Transformation Functions for Image Registration

Image Fusion Systems Research

ag@imgfsr.com

Abstract

This paper presents a survey of past transformation functions for image registration and introduces a new transformation function for registration of images containing rigid as well as nonrigid regions. Included in the survey are similarity transformation, linear transformation, and nonlinear transformation. Nonlinear transformation itself includes thin-plate splines, radial basis functions, approximation methods, piecewise methods, and piecewise approximation methods. Transformation functions for registration of both 2-D and 3-D (volumetric) images are discussed and their properties are investigated. Accuracies of the transformations in registration of images with various degrees of geometric differences are determined and compared.

Keywords: Image Registration, Transformation Function, Thin-Plate Splines, Radial Basis Functions, Piecewise Functions, Approximation, Piecewise Approximation

1 Introduction

Image registration is a computational method for determining the point-by-point correspondence between two images of a scene, which then may be used to fuse complementary information in the images or estimate the geometric and/or intensity difference between the images. The method generally involves determining a number of corresponding control

points in the images and, from the correspondences, determining a transformation function that will determine the correspondence between the remaining points in the images.

This paper discusses the second step in image registration, that is, the determination of a transformation function that will map one image point-by-point to the other. Although determination of corresponding control points in two images is a very important step in image registration, this paper will not cover that. Methods for selection of control points [2, 3, 15, 21, 25, 34, 40, 43, 44, 55, 68, 69, 70] and methods for determination of correspondence between control points [7, 8, 11, 14, 16, 19, 23, 24, 37, 38, 41, 44, 52, 53, 58, 64, 68, 71, 74, 75] have been discussed elsewhere. We assume that a number of corresponding control points in two images of a scene are given and will determine a transformation function that will use the correspondences to map one image point-by-point to the other.

More specifically, the problem to be solved is as follows. Given the coordinates of N corresponding control points in two images of a scene:

$$\{(x_i, y_i), (X_i, Y_i) : i = 1, \ldots, N\}, \tag{1}$$

we want to determine a transformation function $\mathbf{f}(x, y)$ with components $f_x(x, y)$ and $f_y(x, y)$ that satisfy

$$X_i = f_x(x_i, y_i), \tag{2}$$

$$Y_i = f_y(x_i, y_i), \qquad i = 1, \ldots, N, \tag{3}$$

or

$$X_i \approx f_x(x_i, y_i), \tag{4}$$

$$Y_i \approx f_y(x_i, y_i), \qquad i = 1, \ldots, N. \tag{5}$$

Once $\mathbf{f}(x, y)$ is determined, given the coordinates of a point (x, y) in one image, the coordinates of the corresponding point in the other image can be determined. We will refer to the image with coordinates (x, y) as the *reference* and the image with coordinates (X, Y) as the *sensed*. The reference image is usually kept unchanged and sensed image is a newly scanned

or sensed image whose geometry must be changed to resemble that of the reference image so that the images can be overlaid and compared point-by-point.

If we rearrange the coordinates of corresponding control points in the images into two sets of 3-D points as follows,

$$\{(x_i, y_i, X_i) : i = 1, \ldots, N\}, \tag{6}$$

$$\{(x_i, y_i, Y_i) : i = 1, \ldots, N\}, \tag{7}$$

we see that the components of the transformation as defined by equations (2) and (3) represent two single-valued surfaces interpolating the 3-D points given in (6) and (7), and the components of the transformation defined by (4) and (5) represent single-valued surfaces approximating point sets in (6) and (7), respectively. The components of the transformation for registering two 2-D images are, therefore, single-valued surfaces in 3-D. Since the two components of a transformation are similar, they can be determined in the same manner. In the following, given points

$$\{(x_i, y_i, f_i) : i = 1, \ldots, N\}, \tag{8}$$

we will consider the general problem of finding a single-valued function $f(x, y)$ that interpolates or approximates the points. That is,

$$f_i = f(x_i, y_i), \qquad i = 1, \ldots, N, \tag{9}$$

or

$$f_i \approx f(x_i, y_i), \qquad i = 1, \ldots, N. \tag{10}$$

For registration of volumetric images, the control points are in 3-D and the set of corresponding control points can be written as,

$$\{(x_i, y_i, z_i), (X_i, Y_i, Z_i) : i = 1, \ldots, N\}. \tag{11}$$

If we separate the coordinates of corresponding points into three sets of 4-D points,

$$\{(x_i, y_i, z_i, X_i) : \quad i = 1, \ldots, N\}, \tag{12}$$

$$\{(x_i, y_i, z_i, Y_i): \quad i = 1, \ldots, N\}, \tag{13}$$

$$\{(x_i, y_i, z_i, Z_i): \quad i = 1, \ldots, N\}, \tag{14}$$

the single-valued hypersurfaces interpolating the 4-D points:

$$X_i = f_x(x_i, y_i, z_i), \tag{15}$$

$$Y_i = f_y(x_i, y_i, z_i), \tag{16}$$

$$Z_i = f_z(x_i, y_i, z_i), \qquad i = 1, \ldots, N, \tag{17}$$

or approximating them:

$$X_i \approx f_x(x_i, y_i, z_i), \tag{18}$$

$$Y_i \approx f_y(x_i, y_i, z_i), \tag{19}$$

$$Z_i \approx f_z(x_i, y_i, z_i), \qquad i = 1, \ldots, N, \tag{20}$$

will represent the components of a transformation function that will register the 3-D images. Again, because the three components of the transformation can be found in the same manner, we will consider the general problem of finding a single-valued hypersurface $f(x, y, z)$ that interpolates or approximates point set

$$\{(x_i, y_i, z_i, f_i): i = 1, \ldots, N\}. \tag{21}$$

A component of a transformation function can be represented by a variety of functions. The type of function chosen should depend on the type and severity of the geometric difference between images, the accuracy of the control-point correspondences, and the density and organization of the control points. The objective of this paper is to examine the properties of different transformation functions and provide a guide to them with respect to variations in input.

Because of the nonlinear nature of image acquisition systems and, sometimes, the deformable nature of the scene, the images to be registered often have nonlinear geometric differences. Sometimes, the nonlinear geometric difference between the images is negligible

and a linear transformation can be used to register them. In other cases, the geometric difference between images can be large and complex, and a composite of local transformations are needed to register them. The problem becomes even more difficult when the images to be registered contain rigid as well as nonrigid bodies. The transformation function for registration of such images should be able to rigidly register parts of the images while nonrigidly registering the rest.

Knowing a set of corresponding control points in two images, many transformation functions can be used to accurately map the control points to each other. A proper transformation function will map the remaining points in the images accurately as well. Some transformation functions, although mapping corresponding control points to each other accurately, they warp the sensed image too much, causing large errors in the registration away from the control points. Also, since a transformation is computed from the control point correspondences, error in the correspondences will carry over to the transformation function. It is desired for a transformation to smooth the noise and small inaccuracies in the correspondences. Therefore, when noise and inaccuracies are present, approximation methods are preferred over interpolating methods in image registration.

Various transformations have been used in image registration. In the following sections, the properties of the transformations are examined and a guide to them is provided. Also, a new transformation is introduced that maps corresponding control points as well as their tangents to each other, enabling the registration of images with varying rigidities.

To determine the behavior of transformation functions and to find their accuracies in image registration, a number of test images are prepared as shown in Fig. 1. Fig. 1a is an image of size 512×512 pixels containing a 64×64 uniform grid, and Fig. 1b shows the same grid after being translated by (5,8) pixels. Fig. 1c shows counterclockwise rotation of the grid by 0.1 radians about the image center, Fig. 1d shows scaling of the grid by 1.1 with respect to the image center, Fig. 1e shows the grid after linear transformation

$$X = 0.7x - y + 3, \tag{22}$$

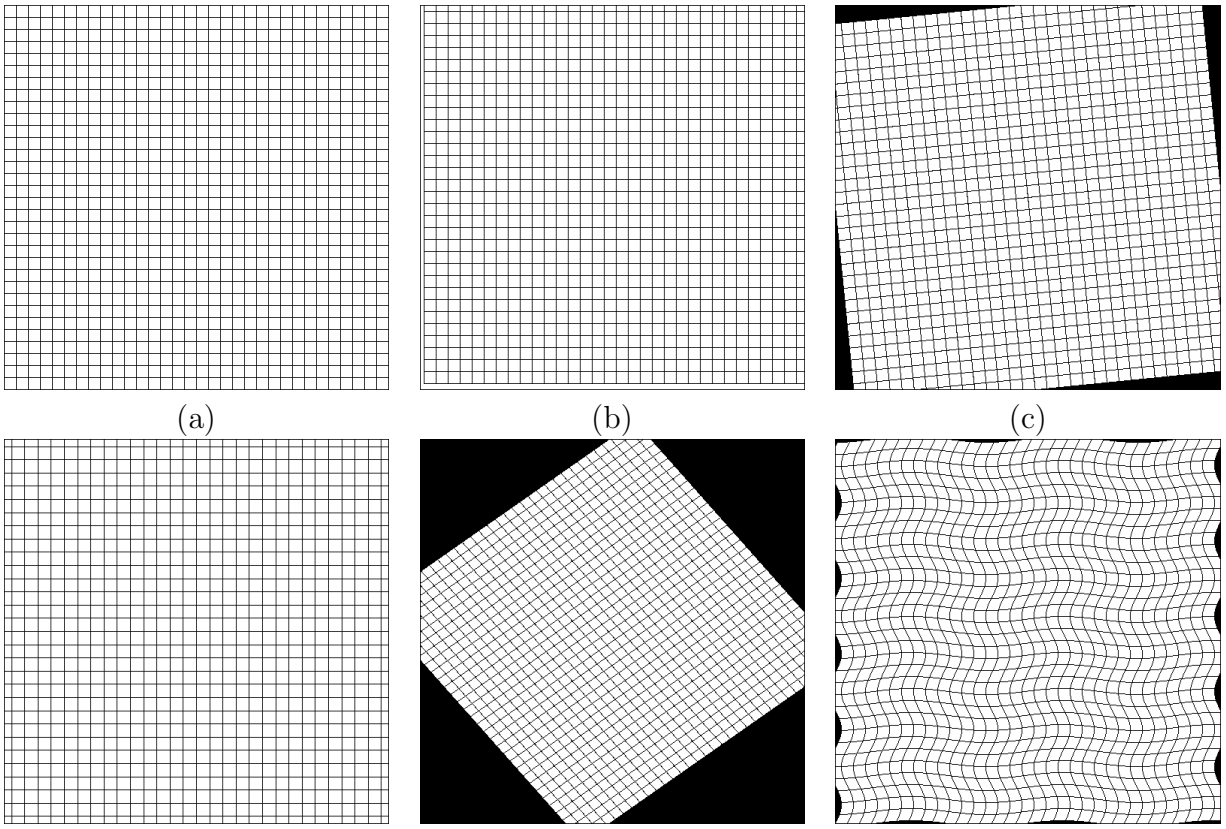


Figure 1: (a) A 64×64 uniform grid in an image of size 512×512 pixels. (b) The grid after translation by (5,8) pixels. (c) The grid after rotation by 0.1 radians counterclockwise about the image center. (d) The grid after scaling by 1.1 with respect to the image center. (e) The grid after the linear transformation defined by (22) and (23). (f) The grid after the sinusoidal transformation defined by (24) and (25).

$$Y = 0.9x + 0.8y + 5, \tag{23}$$

and Fig. 1f shows nonlinear transformation of the grid by sinusoidal function

$$X = x - 8\sin(x/16), \tag{24}$$

$$Y = y + 4\cos(y/32). \tag{25}$$

We will use the coordinates of all or a subset of the corresponding grid points in the images to estimate these transformations.

Different density and organization of point correspondences will be used to estimate the

geometric difference between Fig. 1a and Figs. 1b–f. Fig. 2a shows all grid points in Fig. 1a, Fig. 2b shows a uniformly spaced subset of the grid points, Fig. 2c shows a random subset of the grid points, and Fig. 2d shows a subset of the grid points with variations in the density of the points. Although uniformly spaced control points rarely are obtained in images, they are used here to determine the influence of the spacing and organization of the control points on registration accuracy. Five geometric transformations are applied to the reference image to obtain the sensed images, and four sets of control points are used to estimate the transformation in each case. Therefore, overall, twenty test cases are available.

The control points in the reference image are the grid points shown in Figs. 2a–d. The control points in the sensed image are obtained by 1) translating the control points in the reference image by (5,8) pixels, 2) rotating the control points by 0.1 radians about the image center and rounding the obtained coordinates, 3) scaling the control points by 1.1 with respect to the image center and rounding the coordinates, 4) transforming the control points using the linear transformation given in (22) and (23) and rounding the coordinates, and 5) transforming the control points using the sinusoidal transformation given by (24) and (25) and rounding the coordinates. From the coordinates of corresponding control points the transformations are estimated and compared to the actual ones. In these experiments, although mismatches do not exist, corresponding control points contain quantization noise.

