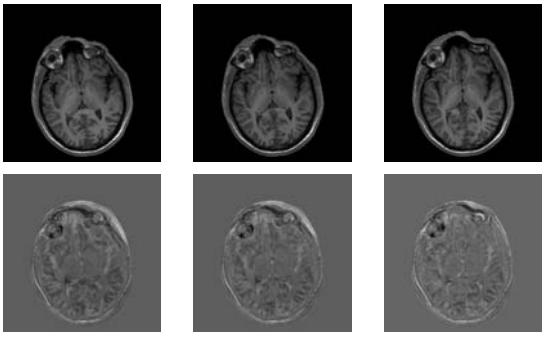


Fig. 4 Registration results of Demons' non-rigid registration algorithm using the same datasets as multi-affine registration (the number of scales is 2). Left column: upper is the central slice of the registration result with 5 iterations and lower is the error compared with the target; middle column: upper is the result with 10 iterations and lower is the error; right column: upper is the result with 20 iterations and lower is the error image

7 Discussion

In this paper, we have described an affine and a multi-affine registration algorithm derived from a local polynomial expansion model. We set the size of local neighborhood for polynomial expansion to 9 or 15 pixels. Comparison of equations of the linear model and the quadratic model (Eqs. (6) and (18)) shows that there are two additional parameters β_1 and β_2 in the quadratic model. We found that in general β_1 should be set smaller than β_2. Usually, we use the parameters $\beta_1 = 0.38$, $\beta_2 = 1.0$, since the weight of the linear item should be larger than that of the quadratic item in function approximation. Another parameter in the multi-affine algorithm is the standard deviation of the Gaussian mask, and

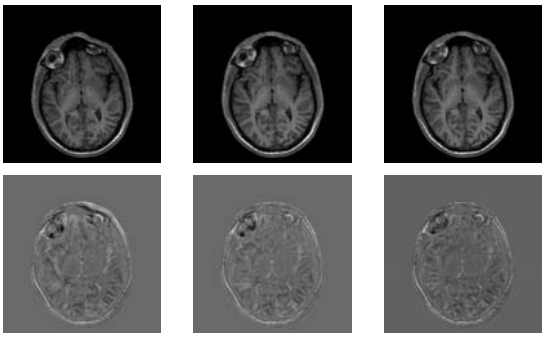


Fig. 5 Registration results of Demons' non-rigid registration algorithm using the same datasets as a multi-affine algorithm (the number of scales is 3). Left column: upper is the central slice of the registration result with 5 iterations and lower is the error compared with the target; middle column: upper is the result with 10 iterations and lower is the error; right column: upper is the result with 20 iterations and lower is the error image

usually we choose $\sigma_i = 15, 30, 60$.

The multi-affine algorithm is obtained by combining several affine registration transforms, where each affine transform is global, but with a focus on a special local region. The idea of multi-affine is very similar to another non-linear registration algorithm named polyaffine (Arsigny *et al.*, 2005). The notable advantage of the polyaffine method is that it guarantees the invertibility of transformation. This property is very helpful when a large displacement exists in the registration process. In multi-affine registration, we apply an affine registration first, and the affine transform we take is a least squares solution of the intensity-based cost function. After that, the large deformations are eliminated, and the multi-affine process combines several locally affine transformations together. Each of these affine transforms is also a local least squares solution. For small deformations, these transforms are all diagonally dominant. Therefore, the final multi-affine deformable transform is still invertible. The main merit of the multi-affine method is its high accuracy and rapid convergence, which is based on the analytical solution and iteration method to search for the parameters of transformation.

8 Conclusions

In this paper, we have presented an affine and a non-linear (multi-affine) registration algorithm

based on local polynomial expansion. We have extended previous work by deriving an expression for the displacement from the quadratic polynomial expansion model and incorporated this into the multi-affine registration method. An advantage of the polynomial expansion model is that the displacement field can be described analytically, which in turn translates into rapid convergence for the algorithm. The presented work improves the accuracy of registration, which ultimately aims at enabling better decision making during the procedures that require alignment of images. The experiments indicate that the presented affine and multi-affine methods are robust. Future work will include a more complete validation study.

Acknowledgements

The software developed for this project is based of the open source software "Spatial Domain Toolbox" (http://www.imt.liu.se/mi/Tools/index.html).

