A Comparative Study of Transformation Functions for Nonrigid Image Registration

Lyubomir Zagorchev and Ardeshir Goshtasby

Dept. Computer Science & Engineering
Wright State University
Dayton, OH 45435

Abstract—Transformation functions play a major role in the registration of images with nonlinear geometric differences. In this paper, the characteristics of thin-plate spline (TPS), multiquadric (MQ), piecewise linear (PL), and weighted mean (WM) transformation functions are explored and their performances in the registration of images with nonlinear geometric differences are compared. TPS and MQ are most suitable when the set of control points is not large (up to a few hundred) and variation in spacing between the control points is small. When spacing between the control points varies greatly, PL produces a more accurate registration than TPS and MQ. When a very large number (in the order of thousands) of control points is available and the control points contain positional noise, WM is preferred over TPS, MQ, and PL because it uses an averaging process that smooths noise among the control points. Use of transformation functions in the detection of incorrect correspondences and in the determination of the nature of geometric difference between two images is also discussed.

Index Terms—Image registration, transformation function, thin-plate spline, multiquadric, radial basis function, piecewise linear, weighted-mean

1 Introduction

Image registration is a computational method for determining the point-by-point correspondence between two images of a scene. It can be used to fuse complementary information in images or estimate the geometric and/or intensity difference between two images. A typical registration method involves 1) determining a number of corresponding control points in the images, 2) determining a transformation function from the correspondences, and 3) determining the correspondence between all points in the images from the obtained transformation.

This paper discusses the second step in image registration, that is, the determination of a transformation function from a given set of corresponding control points in the images. The first step in image registration is perhaps the most difficult one, but it has been the subject of numerous studies. Typically, corner points [1, 12, 14, 28, 32, 42, 51, 52], region centroids [25], and line intersections [50] are used as control points and correspondence between control points in the images is determined by chamfer matching [4, 5, 8, 33], graph matching [11], random sample consensus [13], probabilistic relaxation labeling [23, 41], matching of minimum-spanning-tree edges [56], matching of convex-hull edges [24], Hausdorff distance [31, 44], hierarchical attribute matching [48], clustering [50], template matching [22, 16, 53], and Hough transform [55]. The third step in image registration is a resampling process and once a transformation function is determined, by scanning one of the images, corresponding pixels in the second image can be determined, establishing correspondence between the pixels in the images.

It is assumed that the control points given in the images correctly correspond to each other. The task is to determine a transformation function from the correspondences that maps all points in the images to each other. Cases where some of the correspondences are incorrect are also considered and a method for detecting and eliminating the incorrect correspondences is described.

The problem to be solved is as follows: Given the coordinates of \( N \) corresponding control points in two images,

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

it is required to determine a transformation function \( f(x, y) \) with components \( f_x(x, y) \) and \( f_y(x, y) \) that satisfy

\[
X_i = f_x(x_i, y_i), \quad i = 1, \ldots, N, \quad (2)
\]

or

\[
X_i \approx f_x(x_i, y_i), \quad i = 1, \ldots, N, \quad (4)
\]

Once \( f(x, y) \) is determined, given the coordinates of a pixel \((x, y)\) in one image, the coordinates of the corresponding pixel in the other image can be determined. We will refer to the image with coordinates \((x, y)\) as the reference image and the image with coordinates \((X, Y)\) as the sensed image. The reference image is kept unchanged and the sensed image, which is a newly scanned or sensed image, is resampled to spatially align with the reference image.