2 Similarity Transformations

The similarity transformation or the transformation of Cartesian coordinate systems represent global translational, rotational, and scaling differences between two images and is described by

$$X = xs\cos\theta - ys\sin\theta + h, \tag{26}$$

$$Y = xs\sin\theta + ys\cos\theta + k. \tag{27}$$

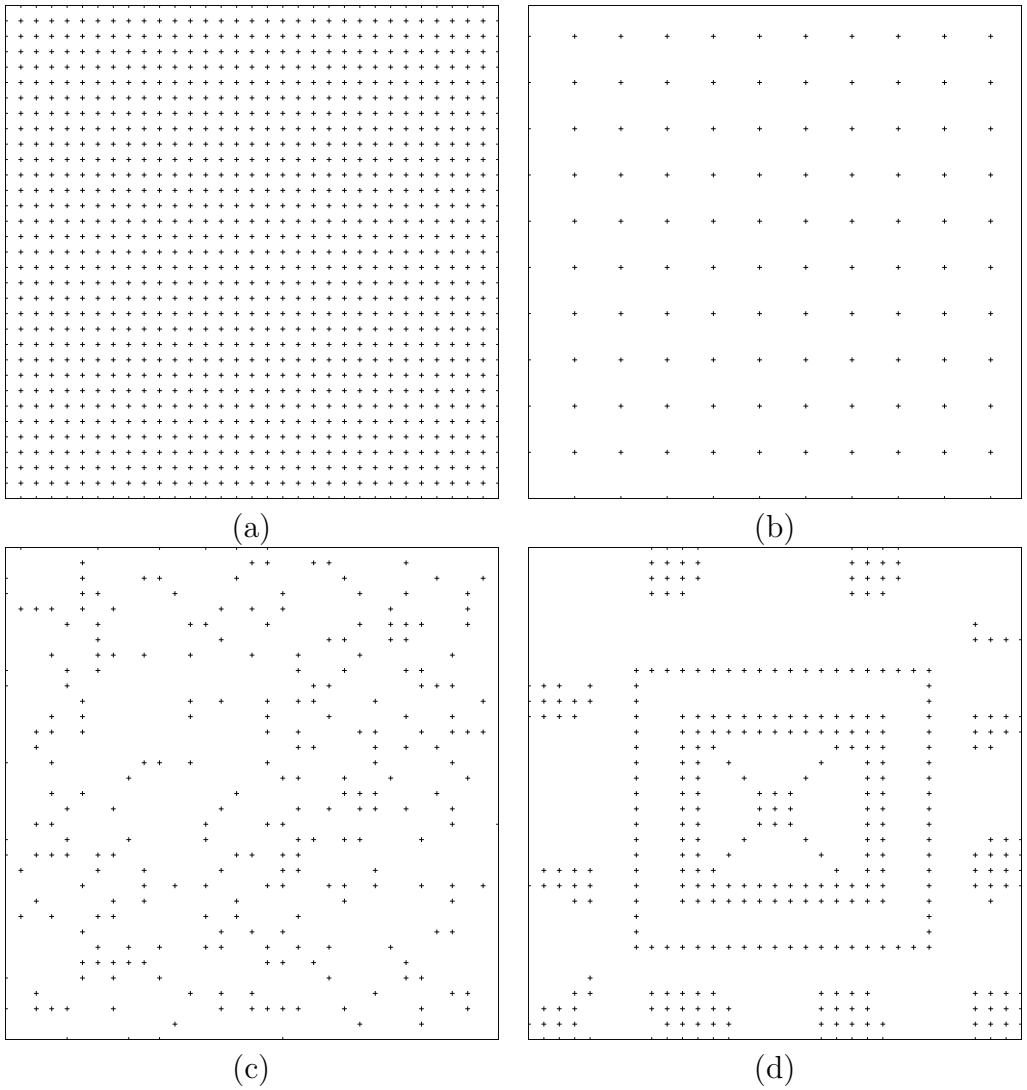


Figure 2: Various density and organization of control points. (a) A grid of uniformly spaced control points. (b) A uniform grid with fewer control points. (c) A set of randomly spaced control points. (d) A set of control points with a varying density.

s, θ, and (h, k) are scaling, rotational, and translational differences between the images, respectively. These four parameters can be determined if the coordinates of two corresponding control points in the images are known. The rotational difference between the images is determined from the angle between the lines connecting the two points in the images. The scaling difference between the images is determined from the ratio of distances between the points in the images. Knowing the scale s and the rotation θ, the translation parameters (h, k) are then determined by substituting the coordinates of one of the correspondences into equations (26) and (27) and solving for h and k.

If the correspondences are known to be noisy or inaccurate, more than two correspondences should be used to determine the parameters by least-squares [54] or clustering [68]. Least-squares is preferred when the inaccuracies can be modeled by zero-mean noise, while clustering is preferred when large errors (outliers) exist among the correspondences.

In 3-D, the similarity transformation can be written as

$$\mathbf{P} = s\mathbf{R}\mathbf{p} + \mathbf{T}, \tag{28}$$

where $\mathbf{P}$ and $\mathbf{p}$ represent corresponding control points in the sensed and reference images, s is the scaling, $\mathbf{R}$ is an orthonormal matrix representing the rotational, and $\mathbf{T}$ is a vector representing the translational differences between the images. First, the scaling difference between images is determined from the average ratio of distances between corresponding point pairs in the images. Knowing the scaling parameter s, the translation vector $\mathbf{T}$ and the rotation matrix $\mathbf{R}$ are determined as follows. Assuming the coordinates of corresponding points in the images after correction for scale are: $\{\mathbf{p}_i = (x_i, y_i, z_i), \mathbf{P}_i = (X_i, Y_i, Z_i) : i = 1, \ldots, N\}$, the relation between corresponding control points in the images can be written as

$$\mathbf{P}_i = \mathbf{R}\mathbf{p}_i + \mathbf{T}, \qquad i = 1, \ldots, N. \tag{29}$$

Vector $\mathbf{T}$ and matrix $\mathbf{R}$ are determined by minimizing

$$E^2 = \sum_{i=1}^{N} ||\mathbf{P}_i - (\mathbf{R}\mathbf{p}_i + \mathbf{T})||^2. \tag{30}$$

An efficient algorithm for determining $\mathbf{R}$ and $\mathbf{T}$ has been described by Arun *et al.* [1]. In this method the rotation matrix is obtained first by minimizing

$$E_R^2 = \sum_{i=1}^{p} ||\mathbf{Q}_i - \mathbf{R}\mathbf{q}_i||^2, \tag{31}$$

and then the translation vector $\mathbf{T}$ is determined from

$$\mathbf{T} = \mathbf{P} - \mathbf{R}\mathbf{p}. \tag{32}$$

In equation (31), $\mathbf{Q}_i = \mathbf{P}_i - \bar{\mathbf{P}}$, $\mathbf{q}_i = \mathbf{p}_i - \bar{\mathbf{p}}$, and $\bar{\mathbf{P}}$ and $\bar{\mathbf{p}}$ are the centers of gravity of the control points in the sensed and reference images, respectively.

If the correspondences are known to be accurate, the translational and rotational differences between the images can be determined from the coordinates of three non-colinear corresponding points in the images. If quantization noise is present, more than three corresponding control points are needed to reduce the effect of noise and determine the registration parameters by the least-squares method as outlined above. If outliers are present also, parameters determined from combinations of three correspondences at a time should be clustered to obtain the transformation parameters. Note that if three corresponding control points are used, determination of the registration parameters involves solution of a nonlinear system of equations, while if four or more points are used, the transformation parameters can be determined by solving a linear system of equations [16].

Similarity transformations are for registration of rigid bodies, but the least squares or clustering methods do not use any rigidity constraint to find the parameters. If noise is not zero-mean and/or a large number of outliers exist, the least squares and clustering methods may fail to find the correct transformation parameters. To find the transformation parameters by specifically using the rigidity constraint, in [17], from a large set of corresponding control points, about a dozen correspondences that had the highest match rating and were widely spread over the image domain were selected. Then, transformation parameters were determined from combinations of four points at a time, and the combination that produced a linear transformation closest to a rigid transformation was used to register the images. The

idea is to use a few correspondences that are very accurate instead of using a large number of correspondences, some of which are inaccurate.

Since similarity transformations are for registration of rigid-bodies, they can be used to register some aerial and satellite images where the scene is rather flat and the platform is at a distance looking down at a normal angle to the scene. In medical images, bony structures can be registered with this transformation. Similarity transformation is widely used in the registration of brain images since the brain is contained in a rigid skull and images of the brain taken a short time apart do not have nonlinear geometric differences, if sensor nonlinearities do not exit.

3 Projective and Linear Transformations

Image acquisition is a projective process and if lens and sensor nonlinearities do not exist, the relation between two images of a rather flat scene can be described by a projective transformation:

$$X = \frac{ax + by + c}{dx + ey + 1}, \tag{33}$$

$$Y = \frac{fx + gy + h}{dx + ey + 1}. \tag{34}$$

a–*h* are the eight unknown parameters of the transformation, which can be determined if the coordinates of four non-colinear corresponding control points in the images are known. If correspondences are known to contain noise, more than four correspondences should be used in a least squares or clustering method to obtain the transformation parameters. When the scene is very far from the camera, the projective transformation can be approximated by a linear or affine transformation:

$$X = ax + by + c, \tag{35}$$

$$Y = dx + ey + f. \tag{36}$$

A linear transformation has six parameters, which can be determined if the coordinates of at least three non-colinear corresponding points in the images are known. Linear transformation

is a weak perspective transformation and has been widely used in the registration of satellite and aerial images [4, 67].

The projective transformation requires that straight lines in the reference image remain straight in the sensed image. If parallel lines remain parallel, affine transformation may be used instead, and if angles between corresponding lines are preserved, the transformation of the Cartesian coordinate system may be used to register the images. Images that can be registered by the transformation of the Cartesian coordinate system can be registered by the affine transformation, and images that can be registered by the affine transformation can be registered by the projective transformation. If a straight line in the reference image maps to a curve in the sensed image, a nonlinear transformation is needed to register the images. Nonlinear transformations are discussed next.

4 Thin-Plate Splines

Thin-plate splines (TPS) or surface splines [33, 49] are perhaps the most widely used transformation functions in the registration of images with nonlinear geometric differences. It was first used by Goshtasby [26] in the registration of remote sensing images and then by Bookstein [6] in the registration of medical images. Given a set of 3-D points as defined by (8), the thin-plate spline interpolating the points is defined by

$$f(x,y) = A_1 + A_2x + A_3y + \sum_{i=1}^{N} F_i r_i^2 \ln r_i^2, \tag{37}$$

where $r_i^2 = (x-x_i)^2+(y-y_i)^2+d^2$. This is the equation of a plate of infinite extent deforming under loads at $\{(x_i, y_i) : i = 1, \ldots, N\}$. The plate deflects under the imposition of loads to take values $\{f_i : i = 1, \ldots, N\}$. d^2 acts like a stiffness parameter. As d^2 approaches zero, the loads approach point loads. As d^2 increases, the loads become more widely distributed producing a smoother surface. Equation (37) contains $N+3$ unknowns. By substituting the coordinates of N points as described by (8) into (37), N relations will be obtained. Three

more relations are obtained from the following constraints:

$$\sum_{i=1}^{N} F_i = 0, \tag{38}$$

$$\sum_{i=1}^{N} x_i F_i = 0, \tag{39}$$

$$\sum_{i=1}^{N} y_i F_i = 0. \tag{40}$$

Constraint (38) ensures that the sum of the loads applied to the plate is 0 so that the plate will remain stationary. Constraints (39) and (40) ensure that moments with respect to x and y axes are zero and so the surface will not rotate under the imposition of the loads. A minimum of three points are needed to obtain a thin-plate spline. For instance, when seven points with coordinates $(1, 1, 10)$, $(1, 20, 10)$, $(20, 1, 10)$, $(20, 20, 10)$, $(11, 8, 15)$, $(7, 17, 25)$, and $(16, 12, 20)$ are used and $d^2 = 0$, we find $A_1 = 61.622$, $A_2 = 0.079$, $A_3 = -0.323$, $F_1 = 0.0001$, $F_2 = 0.0313$, $F_3 = 0.0695$, $F_4 = -0.0498$, $F_5 = 0.0344$, $F_6 = -0.0014$, and $F_7 = -0.0215$.

Using thin-plate splines to map Figs. 1b–f to Fig. 1a with the control point arrangements shown in Fig. 2, we obtain the results shown in Table 1. The errors in each entry of the table are obtained by using the coordinates of corresponding control points to determine the transformation parameters. The obtained transformation is then used to deform the sensed image to overlay the reference image. This transformation exactly maps the control points in the sensed image to corresponding control points in the reference image. The control points shown in Figs. 2b–d were used to estimate the transformation functions, and the control points shown in Fig. 2a were used to estimate the maximum (MAX) error and the root-mean-squared (RMS) error in each case. All errors in the first row in Table 1 are zero because TPS is an interpolating function and it maps corresponding control points exactly to each other. Since the control points used to compute the transformation and the control points used to measure the error are the same, no errors are obtained. Fig. 3a shows the resampling of Fig. 1f to align with Fig. 1a using all the control points. The control points shown in Fig. 2a are overlaid with the resampled image. Although corresponding control

Table 1: The registration accuracy of TPS under different density and organization of the control points as well as geometric difference between images.

		Translation	Rotation	Scaling	Linear	Nonlinear
Uniform (Dense)	MAX	0.0	0.0	0.0	0.0	0.0
	RMS	0.0	0.0	0.0	0.0	0.0
Uniform (Sparse)	MAX	0.0	1.4	1.4	1.4	27.0
	RMS	0.0	0.6	0.6	0.6	5.8
Random	MAX	0.0	1.4	1.4	1.4	14.1
	RMS	0.0	0.7	0.6	0.6	3.5
Irregular	MAX	0.0	2.2	2.0	1.4	20.0
	RMS	0.0	0.8	0.7	0.7	6.7

points map to each other exactly, errors are visible away from the control points, especially around image borders.

When the images have translational differences, no errors are observed in registration independent of the density and organization of the points. When images have rotational, scaling, or linear differences, TPS produces small errors that are primarily due to quantization noise since nearest neighbor resampling is used to generate the transformed image. Errors slightly increase when the density of control points varies across the image domain. When images have nonlinear geometric differences, errors are quite large as shown in the last column in Table 1. Although corresponding control points are mapped to each other, at non-control points, errors increase as one moves away from the control points. Figs. 3a–d show resampling of image 1f to align with image 1a using the control point arrangements shown in Figs. 2a–d, respectively. The control points used in the computations are overlaid with the resampled images for qualitative evaluation of the registration.

