Warping a Neuro-Anatomy Atlas on 3D MRI Data with Radial Basis Functions

H.E. Bennink, J.M. Korbeeck, B.J. Janssen, B.M. ter Haar Romeny

Eindhoven University of Technology - Department of Biomedical Engineering
Group Biomedical Image Analysis
Eindhoven, the Netherlands

Abstract—Navigation for neurosurgical procedures must be highly accurate. Often small structures are hardly seen on pre-operative scans. Fitting a 3D electronic neuro-anatomical atlas on the data assists with the localization of small structures and dim outlines. During surgery also brainshifts occurs. With intra-operative MRI the pre-operative MRI can be warped to the real 3D situation. The paper describes a general 3D landmark-based warping method, based on radial basis functions (thin plate splines) for data of any number of dimensions, including all code in Mathematica.

Keywords—Brain atlas, image warping, radial basis functions, neurosurgery, computer vision

I. INTRODUCTION

Neurosurgical operations require very precise navigation. Not only during resection surgery of brain tumors, also for deep brain stimulation (DBS), i.e. the placement of electrodes for the electrical stimulation of some specific deep-brain structures. In particular, the placement of electrodes for stimulation of the Nucleus Subthalamus (STN) in the deep brain may enable Parkinson patients to reduce their spontaneous tremor, which is often invalidating them.

Today, with Magnetic Resonance Imaging and Computed Tomography scanners it is possible to image the patient in three dimensions at high resolution, typically 0.5-1 mm in each direction.

However, two major problems exist:
- it is often difficult to find the many individual brain nuclei on these scans due to the low contrast and the noise. The mapping of an electronic brain atlas on the scans substantially helps the navigation (fig. 2) [3]. The atlas is based on a mean patient, and will not fit the data. So a 3D warping method is necessary.
- When the skull is opened, brain shift occurs due to pressure changes and loss of liquids. The pre-operative data do no longer correspond to the real situation, making precise navigation, e.g. with avoidance of bloodvessels, difficult. Recently, an open mobile low-field MRI scanner (fig. 1) has been installed in the Maastricht University Hospital (the first in the Netherlands), to image the patient after the trepanation. Due to the lower image quality of the intra-operative scans, registration and warping of the pre-operative scans is necessary.

Fully automatic registration and 3D warping on the grayscale images is difficult, if not impossible. In this paper we describe a warping based on registration of manually selected anatomical landmarks in 3D. Neurosurgeons are very familiar with such landmarks, and good descriptions [6][7] and neuro-anatomical atlases [8] exist. The 3D deformation field is constructed by means of an interpolation based on thin plate splines [1],[2], a special form of the radial basis functions. In Mathematica [16] the code is very short, readable and efficient.
II. THIN PLATE SPLINES

In order to warp a brain atlas onto an MRI scan of the brain we identify so-called landmarks of which the correspondence between the atlas and the scan is known. Because of this correspondence we can find the vector field that can be used for the warping procedure. The vector field is found by interpolating the known vectors at the location of the landmarks. Hence we first introduce interpolation of arbitrarily spaced points in multiple dimensions.

Imagine a thin metal plate of infinite extent that is fixed at certain points \( \tilde{x}_i = \{x_i, y_i\} \) at the heights \( f_i, i \in I \) (and neglect gravity). The metal plate has a shape such that its surface is minimally bent. The bending energy of such a plate \( s(x, y) \) equals
\[
E_{\text{bend}}(s) = \int_\Omega \left( \frac{\partial^2 s(x, y)}{\partial x^2} + 2 \frac{\partial^2 s(x, y)}{\partial x \partial y} + \frac{\partial^2 s(x, y)}{\partial y^2} \right) dxdy
\]

This is an instance of a semi-norm proposed by Jean Duchon. In order to find the shape of a plate that is fixed at the points mentioned above we will have to find the minimizer of this equation such that the constraints, \( s(\tilde{x}_j) = f_i, \forall i \in I \) are met. The shape can be approximated by finding that \( s \) that minimizes
\[
E(s) = \sum_{i \in I} |s(\tilde{x}_i) - f_i|^2 + \lambda E_{\text{bend}}(s)
\]
The first part of this convex energy functional makes sure the constraints are met and the second part smooths the result. \( \lambda \in \mathbb{R} \) is a parameter that controls the quality (deviation from the constraints) of the approximation. A smaller value of \( \lambda \) results in a better approximation, ultimately achieving interpolation when \( \lambda \) tends to 0. Duchon was one of the first who recognised that the solution of the variational problem can be written in the form
\[
s(\tilde{x}) = \sum_{i \in I} w_i \varphi(||\tilde{x} - \tilde{x}_i||) + p(\tilde{x})
\]