If the coordinates of corresponding control points in the images are arranged into two sets of 3-D points:

\[
\{(x_i, y_i, X_i) : i = 1, \ldots, N\},
\]

\[
\{(x_i, y_i, Y_i) : i = 1, \ldots, N\},
\]

then the components of the transformation as defined by equations (2) and (3) can be considered two explicit surfaces that
interpolate the two sets of 3-D points, and the components of the transformation defined by (4) and (5) can be considered explicit surfaces approximating the two sets of 3-D points. 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, the problem of finding a single-valued surface \( f(x,y) \) that interpolates or approximates

\[
\{(x_i, y_i, f_i) : i = 1, \ldots, N\}
\]

is considered. That is, the parameters of a transformation are determined to satisfy

\[
f_i = f(x_i, y_i), \quad i = 1, \ldots, N,
\]

or

\[
f_i \approx f(x_i, y_i), \quad i = 1, \ldots, N.
\]

Given a set of corresponding control points in two images, many transformation functions can be found to accurately map the control points in the sensed image to the corresponding control points in the reference image. A proper transformation function will accurately map the remaining points in the sensed image to the reference image also. Some transformation functions, although mapping corresponding control points to each other accurately, warp the sensed image too much, causing large registration errors in areas 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 in the correspondences are present, approximation methods are preferred over interpolation methods.

Various transformation functions have been used in image registration. In this paper, the properties of four popular transformation functions are examined and a guide to their selection is provided. To determine the behaviors of the transformation functions and to find their accuracies in image registration, a number of test images as shown in Fig. 1 are prepared. Fig. 1a is an image of size 512 \( \times \) 512 pixels containing a 64 \( \times \) 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

\[
X = 0.7x - y + 3, \\
Y = 0.9x + 0.8y + 5,
\]

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

\[
X = x - 8 \sin(x/16), \\
Y = y + 4 \cos(y/32).
\]

Translation and rotation represent rigid transformation and scaling represents a similarity transformation. The rigid, similarity, and linear transformations are used to determine the performances of the methods when (segments of) the images do not have nonlinear geometric differences. An idealistic transformation will not nonlinearly deform (a segment of) an image if (corresponding segments in) the images do not have nonlinear geometric differences. The sinusoidal transformation is selected to create images with local geometric differences and see how well various transformations can compensate for such geometric image differences.

Different densities 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 its density. Although uniformly spaced control points are rarely obtained in images, they are used here to determine the influence of the spacing and organization of the control points on the registration accuracy. Five geometric transformations are applied to the reference image to obtain the sensed images, and four sets of the control points are used to estimate the transformation in each case. Therefore, overall, there are twenty cases to be tested.

The control points in the reference image are the grid points shown in Figs. 2a–d. The control points in the sensed image are the corresponding grid points obtained by transforming them with the above transformations and rounding the obtained coordinates. From the coordinates of corresponding control points, the transformations are estimated and compared with the true transformations. In these experiments, although incorrect correspondences do not exist, corresponding control points contain rounding errors (digital noise).

**2 Thin-Plate Spline Interpolation**

Thin-plate spline (TPS) or surface spline [27, 39] is perhaps the most widely used transformation function in the regis-
The equations are obtained from the following constraints:

\[
\begin{align*}
\sum_{i=1}^{N} F_i &= 0, \quad \text{(16)} \\
\sum_{i=1}^{N} x_i F_i &= 0, \quad \text{(17)} \\
\sum_{i=1}^{N} y_i F_i &= 0. \quad \text{(18)}
\end{align*}
\]

Constraint (16) ensures that the sum of the loads applied to the plate is 0 to make the plate remain stationary. Constraints (17) and (18) ensure that moments with respect to \(x\) and \(y\) axes are zero, so the surface does not rotate under the imposition of the loads. TPS has a linear term and a sum of radially symmetric functions. TPS belongs to a class of radial basis functions widely used in the interpolation of scattered data.

Using TPS to map Figs. 1b–f to Fig. 1a with the control points shown in Fig. 2, the results reported in Table 1 are obtained. 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 so that it spatially aligns with the reference image. This transformation maps the control points in the sensed image to corresponding control points in the reference image exactly. 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 of Table 1 are zero because TPS is an interpolating function and it maps corresponding control points to each other exactly. 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 control points. The control points shown in Fig. 2a are overlaid with the resampled image. Although corresponding control points map to each other exactly, errors are visible away from the control points, especially around the image borders.