From the examples shown in Fig. 3, we see that thin-plate splines are not suitable for registration of images with local geometric differences. In Figs. 3c and 3d we see that although reasonably good accuracy is achieved at and near the control points, errors are large away from the control points. This can be attributed to the fact that logarithmic basis

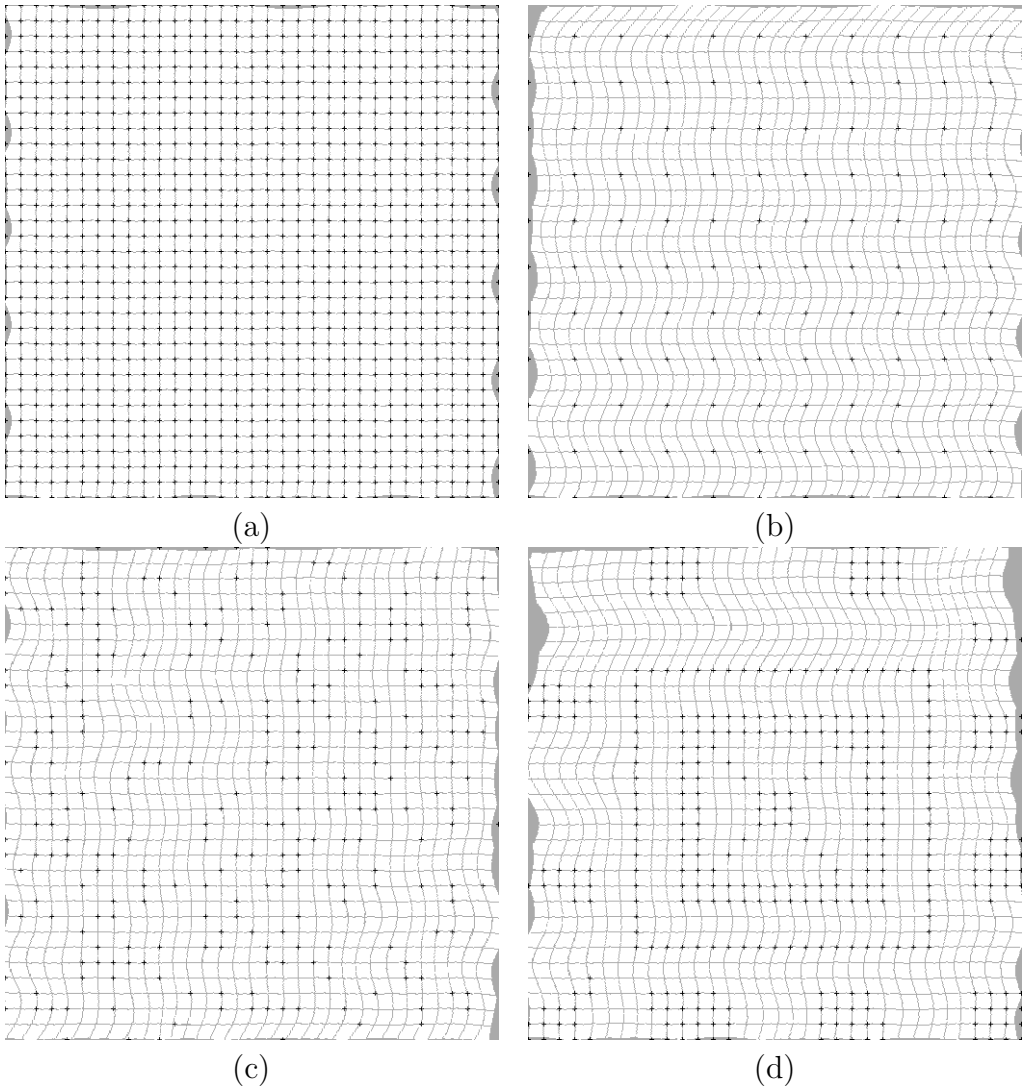


Figure 3: The effect of density and organization of control points in registration accuracy of TPS. (a)–(d) Resampling of Fig. 1f using the control points shown in Figs. 2a–d, respectively, to align with Fig. 1a.

functions, which are rotationally symmetric, are used to define the transformation. When the arrangement of the control points is nonuniform, large errors are obtained in areas where large gaps exist between control points. One should also note that as the number of control points increases, since the matrix of coefficients becomes increasingly large, the numerical stability of the method decreases.

The stiffness parameter d^2 will change the shape of the interpolating surface and that will change the registration accuracy. With the control point arrangements given in Fig. 2, best accuracy was obtained when stiffness parameter was zero. Larger stiffness parameters increased registration errors in these tests cases. The errors reported in Table 1 are for a stiffness parameter of 0. To avoid the creation of high fluctuations away from the control points, Rohr et al. [56, 72] added a smoothing term to the interpolating spline while setting $d^2 = 0$ to obtain a surface that contains smaller fluctuations but approximates the points. As the smoothness term is increased, the obtained surface becomes smoother and fluctuations become smaller, but the surface will get farther from some of the control points. If interaction with the system is allowed, one may gradually change the smoothness parameter while observing the components of the transformation and stopping the process when the surfaces get sufficiently close to the control points. When the control points are noisy and/or are very irregularly spaced, approximating splines should produce better results than interpolating splines. This, however, necessitates interaction with the system to ensure that the surfaces are not overly smoothed.

Thin-plate splines can be extended to volume splines for the registration of volumetric images [18, 72]. A component of a transformation function for the registration of volumetric images will be a single-valued hypersurface in 4-D:

$$f(x, y, z) = A_1 + A_2 x + A_3 y + A_4 z + \sum_{i=1}^{N} F_i r_i^2 \ln r_i^2, \tag{41}$$

where $r_i^2 = (x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 + d^2$. Parameters A_1, A_2, A_3, and A_4 as well as F_i for $i = 1, \ldots, N$ are determined by solving a system of $N + 4$ linear equations. N equations are obtained by substituting coordinates (21) into (41) and letting $f(x_i, y_i, z_i) = f_i$. Four

more equations are obtained from constraints

$$\sum_{i=1}^{N} F_i = 0, \tag{42}$$

$$\sum_{i=1}^{N} x_i F_i = 0, \tag{43}$$

$$\sum_{i=1}^{N} y_i F_i = 0, \tag{44}$$

$$\sum_{i=1}^{N} z_i F_i = 0. \tag{45}$$

These constraints ensure that the spline will not translate or rotate under the loads.

5 Radial Basis Functions

Radial basis functions are, in general, described by

$$f(x, y) = \sum_{i=1}^{N} F_i R_i(x, y). \tag{46}$$

Parameters $\{F_i : i = 1, \ldots, N\}$ are determined by letting $f(x_i, y_i) = f_i$ for $i = 1, \ldots, N$ and solving the obtained system of linear equations. In 2-D, $R_i(x, y)$ is a radial function whose value is proportional or inversely proportional to the distance between (x, y) and (x_i, y_i), that is, for monotonically increasing functions $R_i(x, y) = \{(x - x_i)^2 + (y - y_i)^2\}^{1/2}$. $R_i(x, y)$ shows the weight or influence of F_i on functional value at (x, y). A surface point, therefore, is obtained from a weighted sum of F_i's. Powell [51] provides an excellent review of radial basis functions.

When

$$R_i(r) = [(x - x_i)^2 + (y - y_i)^2 + d^2]^{\frac{1}{2}}, \tag{47}$$

$f(x, y)$ is called a multiquadric (MQ) function [35, 36]. As d^2 is increased, a smoother surface is obtained. When basis functions are logarithmic, the obtained transformation is similar to a thin-plate spline except for a linear term. In a comparative study carried out by Franke [22], it was found that multiquadrics followed by thin-plate splines produced the best accuracy in the interpolation of randomly spaced data.

When monotonically increasing radial basis functions are used, the farther a data point is from point (x, y), the larger the influence of that data point will be on the functional value at (x, y). This means that if radial basis functions with monotonically increasing basis functions are used in image registration, control points farther away will have a larger influence on the transformation at a particular point than nearer control points. Radial basis functions that are monotonically decreasing appear to be more appropriate in image registration since a surface point will depend more on nearby control points than on ones farther away.

Examples of radial basis functions that monotonically decrease are Gaussians [30, 60],

$$R_i(x, y) = \exp\{-\frac{(x - x_i)^2 + (y - y_i)^2}{2\sigma_i^2}\}, \tag{48}$$

and inverse multiquadrics [9, 22, 36],

$$R_i(x, y) = [(x - x_i)^2 + (y - y_i)^2 + d^2]^{-\frac{1}{2}}. \tag{49}$$

Franke [22] has shown through extensive experimentation that monotonically decreasing radial basis functions do not perform as well as monotonically increasing radial basis functions. When applied to image registration, errors obtained from radial basis functions that are proportional to inverse distances, such as inverse multiquadrics and Gaussians, are especially large around image borders where the sum of the basis functions quickly drops.

Registering Figs. 1b–f and Fig. 1a using multiquadrics with the control points shown in Fig. 2, we obtain the results shown in Fig. 4. The error measures are summarized in Table 2. Comparing these results with those obtained in Table 1, we see that TPS generally performs better than MQ, especially when the transformation function to be estimated is linear. This can be attributed to the linear term in TPS. However, when the transformation function is nonlinear, MQ is slightly better.

Radial basis functions can be defined locally so that the coordinates of a small number of control points will be sufficient to find a functional value. Such functions are called compactly supported radial basis functions. Wendland [73] describes a family of compactly supported radial basis functions that are positive definite, and guarantee a solution. For

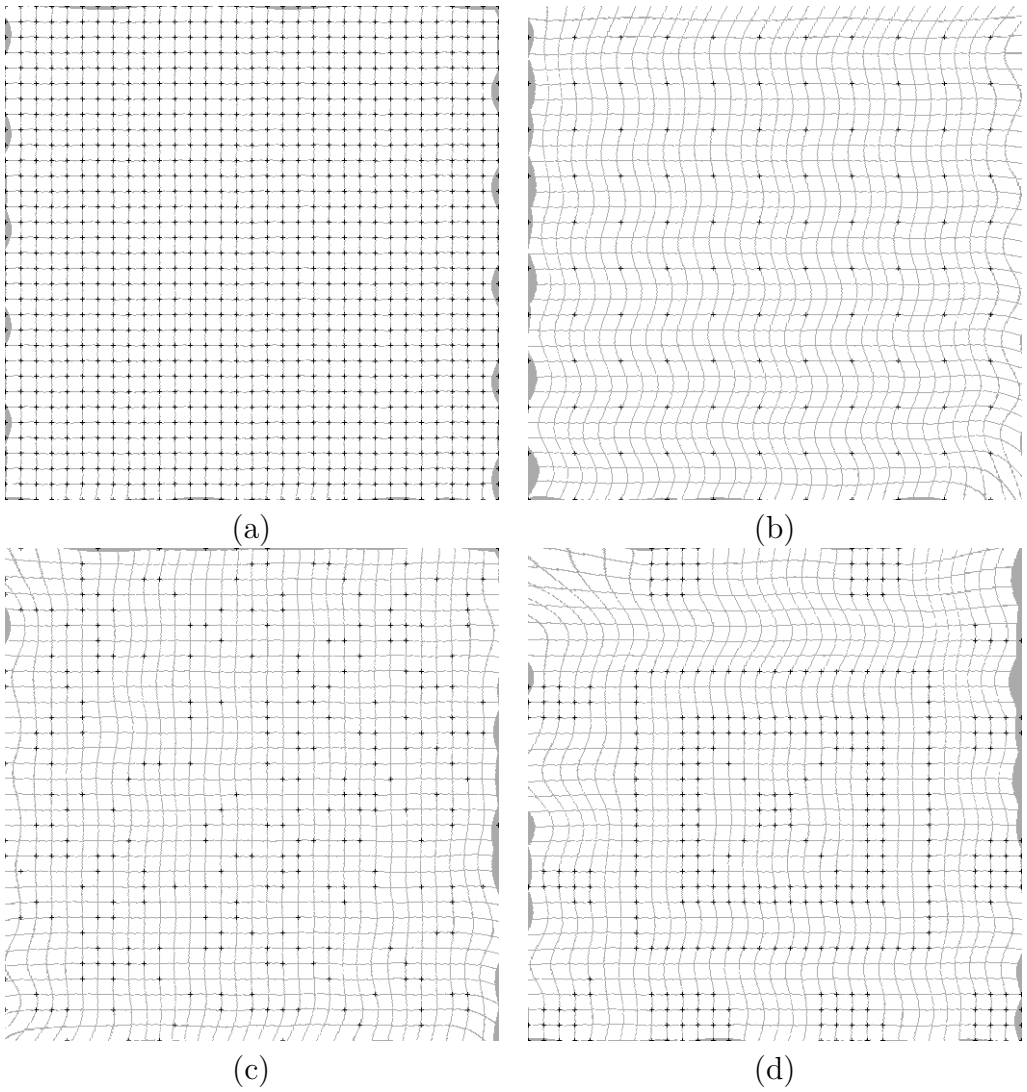


Figure 4: Registration results using multiquadrics as the transformation function. (a)–(d) Resampling of Fig. 1f to overlay Fig. 1a using the control points shown in Figs. 2a–d.

Table 2: Registration accuracy of MQ using the control points shown in Fig. 2 and deformations shown in Figs. 1b–f when compared to Fig. 1a. Numbers inside parentheses show errors when optimal d^2 was used, while numbers not in parentheses are errors obtained with $d^2 = 0$.

		Translation	**Rotation**	**Scaling**	**Linear**	**Nonlinear**
Uniform	MAX	0.0	0.0	0.0	0.0	0.0
(Dense)	RMS	0.0	0.0	0.0	0.0	0.0
Uniform	MAX	12.6 (1.4)	36.7 (6.0)	36.8 (2.8)	22.4 (5.1)	35.4 (29.2))
(Sparse)	RMS	1.5 (0.2)	3.6 (0.9)	3.2 (0.6)	3.3 (1.0)	5.6 (5.5)
Random	MAX	33.5 (3.2)	33.4 (14.0)	21.6 (6.0)	25.3 (8.0)	66.1 (23.1)
	RMS	3.0 (0.2)	3.3 (2.1)	1.7 (1.3)	3.4 (1.4)	7.7 (2.9)
Irregular	MAX	74.2 (46.8)	63.8 (36.7)	65.5 (37.6)	25.1 (11.3)	70.2 (41.3)
	RMS	8.4 (4.4)	7.2 (3.5)	7.1 (3.4)	3.7 (1.9)	9.6 (2.3)

instance, function

$$\phi(r_i) = \begin{cases} (r_i - a)^2, & 0 \le r_i \le 1, \\ 0, & r_i > 1, \end{cases} \tag{50}$$

where $r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}$ is not only 0 at distance $r_i = a$ from (x_i, y_i), but its tangents are also 0. Therefore, the basis functions smoothly vanish at distance $r_i = a$ from their centers, and a weighted sum of them will be a smooth function everywhere in the approximation domain. Parameter a should be selected in such a way that within each region of radius a, at least a few control points are obtained. Parameters $\{F_i : i = 1, \ldots, N\}$ are again determined by solving a system of linear equations. Note that although the basis functions have local support, a global system of equations have to be solved. The basis functions work like weights and a surface point is obtained from a weighted sum of a number of data points surrounding the surface point. Since the sum of the weights varies across the image domain registration accuracy will greatly depend on the organization of the control points.