References

Andronache, A., von Siebenhal, M., Szekeley, G., Cattin, P., 2008. Non-rigid registration of multi-modal images using both mutual information and cross-correlation. *Med. Image Anal.*, **12**(1):3-15. [doi:10.1016/j.media.2007.06.005]

Antoine Maintz, J.B., Viergever, M.A., 1998. A survey of medical image registration. *Med. Image Anal.*, **2**(1):1-36. [doi:10.1016/S1361-8415(01)80026-8]

Arsigny, V., Pennec, X., Ayache, N., 2005. Polyrigid and polyaffine transformations: a novel geometrical tool to deal with non-rigid deformations-application to the registration of histological slices. *Med. Image Anal.*, **9**(6):507-523. [doi:10.1016/j.media.2005.04.001]

Farnebäck, G., 2002. Polynomial Expansion for Orientation and Motion Estimation. PhD Thesis, Dissertation No. 790, Linköping Studies in Science and Technology, Sweden.

Farnebäck, G., Westin, C.F., 2006. Affine and deformable registration based on polynomial expansion. *LNCS*, **4190**:857-864.

Gan, R., Chung, A.C.S., Liao, S., 2008. Maximum distance-gradient for robust image registration. *Med. Image Anal.*, **12**(4):452-468. [doi:10.1016/j.media.2008.01.004]

Gholipour, A., Kehtarnavaz, N., Bridggs, R.W., Gopinath, K.S., Ringe, W., Whittemove, A., Cheshkov, S., Bakhadinrov, K., 2008. Validation of nonrigid registration between functional and anatomical magnetic resonance brain images. *IEEE Trans. Biomed. Eng.*, **55**(2):563-571. [doi:10.1109/TBME.2007.912641]

Hellier, P., Barillot, C., Corouge, I., Gibaud, B., le Houalher, G., Collins, D.L., Evans, A., Malandain, G., Ayache, N., Christensen, G.E., *et al.*, 2003. Retrospective evaluation of inter-subject brain registration. *IEEE Trans. Med. Imag.*, **22**(9):1120-1130. [doi:10.1109/TMI.2003.816961]

Hill, D.L.G., Batchelor, P.G., Holden, M., Hawkers, D.J., 2001. Medical image registration. *Phys. Med. Biol.*, **46**(3):R1-R45. [doi:10.1088/0031-9155/46/3/201]

Holden, M., 2008. A review of geometric transformations for nonrigid body registration. *IEEE Trans. Med. Imag.*, **27**(1):111-128. [doi:10.1109/TMI.2007.904691]

Noblet, V., Heinrich, C., Heitz, F., Armspach, J.P., 2006. Retrospective evaluation of a topology preserving non-rigid registration method. *Med. Image Anal.*, **10**(3):366-384. [doi:10.1016/j.media.2006.01.001]

Pennec, X., Cachier, P., Ayache, N., 1999. Understanding the Demons' algorithm: 3D non-rigid registration by gradient decent. *LNCS*, **1679**:597-605. [doi:10.1007/10704282_64]

Pluim, J.P.W., Antoine Maintz, J.B., Viergever, M.A., 2004. f-Information measures in medical image registration. *IEEE Trans. Med. Imag.*, **23**(12):1508-1516. [doi:10.1109/TMI.2004.836872]

Rehman, T., Haber, E., Pryor, G., Melonakos, J., Tannenbaum, A., 2009. 3D nonrigid registration via optimal mass transport on the GPU. *Med. Image Anal.*, **13**(6):931-940. [doi:10.1016/j.media.2008.10.008]

Schnabel, J.A., Tanner, C., Castellano-Smith, A.D., Degenhard, A., Leach, M.O., Hose, D.R., Hill, D.L.G., Hawkes, D.J., 2008. Validation of nonrigid image registration using finite-element methods: application to breast MR images. *IEEE Trans. Med. Imag.*, **22**(3):238-247. [doi:10.1109/TMI.2002.808367]

Thirion, J.P., 1998. Image matching as a diffusion process: an analogy with Maxwell's demons. *Med. Image Anal.*, **2**(3):243-260. [doi:10.1016/S1361-8415(98)80022-4]

Wells, W.M., Viola, P., Atsumi, H., Nakajima, S., Kikinis, R., 1996. Multi-modal volume registration by maximization of mutual information. *Med. Image Anal.*, **1**(1):35-51. [doi:10.1016/S1361-8415(01)80004-9]

Westin, C.F., 1994. A Tensor Framework for Multidimensional Signal Processing. PhD Thesis, Dissertation No. 348, Linköping Studies in Science and Technology, Sweden.

Yuan, Z.N., Wu, F., Zhuang, Y.T., 2006. Multi-sensor image registration using multi-resolution shape analysis. *J. Zhejiang Univ.-Sci. A*, **7**(4):549-555. [doi:10.1631/jzus.2006.A0549]