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

Figure 3: The effect of density and organization of control points in the registration accuracy of TPS. (a)–(d) The resampling of Fig. 1f to align with Fig. 1a using all control points. The control points shown in Figs. 2a–d, respectively, to align with Fig. 1a.
between the control points varies, some of the coefficients of the radial basis functions become nonzero. When images have nonlinear geometric differences, errors become quite large as shown in the rightmost column in Table 1. Although corresponding control points are mapped to each other quite well, at non-control points, errors can be quite large. Figs. 3a–d show the 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, it can be seen that TPS is not suitable for the registration of images with local geometric differences. In Figs. 3c and 3d it can be seen that although reasonably good accuracy can be achieved at and near the control points, errors are quite large away from the control points. This can be attributed to the fact that logarithmic basis functions are radially symmetric and when the arrangement of the control points is nonuniform, large errors are obtained in areas where large gaps exist. One should also note that as the number of control points is increased, since the matrix of coefficients becomes larger, the numerical stability of the method decreases.

The stiffness parameter \(d^2\) controls the shape of the interpolating surface and changes the registration accuracy. With the control point arrangements given in Fig. 2, best accuracy was obtained when the stiffness parameter was zero. Larger stiffness parameters increased registration errors in these test 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. [43, 54] added a smoothing term to the interpolating spline while setting \(d^2 = 0\) to obtain a surface that would contain smaller fluctuations but approximate rather than interpolate the points. As the smoothness term is increased, the obtained surface becomes smoother and fluctuations become smaller, but the surface gets 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 stop the process when the desired surface is obtained. 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.

### 3 Multiquadric Interpolation

Multiquadric (MQ) interpolation [29, 30] is another method based on radial basis functions and is defined by

\[
f(x, y) = F_1 + F_2 x + F_3 y + \sum_{i=4}^{N} F_i R_i(x, y),
\]

where

\[
R_i(x, y) = \sqrt{(x - x_i)^2 + (y - y_i)^2 + d^2}.
\]

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. As \(d^2\) is increased, a smoother surface is obtained. In a comparative study carried out by Franke [15], MQ was found to have an accuracy slightly better than that of TPS in the interpolation of randomly spaced data.

When registering Figs. 1b–f and Fig. 1a using MQ with the control points shown in Fig. 2, the results in Fig. 4 are obtained. The errors are summarized in Table 2. Comparing these results with those in Table 1, it can be seen that MQ and TPS produce virtually the same result when the images do not have nonlinear geometric differences, but when the images have nonlinear geometric differences, MQ produces slightly better results than TPS. One point to note is that although TPS produces the best accuracy for the images tested in this paper when parameter \(d^2\) is 0, when MQ is used, \(d^2 = 0\) rarely produces the best accuracy, and determination of the optimal \(d^2\) is not trivial. It requires a steepest descent algorithm to estimate it, and that would make MQ several times slower than TPS.

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

### 4 Weighted Mean Method

TPS and MQ are interpolating methods. They map corresponding control points in the images to each other exactly. Methods that map corresponding control points to each other approximately are defined by a weighted sum of the control points, with the sum of the weights equal to 1 everywhere in the approximation domain. A weighted mean (WM) method is formulated as

\[
f(x, y) = \sum_{i=1}^{N} f_i b_i(x, y)
\]

where

\[
b_i(x, y) = \frac{R_i(x, y)}{\sum_{i=1}^{N} R_i(x, y)}
\]