A comparison of globally defined radial basis functions and compactly supported radial basis functions in image registration is provided by Fornefett et al. [20]. Improved accuracy is reported for compactly supported radial basis functions when compared to globally defined

radial basis functions in the registration of intra-operative brain images.

Because radial basis functions are radially symmetric, when control points are irregularly spaced, they cannot model the geometric difference between the images well. Registration accuracy depends on the organization of control points. Basis functions that can adapt to the organization of points will be more appropriate for the registration of images with irregularly spaced control points. Such basis functions are discussed next.

6 Approximation Methods

Radial basis functions, including thin-plate splines, are interpolating methods. They map corresponding control points in the images exactly to each other. Methods that map corresponding control points approximately to each other are called approximation methods. Approximation methods are defined by a weighted sum of the control points, with the weights having a sum of 1 everywhere in the approximation domain. The function approximating a set of points as defined by (8) is given by:

$$f(x,y) = \sum_{i=1}^{N} f_i b_i(x,y) \tag{51}$$

where

$$b_i(x,y) = \frac{R_i(x,y)}{\sum_{i=1}^{N} R_i(x,y)} \tag{52}$$

and $R_i(x,y)$ is a monotonically decreasing radial basis function.

When $R_i(x,y)$ is a Gaussian,

$$R_i(x,y) = \exp\left\{-\frac{(x-x_i)^2 + (y-y_i)^2}{2\sigma_i^2}\right\}, \tag{53}$$

the weights are called rational Gaussians. Although Gaussians are symmetric, rational Gaussians are not symmetric because they have to ensure that the sum of the weights everywhere in the approximation domain is 1 [31]. This property makes the weight functions stretch toward large gaps. Use of rational Gaussians in the correction of the local geometry of images has been demonstrated by Jackoswski, et al. [39].

When $R_i(x, y)$ is an inverse multiquadric function, we have

$$R_i(x, y) = [(x - x_i)^2 + (y - y_i)^2 + d_i^2]^{-1/2}. \tag{54}$$

As d_i approaches 0, $f(x_i, y_i)$ gets closer to f_i. When $d_i = 0$ for all i, the function will interpolate the points. When Gaussians are used, as the standard deviations are reduced, the same effect is observed and at the limit when $\sigma_i = 0$ for all i, the function will interpolate the points. At very small d_i's or σ_i's, flat spots will appear in function f at and surrounding the control points. This is demonstrated in a simple example in Fig. 5. Consider the following five data points: $(0, 0, 0)$, $(1, 0, 1)$, $(1, 1, 1)$, $(0, 1, 0)$, and $(0.5, 0.5, 0.5)$. The single-valued rational Gaussian surface approximating the points with increasing σ_i's are shown in Figs. 5a–d. For small σ_i's, flat horizontal spots are obtained around the data points. This means, for large variations in x and y, only small variations will be obtained in f near the data points. The implication of this is that when such a function is used to represent a component of a transformation function, points uniformly spaced in the reference image will densely map to points near the control points in the sensed image.

Consider the following 5 correspondences:

$$[(0, 0)(0, 0)]; [(1, 0)(1, 0)]; [(1, 1)(1, 1)]; [(0, 1)(0, 1)]; [(0.5, 0.5)(0.5, 0.5)]. \tag{55}$$

The transformations that map points in the reference image to points in the sensed image for increasing σ_i's are depicted in Figs. 5e–h. The images show points in the sensed image mapping to uniformly spaced points in the reference image. As σ_i's are reduced, the density of points increase near the control points and the function more closely approximates the control points. As σ_i's are increased, spacing between the points becomes more uniform, but the function moves away more from the control points, increasing the approximation error. Ideally, we want a transformation function that it will map corresponding control points more closely when σ_i's are decreased without increasing the density of the points near the control points. Some variation in the density of the points is inevitable if the images to be registered have nonlinear geometric differences. Nonlinear geometric differences change the

local density of points in order to stretch some parts of the sensed image while shrinking others as needed to make its geometry resemble that of the reference image.

In order to obtain rather uniformly spaced points in the sensed image when uniformly spaced points are selected in the reference image, parametric surfaces should be used. Using the corresponding control points given in (1), we first compute a parametric surface with components $x = x(u, v), y = y(u, v)$, and $f = f(u, v)$ approximating points

$$\{(x_i, y_i, f_i) : i = 1, \ldots, N\}. \tag{56}$$

with nodes $\{(u_i = x_i, v_i = y_i) : i = 1, \ldots, N\}$. Given pixel coordinates (x, y), in the reference image we will then determine corresponding parameter coordinates (u, v) from $x = x(u, v)$ and $y = y(u, v)$. This requires the solution of two nonlinear equations. Then, knowing (u, v), we evaluate $f = f(u, v)$. The obtained value will, in effect, represent the X or the Y coordinate of the point corresponding to point (x, y). Figs. 5i–l show the resampling result achieved in this manner. As can be observed, this transformation function now maps uniform points in the reference image to almost uniform points in the sensed image while still mapping corresponding control points approximately to each other.

To determine the accuracy of approximation methods in image registration, the control points shown in Fig. 2 and the deformations shown in Fig. 1 were used. MAX and RMS errors for rational Gaussian weights are shown in Table 3. Registration errors vary with the standard deviations of Gaussians, which control the widths of the weight functions. As the standard deviations of Gaussians are increased, the transformation function becomes smoother but errors in registration increase. As the standard deviations of Gaussians are decreased, the transformation function is less smooth, producing sharp edges and corners, but it maps control points to each other more closely. By decreasing the standard deviations of Gaussians, corresponding control points are mapped to each other more closely, but other points in the images may not accurately map to each other. There is a set of standard deviations that produces the least error for a given pair of images. In general, if the geometric difference between images varies sharply, smaller standard deviations should be used than

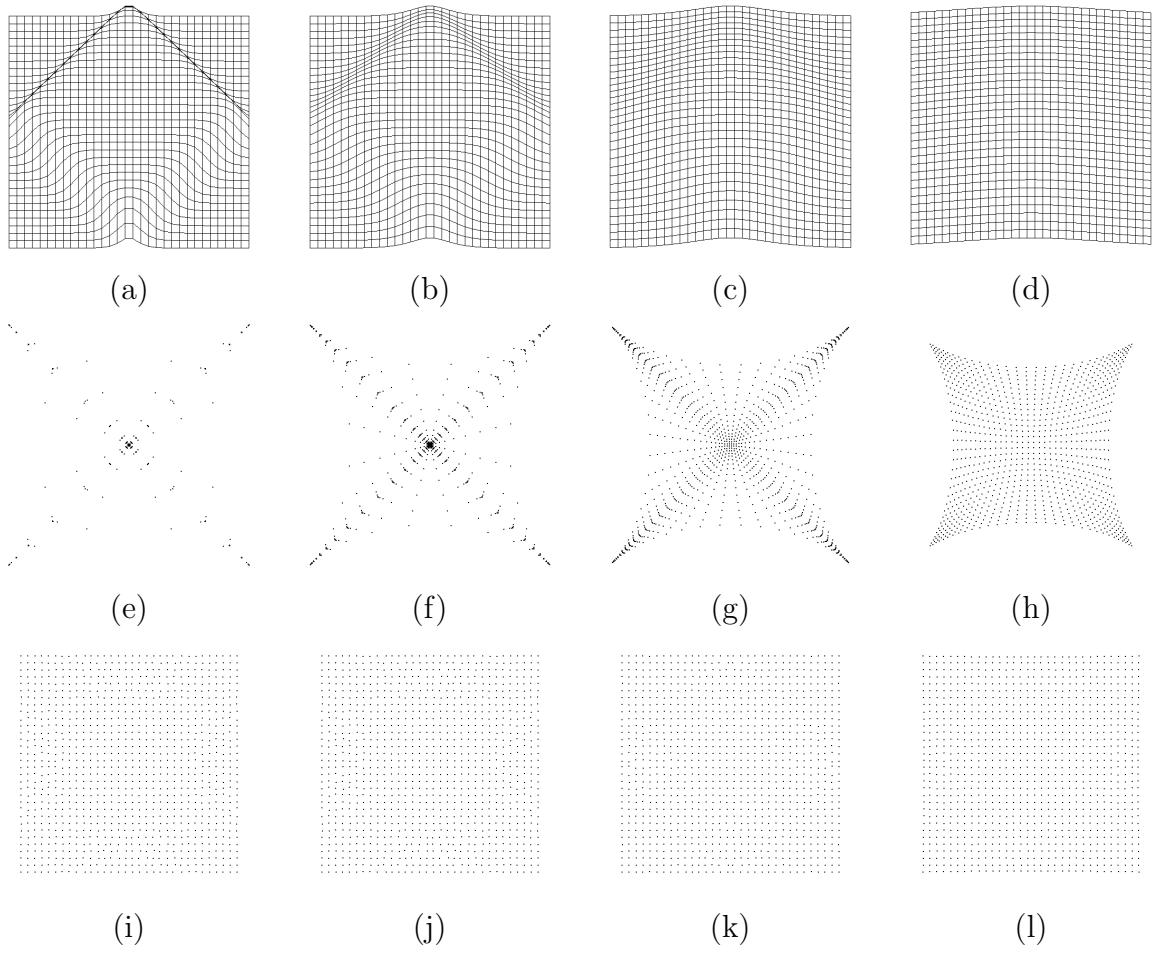


Figure 5: (a)–(d) A component of a transformation represented by a rational Gaussian surface using 5 data points, $\sigma_i = \sigma$ for $i = 1, \ldots, N$, at increasing values of σ. (e)–(h) A transformation function with components represented by single-valued rational Gaussian surfaces at increasing values of σ. Shown here are points in the sensed image mapping to uniformly spaced points in the reference image. (i)–(l) The same as in (e)–(h) but using parametric surfaces.

when the geometric difference between images varies smoothly.

The standard deviations of Gaussians are set inversely proportional to the density of the control points. This will produce narrow Gaussians where the density of points is high and wide Gaussians where the density of points is low, covering large gaps between the control points as necessary. The process can be made totally automatic by setting σ_i equal to the radius r of a circular region surrounding the ith control in the reference image that contains $n-1$ other control points ($n \geq 2$). n can be considered a smoothness parameter. The larger its value, the smoother the obtained transformation will be. This is because as n is increased the widths of Gaussians increase and as n is decreased the widths of Gaussians decrease, more closely reproducing local geometric differences between the images.

Results in Table 3 shown in parentheses were obtained using $n = 5$, and results without parentheses were obtained for $n = 9$. Both MAX and RMS errors are smaller than those shown in Tables 1 and 2, except for errors in the first row. Since registration is done by approximation, corresponding control points do not fall on top of each other exactly so the errors are not zero; however, they are very small. Examples of image registration by approximation using rational Gaussian weights are shown in Fig. 6. The irregular boundaries of the resampled images are due to the fact that rational Gaussian weights approach zero exponentially, and when a control point exists in only one of the images, it cannot be used in the calculation of the transformation, so resampling is not possible near that control point. This is in contrast to monotonically increasing radial basis functions such as TPS and MQ where resampling is possible outside the image domain. The approximation method with rational Gaussian weights, however, has produced more accurate results than TPS and MQ in areas where resampling was possible.

Experiments show that when $n = 2$ or σ_i is equal to the distance of the ith control point to the control point closest to it, local deformations, including noise in the correspondences, are reproduced. As n is increased, noise in the correspondences reduces, producing a smoother resampling. Increasing n too much, however, makes it impossible to reproduce sharp changes

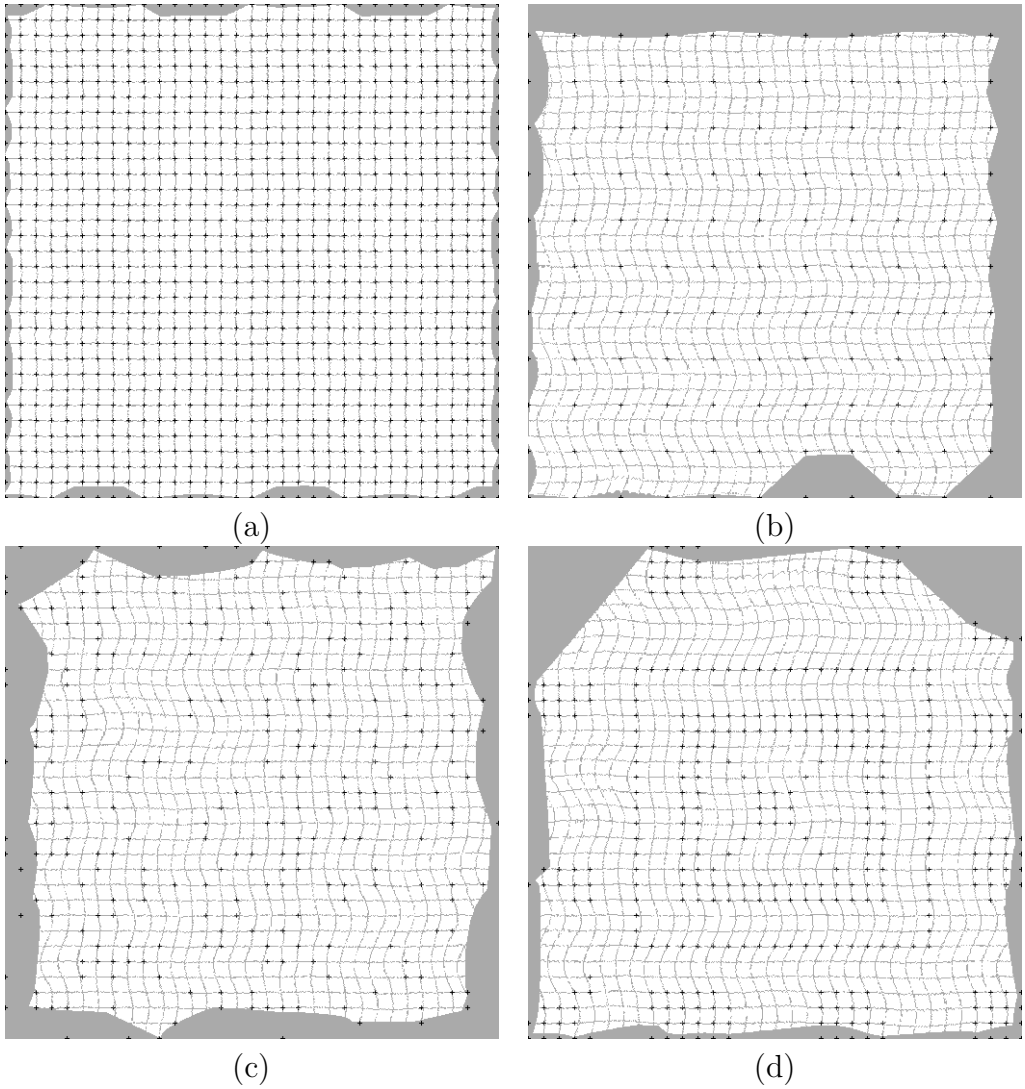


Figure 6: (a)–(d) Resampling of Fig. 1f to overlay Fig. 1a using rational Gaussian weights and the control points shown in Figs. 2a–d. The control points in the reference image and the resampled sensed image are overlaid to show the quality of registration.