Here \( \varphi(r) = \begin{cases} \log(r) & \text{if } 2k - d \text{ even} \\ r^{2k-d} & \text{if } 2k - d \text{ odd} \end{cases} \) \( k \) is a setting for the order of the norm that will be minimized \( k = 2 \). \( d \) sets the number of dimensions, for our fiducial metal plate \( d = 2 \). \( p(\tilde{x}) \) is a polynomial that lies in the null space of the differential operator that appears in the bending energy. In our example it thus takes the form \( p(x) = a_1 + a_2 x + a_3 y \). In order to find the interpolating (or approximating) function of the form of equation (3) we can simply solve a linear system of equations. We are searching for the solution of
\[
\begin{bmatrix}
\mathbf{B} + \lambda I & \mathbf{Q} \\
\mathbf{Q^*} & \mathbf{O}
\end{bmatrix}
\begin{bmatrix}
\tilde{w} \\
\tilde{a}
\end{bmatrix} =
\begin{bmatrix}
\tilde{f} \\
\tilde{0}
\end{bmatrix}
\]
where \( \{B\}_{i,j} = \varphi(||\tilde{x}_i - \tilde{x}_j||) \), \( \{Q\}_{i,j} = \{1, x, y\} \), \( I \) is the identity matrix and \( O = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \). Those \( \tilde{w} \) and \( \tilde{a} \) that solve the linear system of equation (2) can be plugged into equation (1) in order to find the expression for the shape we are looking for.

III. THE LANDMARKS

Well-known neuro-surgical atlases exist, such as by Talairach-Tournoux [11][12]. Recently, a family of electronic atlases has been brought to market, such as the Cerify series of atlases [8] by Nowinski et al. In these atlases landmark coordinates can conveniently be found [6],[7]. In this study we selected 148 well-defined landmarks in coronal, sagittal and transversal slices in the atlas and in 3D in the patient’s MRI dataset. Some examples are shown in fig. 3.

IV. IMPLEMENTATION

We implemented the method that is outlined above (for a 2D image/metal plate) for N dimensions in Mathematica [16] (code: see Appendix). The module takes input similar to the built-in Interpolation function and returns a compiled function.

There is a great advantage to design this method in the high level software system Mathematica. The code is amazingly compact, and can be easily transferred and made public.
A 2D EXAMPLE

The Mathematica code for the radial basis function interpolation is given in the Appendix. Below, the example code (fig. 4) shows the local warping on a 2D image by randomly moving a small set of 7 randomly chosen landmark points. The radial basis function is a Bessel function in this case (to show the versatility).

```
source = Table[Random[Real, {10, 90}], {7}, {2}];
target = source + Table[Random[Real, {5, 5}], {7}, {2}];
grid = Table[{x, y}, {x, 1, 100, 2.5}, {y, 1, 100, 2.5}];
Show[Graphics[{Gray, Line /@ grid, Line /@ grid, Blue,PointSize[0.02], Point /@ source, Red, Circle[#[1, 2] & /@ target]}, AspectRatio -> 1,
Frame -> True];

rbi = RadialBasisInterpolation[{source, target},
RadialBasisNorm -> rbn];
newgrid = Map[rbi @@ # & , grid, {2}];
Show[Graphics[{Gray, Line /@ newgrid, Line /@ newgrid,
PointSize[0.02], Blue, Point /@ (rbi @@ # & /@ source,
Red, (Circle[#, 2] & /@ target)], AspectRatio -> 1,
Frame -> True}];
```

Fig. 4. Mathematica code to warp a grid based on 7 random location shifts. The main code, which is suitable for data of any dimensionality, is given in the Appendix.

After the warping the full new image is interpolated towards the new coordinates. The new points are placed exactly at the location of the old points, the environment is smoothly deformed by radial basis interpolation. The deformation of the coordinate grid can be clearly appreciated (fig. 5).