is the \(i\) weight function and \(R_i(x, y)\) is a monotonically decreasing radial basis function centered at \((x_i, y_i)\).
When $R_i(x, y)$ is a Gaussian, the weights are called rational Gaussians [21]. 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. This property makes the weight functions stretch toward the gaps. As the widths of the weight functions are reduced, the obtained surface gets closer to the points and at the limit when the widths of the weight functions are all 0, the surface will interpolate the points. As the widths of the weight functions decrease, flat spots start to appear in function $f$ at the data points. This is demonstrated in a simple example in Fig. 5. Consider the following five data points: $(0, 0, 0), (1, 0, 0), (1, 1, 0), (0, 1, 0)$, and $(0.5, 0.5, 0.5)$. The single-valued rational Gaussian surface approximating the points with increasing $\sigma_i$'s are shown in Figs. 5a–d. For small $\sigma_i$'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 map densely to points near the control points in the sensed image even when the reference and sensed images do not have any geometric differences.

Consider the following 5 correspondences:

\[
[(0, 0)(0, 0)]; [(1, 0)(1, 0)]; [(1, 1)(1, 1)]; [(0, 1)(0, 1)]; [(0.5, 0.5)(0.5, 0.5)]
\]  

(23)

The transformation that maps uniformly spaced points in the reference image to points in the sensed image for increasing $\sigma_i$'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 $\sigma_i$'s are reduced, the density of points increase near the control points and the function more closely approximates the control points. As $\sigma_i$'s are increased, spacing between the points becomes more uniform, but the function moves away from the control points, increasing the approximation error. Ideally, we want a transformation function that will map corresponding control points more closely when $\sigma_i$'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.

When the images do not have any geometric differences, in order to obtain rather uniformly spaced points in the sensed image when using uniformly spaced points in the reference image, each component of the transformation should be a parametric surface. 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

\[
\{(x_i, y_i, f_i) : i = 1, \ldots, N\}
\]  

(24)

with parameter coordinates (nodes) $(u_i = x_i, v_i = y_i) : i = 1, \ldots, N$. Given pixel coordinates $(x, y)$ in the reference image, first, corresponding parameter coordinates $(u, v)$ are determined from $x = x(u, v)$ and $y = y(u, v)$. Then, knowing $(u, v)$, $f = f(u, v)$ is evaluated. 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. The nonlinear equations can be solved by initially letting $u = x$ and $v = y$ and iteratively revising $u$ and $v$ by Newton's method [35].

To determine the accuracy of approximation methods in image registration, the control points shown in Fig. 2 and the geometric differences 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 becomes more detailed, mapping corresponding 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 when the geometric difference between images varies smoothly.

The standard deviations of Gaussians are set inversely proportional to the density of control points. This will produce narrow Gaussians where the density of points is high and wide Gaussians where the density of points is low. The process can be made totally automatic by setting $\sigma_i$ equal to the radius of a circular region surrounding the $i$th control point.
point in the reference image containing \(n - 1\) 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, producing a smoother surface, and as \(n\) is decreased the widths of Gaussians decrease, producing a more detailed surface.

The results shown in Table 3 in parentheses are obtained with \(n = 5\), and results without parentheses are obtained when \(n = 9\). 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 map to 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 particularly when the standard deviation of the Gaussians are very small, the surface quickly vanishes outside the convex hull of the control points. This is in contrast to monotonically increasing radial basis functions such as TPS and MQ where resampling is possible outside the convex hull of the control points. The approximation method with rational Gaussian weights, however, has produced more accurate results than TPS and MQ in areas where resampling was possible. In all experiments reported in this paper, only control points existing within both images were used in the calculations. Control points falling outside the sensed image domain due to sinusoidal deformation of the reference image were not used in the calculations.

Experiments show that when \(n = 2\) or \(\sigma_i\) is equal to the distance of the \(i\)th 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 is reduced, producing a smoother resampling. Increasing \(n\) too much, however, will make it impossible to reproduce sharp changes in geometric differences between images. \(n\) should, therefore, be selected using information about the geometric differences between the images as well as the density of the points.