Table 3: Registration accuracy of the rational Gaussian transformation when using the control points in Fig. 2 and the geometric deformations shown in Fig. 1. The errors are for $n = 9$. When reducing n to 5, the errors inside the parentheses are obtained.

		Translation	**Rotation**	**Scaling**	**Linear**	**Nonlinear**
Uniform	MAX	0.0 (0.0)	1.0 (0.0)	0.0 (0.0)	1.0 (1.0)	2.2 (1.4)
(Dense)	RMS	0.0 (0.0)	0.1 (0.0)	0.0 (0.0)	0.1 (0.1)	1.2 (0.6)
Uniform	MAX	0.0 (0.0)	1.4 (0.0)	1.0 (0.0)	1.4 (1.0)	9.0 (7.1))
(Sparse)	RMS	0.0 (0.0)	0.3 (0.0)	0.1 (0.0)	0.3 (0.1)	4.0 (1.2)
Random	MAX	0.0 (0.0)	1.4 (1.4)	1.4 (1.4)	1.4 (1.4)	10.2 (9.0)
	RMS	0.0 (0.0)	0.6 (0.4)	0.6 (0.4)	0.6 (0.3)	3.7 (1.3)
Irregular	MAX	0.0 (0.0)	1.4 (1.4)	1.4 (1.4)	1.4 (1.4)	15.1 (15.0)
	RMS	0.0 (0.0)	0.5 (0.2)	0.6 (0.3)	0.5 (0.2)	4.0 (1.8)

in geometric differences between images. n should, therefore, be selected using information about geometric differences between the images as well as the density of the points. If no information about geometric differences between the images is provided, information about the density of control points should be used to select n. Since feature selection methods usually find more control points at and near region boundaries, this method can accurately align region boundaries.

Formula (51) defines a component of a transformation function in terms of a weighted sum of a component of the control points in the sensed image. Computationally, this weighted mean approach is more stable than TPS and MQ because it does not require the solution of a system of equations; therefore, it can handle a very large number of control points in any arrangement. When TPS or MQ is used, the system of equations to be solved may become unstable or singular for a certain arrangement of points.

Methods to make weighted means local have been proposed by Maude [46] and McLain [48], using rational weights of the form given in (52) but with

$$R_i(x,y) = \begin{cases} 1 - 3r_i^2 + 2r_i^3, & 0 \le r_i \le 1, \\ 0, & r_i > 1, \end{cases} \tag{57}$$

where

$$r_i = [(x - x_i)^2 + (y - y_i)^2]^{\frac{1}{2}} / r_n \tag{58}$$

and r_n is the distance of control point (x_i, y_i) to its $(n-1)$st closest control point in the reference image. Note that since

$$\left[\frac{dR_i(x,y)}{dr_i}\right]_{r_i=1} = 0 \tag{59}$$

not only $R_i(x,y)$ vanishes at $r_i = 1$, it vanishes smoothly. Therefore, $f(x,y)$ will be continuous and smooth everywhere in the approximation domain. Knowing the weight function at a control point, the coefficients of a polynomial are then found to interpolate that point and $(n-1)$ points closest to it. The weighted sum of the polynomial is then used to represent a component of the transformation. Therefore, the formula for a local weighted mean is

$$f(x,y) = \frac{\sum_{i=1}^{N} R_i(x,y) p_i(x,y)}{\sum_{i=1}^{N} R_i(x,y)} \tag{60}$$

where $p_i(x,y)$ is the polynomial interpolating data at (x_i, y_i) and those at $(n-1)$ of its closest points. Note that this requires the solution of a small system of equations to find each local polynomial. The polynomials encode information about geometric differences between the images in small neighborhoods. Therefore, the method is suitable for the registration of images where a large number of control points is available and sharp geometric differences exist between the images. When the density of control points is rather uniform, local polynomials of degree two will suffice. However, for very irregularly spaced control points, a polynomial of degree two may create holes in the transformation function in areas where large gaps exist between the control points, and polynomials of higher degrees are needed to widen the local functions to cover the holes. Polynomials of high degrees, however, are known to produce fluctuations away from the interpolating points. Therefore, when the density of control points varies across the image domain, the local weighted mean may not be a suitable transformation for registration of the images and the global weighted mean is preferred. A comparison of local and global approximation methods in image registration is provided in [29].

The process of finding transformation functions for registration of 3-D images is the same as that for 2-D images except that in the equations, $[(x-x_i)^2 + (y-y_i)^2]$ should be replaced

with $[(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2]$.

7 Piecewise Methods

If control points in the reference image are triangulated, by knowing the correspondence between control points in sensed and reference images, corresponding triangles in the sensed image can be obtained. Piecewise methods are those that map areas within corresponding triangles to each other. If a linear transformation function is used to map a triangle in the sensed image to the corresponding triangle in the reference image, the transformation becomes piecewise linear. The transformation will be continuous, but it will not be smooth. When the regions are small or local geometric difference between the images is small, piecewise linear may be sufficient. However, if local deformations are large, tangents at the two sides of a triangle edge may be quite different, resulting in an inaccurate registration.

The choice of triangulation will affect the registration result. As a general rule, elongated triangles should be avoided and preference should be given to triangles without acute angles. Algorithms that maximize the minimum angle in triangles is know as Delaunay triangulation [32, 42]. Better approximation accuracy will be achieved if 3-D points obtained from control points in the reference image and the X or the Y coordinate of corresponding control points in the sensed image are used to obtain the triangles [5, 63].

In order to provide the same tangent at both sides of a triangle edge, corresponding triangular regions in the images should be mapped to each other by polynomials of degrees two or higher and the gradients of the polynomials in the direction normal to the triangle edge being shared should be the same. Various methods for fitting piecewise smooth surfaces over triangles have been proposed [10, 12, 13, 61, 65].

Using the coordinates of control points in the reference image and the X or the Y coordinate of the corresponding control point in the sensed image, 3-D triangle meshes are obtained. Fast subdivision techniques have been developed as a means to efficiently fit piecewise smooth surfaces to triangle meshes. Subdivision methods use corner-cutting

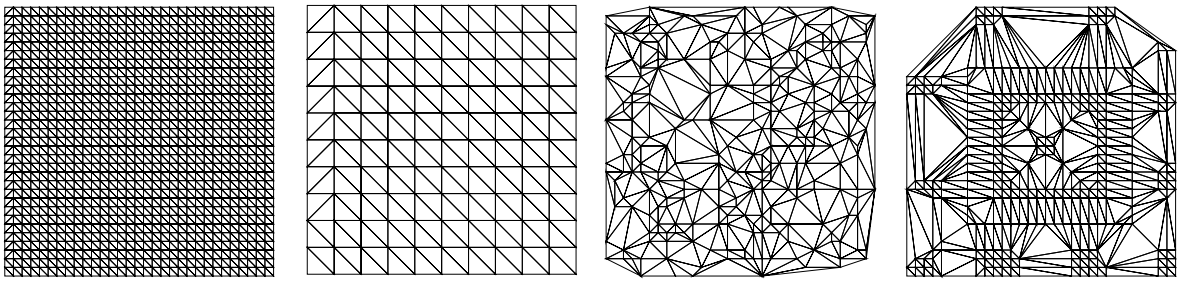
rules to produce a limit smooth surface by recursively cutting off corners in the polyhedron obtained from the 3-D triangles [47, 50, 66]. Subdivision surfaces contain B-spline, piecewise Bézier, and non-uniform B-spline (NURBS) as special cases [62]. Therefore, each component of a transformation can be considered a B-spline, a piecewise Bézier, or a NURBS surface. These surfaces are very suitable for representation of transformation functions for nonrigid image registration because a local deformation or inaccuracy in the correspondences is kept local.

Piecewise linear [27] and piecewise cubic [28] transformation functions have been used to register images with nonlinear geometric difference. The methods are efficient and are easy to implement. However, good accuracy is achieved only within the convex hull of the control points where triangles are obtained. For areas outside the convex hull of the control points, extrapolation has to be used, which often results in large errors, especially when higher order polynomials are used to represent the patches.

Table 4 shows the results of mapping Figs. 1b–f to Fig. 1a with control points shown in Fig. 2 using piecewise linear transformations. Errors are slightly worse than those obtained by the approximation method using rational Gaussian weights, but they are much better than those of radial basis functions, including TPS. To obtain these results, the control points in the reference image were triangulated by Delaunay triangulation as shown in Fig. 7. Then corresponding triangles were obtained in the sensed image and a linear transformation function was found to map a triangle in the sensed image to the corresponding triangle in the reference image. Fig. 8 shows resampling of Fig. 1f to overlay Fig. 1a using the control points shown in Fig. 2.

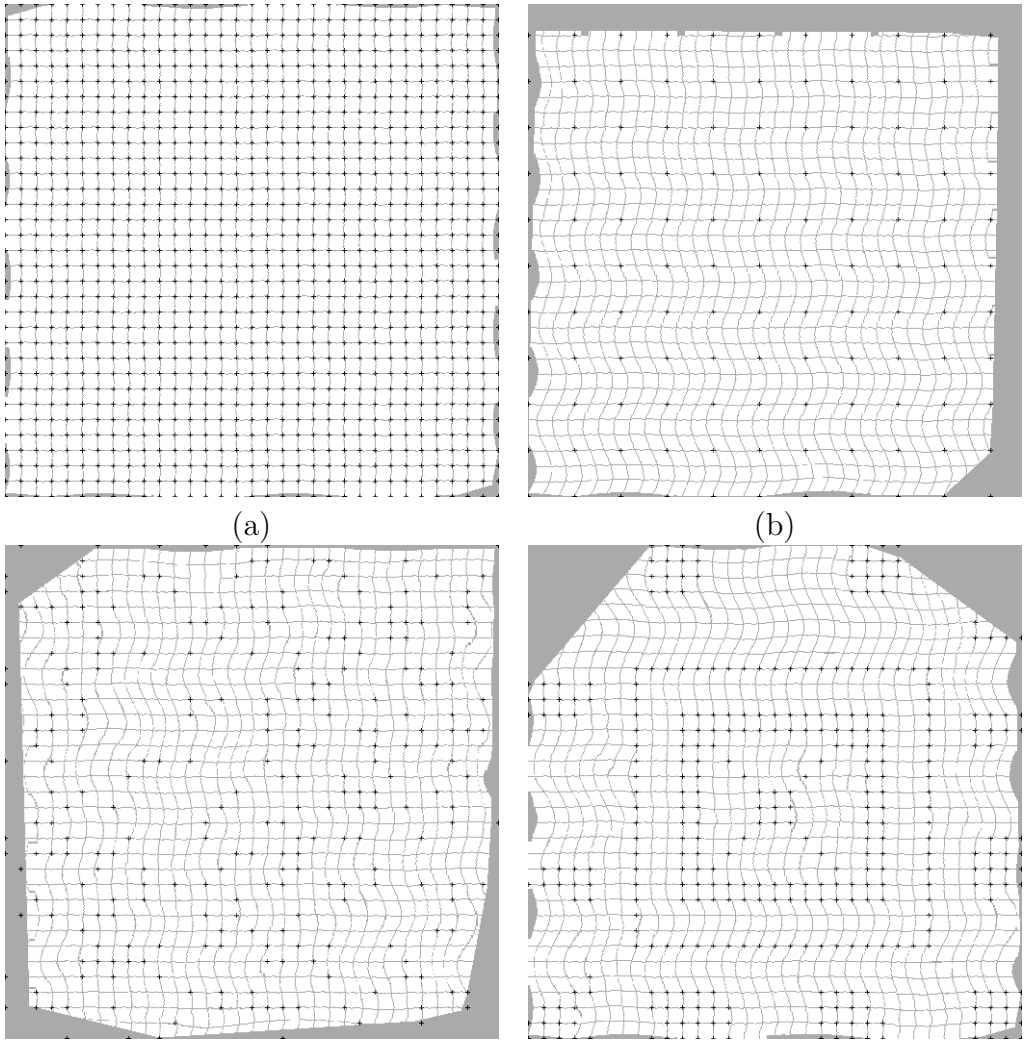
8 Piecewise Approximation Methods

Piecewise methods are attractive because they keep local deformations and errors local and determine a functional value from a small subset of the control points. The piecewise method described in the preceding section triangulates the control points, and thus individual re-



(a) (b) (c) (d)

Figure 7: (a)–(d) Triangulation of control points in Figs. 2a–d, respectively.