V. RESULTS

In total 148 landmarks were set in both the 3D MRI data and the electronic atlas, in coronal, sagittal and transversal slices. After 3D warping by thin plate splines, the result was good around the set landmarks, but unsatisfactory in regions far away from the landmarks. New points were added in those regions (typically 3-10) for a second warping iteration. After three iterations the result was satisfactory. Some examples of the warped atlas on the data are shown in fig. 6.

VI. CONCLUSIONS

An elegant and efficient method is described to warp data in any dimensions based on sets of corresponding landmarks. Automated landmark extraction should be the next step. Fast implementations of this method are appearing, based on graphics card hardware, and splines interpolation. Code in Mathematica [16] is short, easily transferable, educational and applicable to data of any dimension.

ACKNOWLEDGEMENT

The authors like to thank the A*Star Bio-Informatics Institute in Singapore for making available the Cerify Student brain atlas, and invaluable help in the landmark allocation. We thank Markus van Almsick for the MathVisionTools library (www.mathvisiontools.net).

VII. REFERENCES

The Gaussian norm function is implemented as a compiled function calculating a Gaussian weighted vector norm \( \phi(r) = e^{-\frac{r^2}{2a^2}} \).

**Mathematica code:**

```mathematica
Options[RadialBasisInterpolation] = 
{RadialBasisNorm -> Automatic, Smoothness -> 0.};
ThinPlateSplineNorm[1] = 
Compile[{x, y}, 
  Module[{values, valueRank, valueDimensions, 
    valueRank, \phi, \lambda, \phi = (RadialBasisNorm, Smoothness) /, (opts) /, 
    Options[RadialBasisInterpolation]; 
    (points, values) = N[Transpose[data]]; 
    If[Depth[data] = 2, points = List/@points; 
    n, pointDim] = Dimensions[points]; 
    valueDimensions = Rest[Dimensions[values]]; 
    valueRank = Length[valueDimensions]; 
    If[\phi === Automatic, 
    \phi = ThinPlateSplineNorm[Max[2, pointDim]]]; 
    Q = Table[0, {pointDim}, {pointDim}]; 
    B = Outer[\phi, points, points]; 
    Q = (Prepend[Q, 1]) @/points; 
    A = MapThread[Join, 
    B = PadRight[Transpose[values], 
    Append[valueDimensions, n - pointDim + 1], 0]; 
    x = Map[LinearSolve[A, \phi], b, (valueRank)]; 
    w = Map[Take[\phi, n] \phi, x, (valueRank)]; 
    a = Map[Take[\phi, -2, 0] \phi, x, (valueRank)]; 
    slots = Array[Slot, {pointDim}]; 
    Compile[\{\phi, \lambda\}, (Point. \phi) /slots, 
    a, Prepend[slots, 1], w, \phi, slots, Transpose[points]]
}
```

**VIII. APPENDIX: MATHEMATICA CODE**

RadialBasisInterpolation[
{\{x1, y1\}, \{x2, y2\}, ..., \{xn, yn\}}]
constructs a compiled approximate function that interpolates
the y-value at any given x-value. \( x_i \) and \( y_i \) can be tensors
in any number of dimensions. The interpolation function is a function of the form:
\[
y(x) = ax + \sum_{i=1}^{m} w_i \phi(x - x_i) .
\]

Here \( a \) is the linear part of the interpolation function, \( w_i \) is
the weight of basis \( x_i \) and \( \phi \) is a function that calculates the
vector norm of the vector \( (x - x_i) \). The implemented radial basis norm functions \( \phi \) are:

- ThinPlateSplineNorm[p] (fig. 7) with \( p = 2k - d \)
- GaussianNorm[\sigma]

Thin plate splines are fundamental solutions of \( \Delta \phi = 0 \).
In \( d \) dimensions these solutions are given by
\( \phi^{(k)}(r) = r^{2k-\Delta} \log r \) for \( k \geq d \) and \( d \) even and
\( \phi^{(k)}(r) = r^{2k-\Delta} \) for \( k \geq d \) and \( d \) odd.

![Fig. 7. ThinPlateSplineNorm[p] functions for p = 2 (red), p = 3 (green) and p = 4 (blue).](image-url)

Address of the corresponding author:

Author: Bart M. ter Haar Romeny
Institute: Eindhoven University of Technology
Street: Den Dolech 2 – WH2 108
City: NL-5600 MB Eindhoven
Country: the Netherlands
Email: B.M.terHaarRomeny@tue.nl