(a) (b)

(c) (d)

Figure 8: (a)–(d) Piecewise linear resampling of image 1f to align with image 1a using the control points shown in Figs. 2a–d, respectively.

Table 4: Registration accuracy of the piecewise linear transformation using the control points shown in Fig. 2 and the deformations shown in Fig. 1b–f when compared to Fig. 1a.

		Translation	Rotation	Scaling	Linear	Nonlinear
Uniform	MAX	0.0	0.0	0.0	0.0	0.0
(Dense)	RMS	0.0	0.0	0.0	0.0	0.0
Uniform	MAX	0.0	1.4	1.4	1.4	15.0
(Sparse)	RMS	0.0	0.5	0.5	0.5	4.0
Random	MAX	0.0	1.4	1.4	1.4	16.1
	RMS	0.0	0.6	0.6	0.6	4.0
Irregular	MAX	0.0	1.4	1.4	1.4	16.0
	RMS	0.0	0.5	0.4	0.6	5.8

gions used in the registration are triangular. Unless the triangles are very small, their use in registration of free-form regions is limited. We would like a piecewise method that can map complex regions in the images to each other. This is necessary when regions of different elasticities/rigidities in the images deform differently. A single transformation cannot accurately model all local geometric differences between two images. To ensure that a smooth mapping is obtained across regions, we require that at the boundary between two adjacent regions not only the transformation functions that register the regions produce the same value, but that they produc the same tangents.

We would like a transformation where it not only maps corresponding control points to each other, but it maps tangents at corresponding control points to each other also. Transformation functions based on thin-plate splines have been proposed that enforce desired tangents at corresponding control points [45, 57]. However, because thin-plate splines use radially symmetric basis functions, when the control points are irregularly spaced, large errors are obtained away from the control points. Therefore, use of thinn-plate splines in limited to images containing sparse and rather uniformly spaced control points. The weighted averaging method described here uses weight functions that adapt to the density and organization of the points. Therefore, it can register images that contain dense control points on regions boundaries and sparse control points elsewhere.

Let's suppose the parametric equation of the tangent at control point (x_i, y_i) in the reference image is

$$x_i(t) = a_x t + b_x, \tag{61}$$

$$y_i(t) = a_y t + b_y. \tag{62}$$

a_x and a_y are gradients of the tangent with respect to x and y axes, respectively, and b_x and b_y are determined such that the line passes through point (x_i, y_i) at $t = t_i$.

Similarly, let's suppose the parametric equation of the corresponding tangent in the sensed image is

$$X_i(s) = A_x s + B_x, \tag{63}$$

$$Y_i(s) = A_y s + B_y. \tag{64}$$

The transformation that maps $[x_i(t), y_i(t)]$ to $[X_i(s), Y_i(s)]$ has two components. One component will pass through point $[x_i(t_i), y_i(t_i), X(s_i)]$ and have gradients with respect to x, y, X axes equal to a_x, a_y, A_x, and another will pass through $[x_i(t_i), y_i(t_i), Y_i(s_i)]$ and have gradients a_x, a_y, A_y. t_i is the parameter at which $x_i(t_i) = x_i$ and $y_i(t_i) = y_i$, and s_i is the parameter at which $X_i(s_i) = X_i$ and $Y_i(s_i) = Y_i$. Note that the weighted sum of 3-D points used to describe a component of a transformation function in an approximation method can be considered a weighted sum of planes that pass through (x_i, y_i, X_i) and (x_i, y_i, Y_i) and are parallel to the xy plane ($a_x = a_y = 0$).

A component of the transformation that maps corresponding control points as well as corresponding tangents to each other can be written as

$$f(x, y) = \sum_{i=1}^{N} f_i(x, y) b_i(x, y), \tag{65}$$

where $b_i(x, y)$ is the ith weight function as defined by (52), and $f_i(x, y)$ represents the plane passing through (x_i, y_i, f_i) and having gradients (a_x, a_y, f'_i). In order to parametrize all planes together, it is necessary to set (t_i, s_i) proportional to (x_i, y_i). In the following, we let $t_i = x_i$ and $s_i = y_i$ to simplify the computations and make the surface a function of x and

y. f_i is either X_i or Y_i and f'_i is either A_x or A_y depending on whether the X or the Y component of the transformation is being computed.

Given the coordinates of corresponding control points $\{(x_i, y_i), (X_i, Y_i) : i = 1, \ldots, N\}$ and corresponding tangents $\{(a_x, a_y), (A_x, A_y) : i = 1, \ldots, N\}$, we first determine two sets of parametric planes that pass through points $\{(x_i, y_i, X_i) : i = 1, \ldots, N\}$ and $\{(x_i, y_i, Y_i) : i = 1, \ldots, N\}$ and have gradients $\{(a_x, a_y, A_x) : i = 1, \ldots, N\}$ and $\{(a_x, a_y, A_y) : i = 1, \ldots, N\}$, respectively. Then, a weighted sum of each set of planes as described by equation (65) is computed to obtain a component of the transformation. The process in 3-D is the same except that the planes become hyperplanes, passing through points $\{(x_i, y_i, z_i, X_i) : i = 1, \ldots, N\}$, $\{(x_i, y_i, z_i, Y_i) : i = 1, \ldots, N\}$, and $\{(x_i, y_i, z_i, Z_i) : i = 1, \ldots, N\}$ and having gradients $\{(a_x, a_y, a_z, A_x) : i = 1, \ldots, N\}$, $\{(a_x, a_y, z_z, A_y) : i = 1, \ldots, N\}$, and $\{(a_x, a_y, z_z, A_z) : i = 1, \ldots, N\}$, respectively.

If a volume is composed of materials of different elastic properties, geometric difference between two images of the volume may vary from point to point. In medical images this is often observed due to the presence of tissues with different elastic properties. Consider two images obtained of a bone and surrounding soft tissues. The transformation that registers the images should register the bone rigidly while registering the soft tissues nonrigidly. Although soft tissues deform, closer to the bone the deformation is smaller than at points farther away. We would like to register the images in such a way that not only the points on the bones align but the tangents at the bones align also. This will ensure that tissue deformations next to the bone will be minimal, although tissue deformations away from the bone could be large.

Two adjacent tissues with different elastic properties require a transformation that can take into consideration the elastic properties of the regions. In a piecewise approach, different transformations can be used to register different regions in an image. Rigidity can be incorporated into a transformation by controlling the smoothness of the transformation. When approximation methods are used, the widths of the weight functions determine the

smoothness or rigidity. The smaller the widths of weight functions in a neighborhood, the better the transformation will reproduce the local geometric difference between the images, making it suitable for the registration of more elastic regions. As the widths of weight functions are increased, the obtained transformation will capture less local geometric difference between images and is more suitable for registration of less elastic regions.

Since a sharp transition in the transformation function cannot be reproduced effectively with a single transformation, if rigidity varies sharply, piecewise transformations, each reflecting rigidity in a region, are needed to accurately register the images. This requires that the images be segmented into regions so that pixels or voxels within each region are known. Then, a transformation is obtained for each region, and to ensure that transformations at two sides of a region boundary join smoothly, the same tangent is required at corresponding control points. Note that this requires that correspondence between some or all pixels on corresponding region boundaries are somehow determined.

In images where segmentation is not possible or tangents at corresponding control points are not known, the planes to be used in the transformation cannot be determined. However, one may fit a plane to each point (x_i, y_i, f_i) and a few of its neighbors and use that as the plane in the equation of a component of a transformation. The planes decode the geometric difference between corresponding local neighborhoods in the images. Therefore, information about the geometric difference between small neighborhoods surrounding corresponding control points is used to register the images.

Examples of image registration in this manner are shown in Fig. 9. Figs. 9a–d show the transformation of Fig. 1f to take the geometry of Fig. 1a using the control points shown in Fig. 2a–d. Results of registering Figs. 1b–f to Fig. 1a using control points of Figs. 2a–d are summarized in Table 5. Compared to the approximation method with the same weight functions, we see that, overall, errors have slightly improved. Rational Gaussian weights are used in these experiments with $n = 9$, and σ_i is set equal to the radius of the smallest circle centered at (x_i, y_i) containing 9 control points.

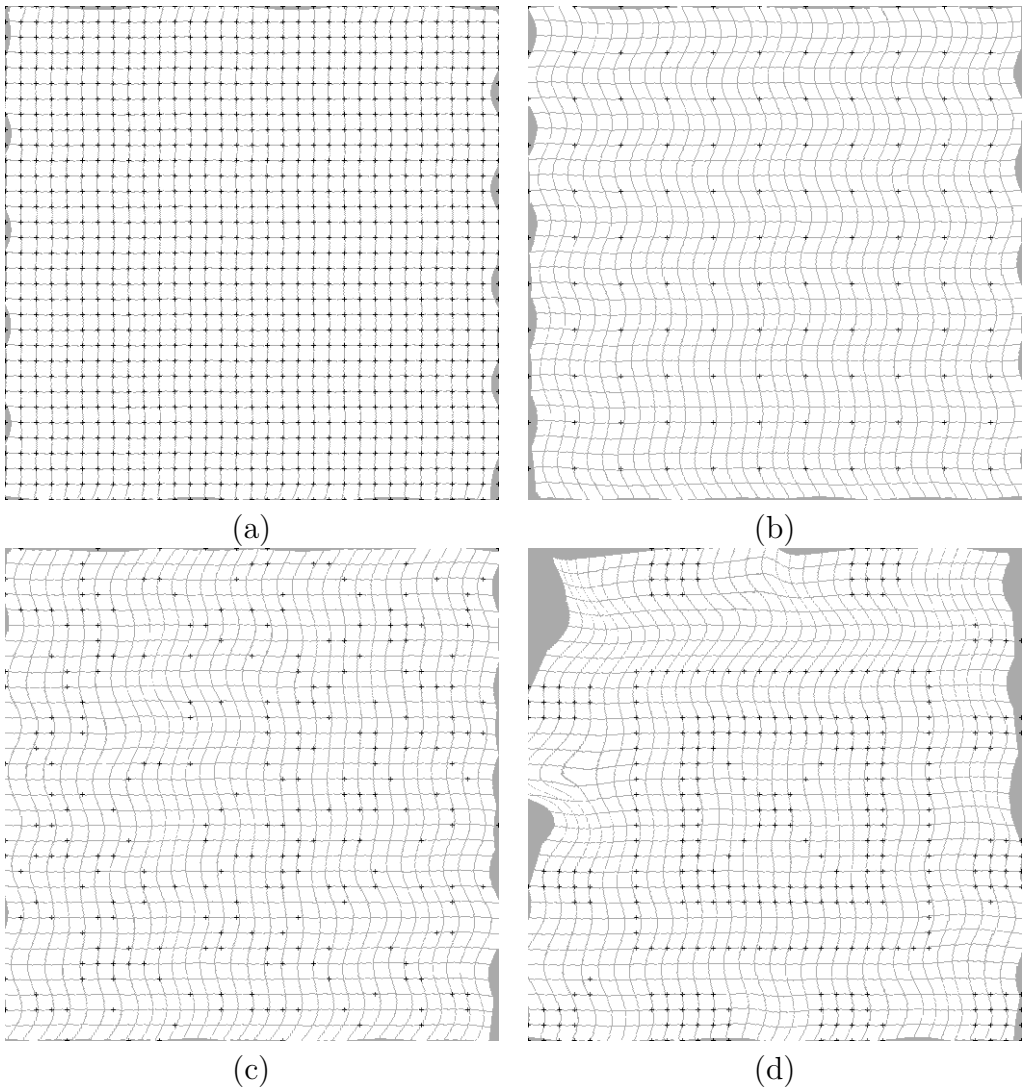


Figure 9: (a)–(d) Resampling of Fig. 1f by piecewise approximation to align with Fig. 1a using the control points shown in Figs. 2a–d.

Table 5: Registration accuracy of the piecewise approximation method using the control points shown in Figs. 2a–d and the deformations shown in Figs. 1b–f when compared to Fig. 1a. The standard deviation of the Gaussian associated with a control point was set equal to the distance of that control point to the 8th closest control point in the reference image.

		Translation	**Rotation**	**Scaling**	**Linear**	**Nonlinear**
Uniform	MAX	0.0	1.0	0.0	1.0	3.6
(Dense)	RMS	0.0	0.1	0.0	0.1	0.7
Uniform	MAX	0.0	1.4	1.4	1.4	14.8
(Sparse)	RMS	0.0	0.5	0.5	0.5	4.4
Random	MAX	0.0	1.4	1.4	1.4	16.1
	RMS	0.0	0.6	0.6	0.6	4.0
Irregular	MAX	0.0	1.4	1.0	1.4	27.0
	RMS	0.0	0.5	0.4	0.6	7.4

If the density of control points varies across the image domain and a fixed number of control points is used to compute the planes, the neighborhood sizes used in the computations will vary from point to point and vary with the density of the control points. Assuming the distance of control point (x_i, y_i) to the $(n-1)$st closest control point is r_i, we have set the standard deviation of the Gaussian associated with the ith control point equal to r_i. The larger the value of n, the smoother (more rigid) the transformation will appear. In Table 5, the standard deviation of Gaussian associated with a control point was set equal to the distance of that point to the 8th closest control point. Neighborhood size should be chosen taking into consideration the noisiness of the control point correspondences and the degree of local geometric differences between images. If the control point correspondences are noisy or if local geometric difference between images is not large, a larger neighborhood size should be used than when the control point correspondences are accurate or when the images have large local geometric differences.

Note that when a weighted sum of planes as opposed to a weighted sum of points is used to define a component of a transformation, flat horizontal spots will not be obtained for the narrow weights, because the planes are no longer horizontal. Therefore, the introduction of

parameters u and v as was necessary in approximation methods is no longer needed. We see that piecewise approximation methods that map corresponding control points as well as their tangents to each other actually have computational advantages over approximation methods that map only corresponding control points to each other.

9 Computational Complexity

Assuming the images being registered are of size $n \times n$ and N corresponding control points are available, the time to register the images consists of the time to compute the transformation and the time to resample the sensed image to the coordinate system of the reference image. In the following, the order of operations needed to compute each transformation discussed above is determined. An operation is considered to be an addition, a subtraction, a multiplication, a division, a comparison, or a small combination of them.

If the correspondences are known to be accurate, only two of the correspondences are sufficient to determine the parameters for a similarity transformation. If the correspondences are known to contain small errors, least-squares should be used to determine the transformation. This requires in the order of N operations. If the correspondences are known to contain large errors, first the inaccurate correspondences should be identified and removed. This can be achieved by clustering the parameters of the transformation using combinations of two point correspondences at a time [68]. Alternatively, combinations of three point correspondences at a time can be used to determine the parameters of linear transformations, and the three correspondences that produce the linear transformation closest to a similarity transformation are selected to determine the transformation parameters [17]. When N is large, since a large number of point pairs or point triples can be obtained, one should use, for example, a subset of the points that fall on the convex hull [25] or the minimum spanning tree [75] of the control points to reduce the computation time. Selection of such control point pairs requires in the order of $N \log N$ operations. Once the translational, rotational, and scaling differences between images are determined, resampling of the sensed image to

the coordinate system of the reference image takes in the order of n^2 operations. Therefore, the computational complexity of the similarity transformation is at least $O(N) + O(n^2)$.

To find the parameters of an affine transformation, coordinates of at least three correspondences are needed. If the correspondences are known to be accurate, three correspondences are sufficient to determine the transformation parameters. If the correspondences contain large errors, combinations of three correspondences should be used to determine the parameters for various affine transformations and by clustering the parameters determining the set of parameters that receives the most support from the point triples [68]. Again, since a large number of point triples can be obtained, points that form consecutive convex hull edges or two connected minimum spanning tree edges of the control points may be used to limit the computations. This requires in the order of $N \log N$ operations. If the inaccuracies are due to quantization noise, least-squares may be used to determine the parameters of the transformation. In this case, computations will be in the order of N. Once the transformation is obtained, it should be evaluated at each pixel in the reference image. Therefore, overall, the computational complexity of affine transformation is at least $O(N) + O(n^2)$. Computational complexity of the projective transformation is similar to that of affine.

To determine a component of the TPS transformation, a system of N linear equations needs to be solved. This requires in the order of N^2 operations. Resampling of each point in the sensed image into the space of the reference image requires use of all N correspondences. Therefore, resampling by TPS requires in the order of n^2N operations. Overall, the computational complexity of TPS is $O(N^2) + O(n^2N)$. The computational complexity of MQ and other monotonically increasing radial basis functions is also $O(N^2) + O(n^2N)$. If compactly supported radial basis functions are used, the process still requires the solution of a system of N linear equations, although many of the entries in the matrix of coefficients will be 0. The time needed to resample a sensed image point to the space of the reference image requires only a subset of the control points in the neighborhood of the point under consideration. Therefore, the computation complexity of compactly supported radial basis

functions in image registration is $O(N^2) + O(n^2)$.

Approximation methods use rational weights with coefficients that are the coordinates of the control points. Therefore, a transformation is immediately obtained without solving a system of equations. The mapping of each point in the sensed image to the space of the reference image takes in the order of N operations because theoretically coordinates of all correspondences should be used to find each resampled point. Therefore, overall, the computational complexity of the method is $O(n^2N)$. In practice, however, since monotonically decreasing functions, such as Gaussians, approach zero abruptly, it is sufficient to use only those control points that are in the immediate vicinity of the point under consideration. Since digital accuracy is sufficient, use of control points farther than a certain distance to the point under consideration will not affect the result. For this to be possible though, the control points should be binned in a 2-D array so that given a point in the sensed image, the control points surrounding it could be determined without examining all the control points. If local weights are used, computation of a resampled point requires use of a very small subset of the control points that are in the neighborhood of the point under consideration. Therefore, computational complexity of local weighted mean or local approximation is $O(n^2)$.

Piecewise methods first require the time to triangulate the control points in the reference image. This requires in the order of $N \log N$ operations. Once the triangles are obtained, there is a need to find a transformation for the corresponding triangles. This requires in the order of N operations. Resampling of each point in the sensed image to the space of the reference image takes a small number of operations. Therefore, overall, the computational complexity of the piecewise linear function $O(N \log N) + O(n^2)$. Piecewise quadratic, cubic, and higher degree functions require a little more time in each operation, but the number of operations to be performed is the same. Therefore, the computational complexity of piecewise methods is $O(N \log N) + O(n^2)$.

Finally, to compute a transformation by piecewise approximation, it is required to com-

Table 6: Computational complexity of various transformation functions when N corresponding control points are available and the images are of size $n \times n$ pixels.

Type of Transformation	**Computational Complexity**
Similarity	$O(N) + O(n^2)$
Affine and Projective	$O(N) + O(n^2)$
Radial Basis Functions (RBF)	$O(N^2) + O(n^2 N)$
Compactly Supported RBF	$O(N^2) + O(n^2)$
Global Approximation	$O(n^2 N)$
Local Approximation	$O(n^2)$
Piecewise Linear	$O(N \log N) + O(n^2)$
Piecewise Approximation (PA)	$O(n^2 N)$
Local PA	$O(n^2) + O(N)$

pute two planes for each control point in the reference image. This requires in the order of N operations. Once the planes are determined, the transformation is immediately obtained. Resampling of each point in the sensed image to the space of the reference image takes in the order of N operations when globally defined rational weights are used. This requires $O(n^2 N)$ operations. Therefore, the overall computational complexity of the piecewise approximation method is $O(n^2 N)$. If rational Gaussians are used as the weights, since Gaussians approach zero exponentially, in practice, it is only necessary to use control points within a small distance of the point under consideration. Therefore, for very small standard deviations, the computational complexity will be $O(n^2) + O(N)$. If rational weights with local support are used, a small subset of the control points that are in the vicinity of the point under consideration need to be used in the computations. Therefore, the computational complexity of the method becomes $O(n^2) + O(N)$.

Table 6 summarizes the computational complexity of the transformation functions discussed in this paper for 2-D images. In 3-D, similar results are obtained with the exception that n^2 is replaced with n^3.

10 What Can We Learn from Transformation Functions?

Transformation functions contain information about geometric differences between images. This information is sometimes crucial in understanding the contents of images. The presence of sharp differences in geometries of two images could be due to the local motion or deformation in the underlying scene and may be significant in interpretation of the scene. Figs. 10a and 10b show the X and Y components of the transformation obtained for the registration example in Fig. 9a. The sinusoidal geometric difference between Figs. 1f and 1a are clearly reflected in the components of the transformation.

If some information about geometric differences between two images is known, that information along with the obtained transformation can be used to identify the inaccurate correspondences. For instance, if the images are known to have only linear geometric differences, $f_x(x, y)$ and $f_y(x, y)$ will be planar. Planes are obtained only when all the correspondences are accurate though. Inaccuracies in the correspondences will result in small dents and bumps in the planes. The geometric difference between Figs. 1a and 1f is sinusoidal as demonstrated in Figs. 10a and 10b. This is observed when all the correspondences are accurate.

In an experiment, we displaced two of the control points in Fig. 2a horizontally, two vertically, and one both horizontally and vertically by 15 pixels. The displacements are reflected in the obtained transformation as depicted in Figs. 10c and 10d. The dark and bright spots in these image are centered at the control points that were displaced. A bright spot shows a positive displacement while a dark spot shows a negative displacement. When displacements are caused by inaccuracies in the correspondences, these small dents and bumps (dark and bright spots) identify the inaccurate correspondences.

Note that horizontal displacements appear only in the X-component of the transformation and vertical displacement appears only in the Y-component of the transformation. An inaccurate correspondence, therefore, could appear in one or both components of a transfor-

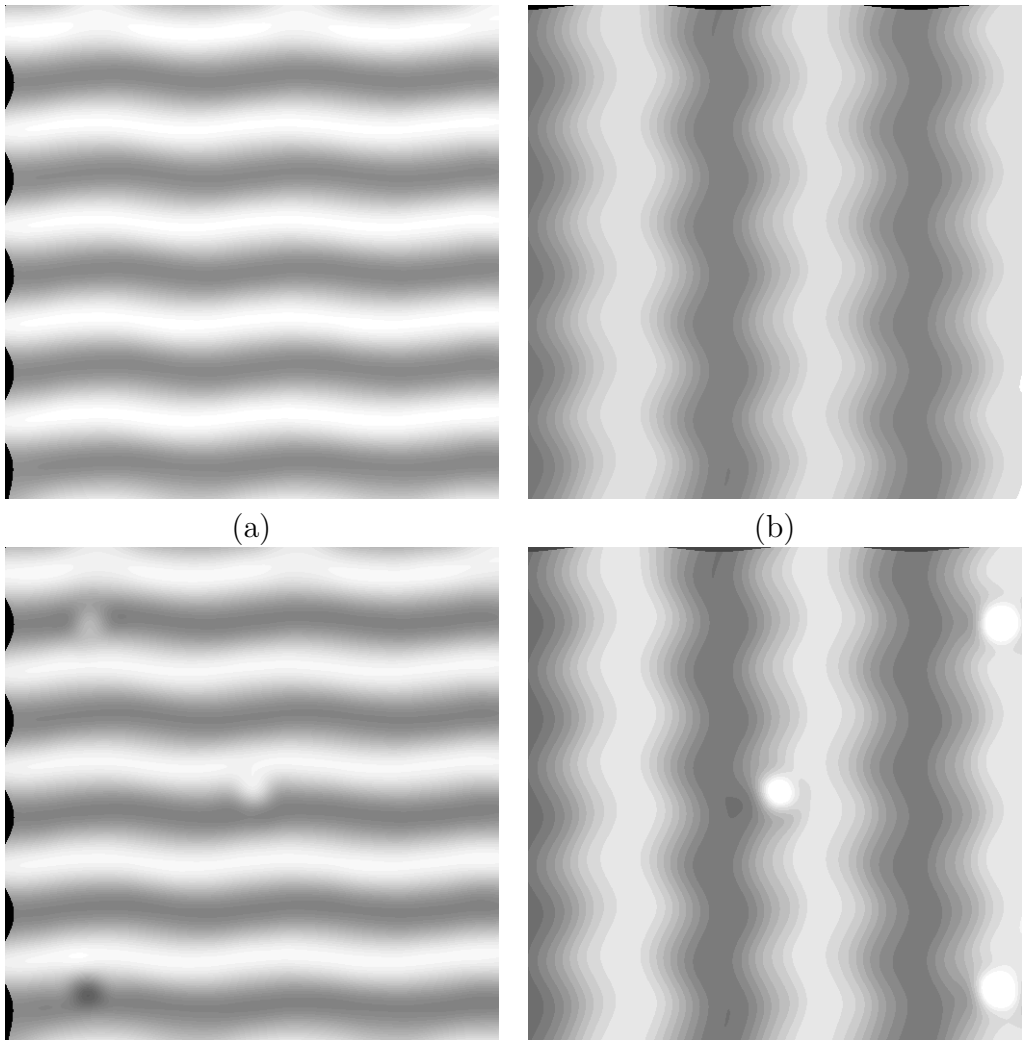


Figure 10: (a), (b) The X and Y components of the transformation for mapping Fig. 1f to Fig. 1a (shown here are actually $f_x(x,y) - x$ and $f_y(x,y) - y$ when mapped to 0–255 to enhance viewing). Brighter points show higher functional values. (c), (d) The X and Y components of the transformation after displacing five of the control points in the sensed image. Errors in the correspondences have resulted in small bumps and dents in surfaces representing the components of the transformation. The bumps and dents are centered at the control points in the reference image that have inaccurate correspondences.

mation. A transformation not only provides information about geometric differences between two images, it contains valuable information about the accuracies of the correspondences, which can be used to identify and remove the inaccurate correspondences. If local geometric differences between two images is known to be small, large gradients in the components of a transformation point to the locations of the inaccurate correspondences.

11 Concluding Remarks

Two factors control the image registration accuracy. They are the accuracy of the control-point correspondences and the accuracy of the transformation function representing geometric differences between images. This paper provided a survey of various transformation functions used in image registration.

Transformation functions for both rigid and nonrigid registration were discussed. If information about the geometric difference between images is available, that information should be used to select the transformation. For instance, if it is known that the images are obtained from a distant platform of a rather flat scene, a linear or a projective transformation is sufficient to register the images. Use of a nonlinear transformation will not only require more computation time, but it may actually worsen the registration accuracy. If no information about the geometric difference between images is available, a transformation function that can adapt to the local geometric differences between the images should be used.

Use of radial basis functions, approximation methods, piecewise methods, and piecewise approximation methods in the registration of images with nonlinear geometric differences was discussed. In a set of experiments on various transformations, it was found that radial basis functions, including TPS, produce the worst results. This is believed to be due to three main reasons. First, the basis functions from which a component of a transformation is determined are radially symmetric and when spacing between the control points is not symmetric, large errors are obtained in areas where the arrangement of control points is nonuniform. Second, because the logarithmic functions used in TPS are monotonically increasing, errors in the

correspondences or local image deformations spread over the entire resampled image. Also, the addition of a new control point to the set of control points requires recomputation of the transformation and resampling of the entire image. Third, a system of equations has to be solved to find each component of a transformation, and when the number of correspondences is very large, this becomes impractical or impossible. If a small number of widely spread control points are available and the local geometric difference between the images is not large, thin plate splines are effective. This has been demonstrated in registration of aerial and satellite images [26].

Piecewise methods are attractive because the effect of a control point is limited to a small neighborhood surrounding it. This property not only makes the computations more stable, it keeps inaccuracies in the correspondences local without spreading them to the entire image domain. Recent subdivision schemes provide efficient means for computing piecewise smooth transformation functions.

Approximation methods are preferred over radial basis functions, including TPS, in non-rigid image registration because of four main reasons. First, they do not require the solution of systems of equations. A component of a transformation is obtained immediately from the given control points. Second, because they do not map corresponding control points exactly to each other, digital errors in the correspondences as well as small mismatch errors are smoothed. Third, the rational weight functions adapt to the density and organization of the points. The weight functions automatically extend to large gaps between control points and their widths increase or decrease with the spacing between the points. This makes it possible to register images where density of control points varies greatly in the image domain. Fourth, the width of all weight functions can be controlled globally to control the smoothness or the rigidity of the transformation.

A class of approximation methods that can ensure desired tangents at corresponding control points was also discussed. These transformations not only map corresponding control points to each other, they also map tangents at corresponding control points to each other.

This is especially useful when registering nonrigid images where the geometric difference between images varies from one region to another.

References

[1] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Trans. Pattern Analysis and Machine Intelligence,* **9**, 5, pp. 698–700, 1987.

[2] P. R. Beaudet, "Rotationally invariant image operators," *Proc. Int. Conf. Pattern Recognition,* pp. 579–583, 1978.

[3] W. Beil, K. Rohr, and H. S. Stiehl, "Investigation of approaches for the localization of anatomical landmarks in 3D medical images," in *Computer Assisted Radiology and Surgery,* H. U. Lemke, M. W. Vannier, and K. Inamura (Eds.), Elsevier Science B. V., pp. 265–270, 1997.

[4] R. Bernstein, "Digital image processing of earth observation sensor data," *IBM J. Research and Development,* pp. 40–67, 1976.

[5] M. Bertram, J. C. Barnes, B. Hamann, K. I. Joy, H. Pottmann, and D. Wushour, "Piecewise optimal triangulation for the approximation of scattered data in the plane," *Computer Aided Geometric Design,* **17**, pp. 767–787, 2000.

[6] F. L. Bookstein, "Principal warps: thin-plate splines and the decomposition of deformations," *IEEE Trans. Pattern Analysis and Machine Intelligence,* **11**, 6, pp. 567–585, 1989.

[7] G. Borgefors, "Hierarchical chamfer matching: A parametric edge matching technique," *IEEE Trans. Pattern Analysis and Machine Intelligence,* **10**, 6, pp. 849–865, 1988.

[8] H. G. Borrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf, "Parametric correspondence and chamfer matching: Two new techniques for image matching," *Proc. Joint Conf. Artificial Intelligence,* pp. 659–663, 1977.

[9] R. E. Carlson, "The parameter R^2 in multiquadric interpolation," *Computers Math. Applic.*, **21**, 9, pp. 29–42, 1991.

[10] L. H. T. Chang and H. B. Said, "A C^2 Triangular patch for the interpolation of functional scattered data," *Computer-Aided Design*, **29**, 6, pp. 407–412, 1997.

[11] L. P. Chew, M. T. Goodrich, D. P. Huttenlocher, K. Kedem, J. M. Kleinberg, and D. Kravets, "Geometric pattern matching under Euclidean motion," *Computational Geometry Theory and Applications,* vol. 7, pp. 113–124, 1977.

[12] C. K. Chui and M.-J. Lai, "Filling polygonal holes using C^1 cubic triangular spline patches," *Computer Aided Geometric Design*, **17**, pp. 297–307, 2000.

[13] P. Constantini and C. Manni, "On a class of polynomial triangular macro-elements," *Computational and Applied Mathematics*, **73**, pp. 45–64, 1996.

[14] A. D. J. Cross and E. R. Hancock, "Graph matching with a dual-step EM algorithm," *IEEE Trans. Pattern Analysis and Machine Intelligence,* **20**, 11, pp. 1236–1253, 1998.

[15] W. A. Davis and S. K. Kenue, "Automatic selection of control points for the registration of digital images," *Proc. Int'l J. Conf. Pattern Recognition*, pp. 936–938, 1978.

[16] L. Ding, A. Goshtasby, and M. Satter, "Volume image registration by template matching," *Image and Vision Computing,* **19**, 12, pp. 821–832, 2001.

[17] L. Ding and A. Goshtasby, "Registration of multi-modal brain images using the rigidity constraint," *2nd IEEE Int'l Sym. Bioinformatics & Bioengineering,* Nov. 4–6, pp. 1–6, 2001.

[18] A. C. Evans, W. Dai, L. Collins, P. Neelin, and S. Marrett, "Warping of a computerized 3-D atlas to match brain image volumes for quantitative neuroanatomical and functional analysis," *SPIE Image Processing,* vol. 1445, pp. 236–247, 1991.

[19] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM,* **24**, pp. 381–395, 1981.

[20] M. Fornefett, K. Rohr, and H. S. Stiehl, "Radial basis functions with compact support for elastic registration of medical images," *Image and Vision Computing,* **19**, pp 87–96, 2001.

[21] W. Förstner, "A feature based correspondence algorithm for image matching," *Int. Arch. Photogramm. Remote Sensing,* **26**, pp. 150–166, 1986.

[22] R. Franke, "Scattered data interpolation: Tests of some methods," *Mathematics of Computation,* vol. 38, no. 157, pp 181–200, 1982.

[23] A. Goshtasby and C. V. Page, "Image matching by a probabilistic relaxation labeling process," *Proc. 7th Int. Conf. Pattern Recognition,* **1**, pp. 307–309., 1984.

[24] A. Goshtasby and G. Stockman, "Point pattern patching using convex hull edges," *IEEE Trans. Systems, Man, and Cybernetics,* **15**, pp. 631–637, 1985.

[25] A. Goshtasby, G. Stockman, and C. Page, "A region-based approach to digital image registration with subpixel accuracy," *IEEE Trans. Geoscience and Remote Sensing,* **24**, 3, pp. 390–399, 1986.

[26] A. Goshtasby, "Registration of image with geometric distortion," *IEEE Trans. Geoscience and Remote Sensing,* **26**, 1, pp. 60–64, 1988.

[27] A. Goshtasby, "Piecewise linear mapping functions for image registration," *Pattern Recognition,* **19**, 6, pp. 459–466, 1986.

[28] A. Goshtasby, "Piecewise cubic mapping functions for image registration," *Pattern Recognition*, **20**, 5, pp. 525–533, 1987.

[29] A. Goshtasby, "Image registration by local approximation methods," *Image and Vision Computing,* **6**, 4, pp. 255–261, 1988.

[30] A. Goshtasby and W. D. O'Neill, "Surface fitting to scattered data by a sum of Gaussians," *Computer-Aided Geometric Design,* **10**, pp. 143–15, 1993.

[31] A. Goshtasby, "Design and recovery of 2-D and 3-D shapes using rational Gaussian curves and surfaces," *Int'l J. Computer Vision,* **10**, 3, pp. 233-256, 1995.

[32] P. J. Green and R. Sibson, "Computing Dirichlet tessellation in the plane," *Computer Journal,* **21**, pp. 168–173, 1978.

[33] R. L. Harder and R. N. Desmarais, "Interpolation using surface splines," *J. Aircraft,* **9**, 2, pp. 189–191, 1972.

[34] C. Harris and M. Stephens, "A combined corner and edge detector," *Proc. 4th Alvey Vision Conf.*, pp. 147–151,1988.

[35] R. L. Hardy, "Multiquadric equations of topography and other irregular surfaces," *Journal of Geophysical Research,* **76**, 8, pp 1905–1915, 1971.

[36] R. L. Hardy, "Theory and applications of the multiquadric-biharmonic method - 20 years of discovery - 1969-1988," *Computers Math. Applic.,* **19**, 8/9, pp 163–208, 1990.

[37] P. Huttenlocher and W. J. Rucklidge, "A multiresolution technique for comparing images using Hausdorff distance," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 705–706, 1993.

[38] M. Irani and P. Raghavan, "Combinatorial and experimental results for randomized point-matching algorithms," *Proc. 12th Annual ACM Symposium Computational Geometry,* pp. 68–77, 1996.

[39] M. Jackowski, A. Goshtasby, S. Bines, D. Roseman, and C. Yu, "Correcting the geometry and color of digital images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* **19**, 10, pp. 1152–1157, 1997.

[40] L. Kitchen and A. Rosenfeld, "Gray-level corner detection," *Pattern Recognition Letters,* **1**, pp. 95–102, 1982.

[41] D. Kozinska, O. J. Tretiak, J. Nissanov, and C. Ozturk, "Multidimensional alignment using the Euclidean distance transform," *Graphical Models and Image Processing,* **59**, 6, pp. 373–387, 1997.

[42] C. L. Lawson, "Software for C^1 surface interpolation," in *Mathematical Software III*, J. R. Rice (Ed.), Academic Press, London, pp. 161–194, 1977.

[43] B. Likar and F. Pernu, "Extraction of corresponding points for the registration of medical images," *Medical Physics,* **26**, 8, pp. 1678–1686, 1999.

[44] C. Lorenz and N. Krahnstver, "Generation of point-based 3D statistical shape models for anatomical objects," *Computer Vision and Image Understanding,* **77**, pp. 175–191, 2000.

[45] K. Mardia, J. T. Kent, C. R. Goodall, and J. Little, "Kriging and splines with derivative information, *Biometrika*, **83**, 1, pp. 207–221, 1996.

[46] A. D. Maude, "Interpolation–mainly for graph plotters," *The Computer Journal,* **16**, 1, pp. 64–65, 1973.

[47] J. Maillot and J. Stam, "A unified subdivision scheme for polygonal modeling," *EUROGRAPHICS*, A. Chalmers and T.-M. Rhyne (Eds.), **20**, 3, 2001.

[48] D. H. McLain, "Two-dimensional interpolation from random data," *The Computer Journal*, **19**, pp. 178–181, 1976.

[49] J. Meinguet, "An Intrinsic approach to multivariate spline interpolation at arbitrary points, in *Polynomial and Spline Approximation,* B. N. Sahney (Ed.), D. Reidel Publishing Company, pp. 163–190, 1979.

[50] J. Peters and U. Reif, "The simplest subdivision scheme for smoothing polyhedra," *ACM Trans. Graphics*, **16**, 4, pp. 420–431, 1997.

[51] M. J. D. Powell, "Radial basis functions for multivariate interpolation: A review," in *Algorithms for Approximation*, J. C. Mason and M. G. Cox (Eds.), Clarendon Press, Oxford, pp. 143–167, 1987.

[52] S. Ranade and A. Rosenfeld, "Point pattern matching by relaxation," *Pattern Recognition,* **12**, pp. 269–275, 1980.

[53] A. Rangarajan, H. Chui, and F. Bookstein,"The softassign Procrustes matching algorithm," in *Information Processing in Medical Imaging,* Springer, pp. 29–42, 1997.

[54] T. J. Rivlin, "Least-squares approximation," *An Introduction to the Approximation of Functions,* Blaisdell Publishing, pp. 48–65, 1969.

[55] K. Rohr, "Localization properties of direct corner detectors," *J. Math. Imaging and Vision,* **4**, pp. 139–150, 1994.

[56] K. Rohr, H. S. Stiehl, R. Sprengel, T. M. Buzug, J. Weese, and M. H. Kuhn, "Landmark-based elastic registration using approximating thin-plate splines," *IEEE Transactions on Medical Imaging,* **20**, 6, pp. 526–534, 2001.

[57] K. Rohr, M. Fornefett, and H. S. Steihl, "Spline-based image registration: Integration of landmark errors and orientation attributes," *Computer Vision and Image Understanding*, **90**, pp. 153–168, 2003.

[58] J. Rucklidge, "Efficiently locating objects using the Hausdorff distance," *Int. J. Computer Vision,* **24**, 3, pp. 251–270, 1997.

[59] L. Schad, S. Lott, F. Schmitt, V. Sturm, and W. J. Lorenz, "Correction of spatial distortion in MR imaging: A prerequisite for accurate stereotaxy," *J. Computer Assisted Tomography,* **11**, 3, pp. 499–505, 1987.

[60] I. P. Schagen, "The use of stochastic processes in interpolation and approximation," *Intern. J. Computer Math.,* **8**, section B, pp. 63–76, 1980.

[61] J. W. Schmidt, "Scattered data interpolation applying regional C^1 splines on refined triangulations," *Math. Mech.*, **80**, 1, pp. 27–33, 2000.

[62] P. Shröder and D. Zorin, "Subdivision for modeling and animation, *SIGGRAPH Course No. 36 Notes*, 1998.

[63] L. L. Schumaker, "Computing optimal triangulations using simulated annealing," *Computer Aided Geometric Design,* **10**, pp 329–345, 1993.

[64] D. Shen and C. Davatzikos, "HAMMER: Hierarchical attribute matching mechanism for elastic registration," *IEEE Workshop on Mathematical Methods in Biomedical Image Analysis,* Kauai, Hawaii, Dec. 9–10, pp. 29–36, 2001.

[65] L. A. Shirman and C. H. Sequin, "Local surface interpolation with shape parameters between adjoining gregory patches," *Computer Aided Geometric Design*, **7**, pp. 375–388, 1990.

[66] J. Stam, "On subdivision schemes generalizing uniform B-spline surfaces of arbitrary degree," *Computer Aided Geometric Design*, **18**, pp. 383–396, 2001.

[67] D. Steiner and M. E. Kirby, "Geometric referencing of LANDSAT images by affine transformation and overlaying of map data," *Photogrammetria*, **33**, pp. 41–75, 1977.

[68] G. Stockman, S. Kopstein, and S. Benett, "Matching images to models for registration and object detection via clustering," *IEEE Trans. Pattern Analysis and Machine Intelligence,* **4**, 3, pp. 229–241, 1982.

[69] J-P Thirion and A. Gourdon, "Computing the differential characteristics of isointensity surfaces," *Computer Vision and Image Understanding,* **61**, 2, pp. 190–202, 1995.

[70] M. Trajković and M. Hedley, "Fast corner detection," *Image and Vision Computing,* **16**, pp. 75–87, 1998.

[71] G. J. Vanderburg and A. Rosenfeld, "Two-stage template matching," *IEEE Trans. Computers,* **26**, 4, pp. 384–393, 1977.

[72] G. Wahba, *Spline Models for Observation Data*, Philadelphia, PA, SIAM Press, 1990.

[73] H. Wendland, "Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree," *Advances in Computational Mathematics,* **4**, pp 389–396, 1995.

[74] S. Yam and L. S. Davis, "Image registration using generalized Hough transform," *Proc. IEEE Conf. Pattern Recognition and Image Processing,* pp. 526–533, 1981.

[75] C. T. Zhan, "An algorithm for noisy template matching," *IFIP Congress,* Stockholm, Aug. 5-10, pp. 698–701, 1974.