Formula (21) 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 and MQ are used, the condition number of the matrix of coefficients increases with the size of the matrix of coefficients [6]. For certain arrangements of the control points, even very small systems of equations may produce very unstable equations involving large errors in registration.

Methods to make the weighted mean method local have been proposed by Mauve [30] and McLain [38] using rational weights of the form given in (22) but with

\[
R_i(x, y) = \begin{cases} 
1 - 3x_i^2 + 2r_i^3, & 0 \leq r_i \leq 1, \\
0, & r_i > 1,
\end{cases}
\]  

(25)

where

\[
r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}/r_n
\]  

(26)

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

\[
\frac{dR_i(x, y)}{dr_i} \bigg|_{r_i=1} = 0,
\]  

(27)

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 polynomials is then used to represent a component of the transformation. Therefore, the formula for a local weighted mean is

\[
f(x, y) = \frac{\sum_{i=1}^{N} R_i(x, y) p_i(x, y)}{\sum_{i=1}^{N} R_i(x, y)}
\]  

(28)

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, 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 [20].

5 Piecewise Linear Method

Piecewise linear (PL) transformation linearly maps corresponding regions in the images to each other. Although so far only triangular regions have been used, regions of any shape and size can be used. PL mapping will be continuous, but it will not be smooth. When the regions are small or local geometric differences between the images are small, PL may be sufficient. However, if local geometric differences between the images are large, tangents at the two sides of a region boundary may become very different, resulting in registration inaccuracies.

If control points in the reference image are triangulated, by knowing the correspondence between the control points in the sensed and reference images, corresponding triangles in the sensed image will be known. The choice of triangulation will affect the registration accuracy. 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 are known as Delaunay triangulation [26, 34]. Better approximation accuracy will be achieved if \( f \) is used in addition to \((x_i, y_i)\) to triangulate the data points [2, 47].

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 [7, 9, 10, 45].

Using the coordinates of control points in the reference image and the \( X \) or the \( Y \) coordinate of the corresponding control points 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 [37, 40, 49]. Subdivision surfaces contain B-spline, piecewise Bézier, and non-uniform B-spline (NURBS) as special cases [46]. 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.

PL transformation is efficient and often sufficient for nonrigid image registration [18]. An extension of piecewise linear registration to piecewise cubic registration is made in [19]. Piecewise methods can register image regions within the convex hulls of the control points. Although extrapolation is possible outside the convex hulls, it could lead to large registration errors.

Table 4 shows the results of mapping Figs. 1b–f to Fig. 1a with control points shown in Fig. 2 using PL transformation. Errors are slightly worse than those obtained by WM with rational Gaussian weights, but they are much better than those of TPS and MQ. 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 function was used 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.

6 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 function and the time to resample the sensed image to the coordinate system of the reference image. In the following, the computational complexity of each transformation discussed above is determined. An addition, a subtraction, a multiplication, a division, a comparison, or a small combination of them is considered an operation in the following discussions.
To determine a component of the TPS transformation, a system of $N$ linear equations has to be solved. This requires on 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 on the order of $n^2N$ operations. Overall, the computational complexity of TPS is $O(N^2) + O(n^2N)$. The computational complexity of MQ is also $O(N^2) + O(n^2N)$. MQ is, however, several times slower than TPS as determination of the optimal parameter $d^2$ requires an iterative steepest descent algorithm.

WM transformation uses rational weights with coefficients that are the coordinates of the control points. Therefore, a transformation is immediately obtained from the coordinates of corresponding control points without solving a system of equations. Mapping of each point in the sensed image to the space of the reference image takes on the order of $N$ operations because coordinates of all correspondences are 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 Gausians, 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.

The computational complexity of the PL transformation involves the time to triangulate the control points. This is on the order of $N \log N$ operations. Once the triangles are obtained, there is a need to find a transformation for each corresponding triangle. 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 PL transformation is $O(N \log N) + O(n^2N)$.

Table 5 summarizes the computational complexities of the transformation functions discussed in this paper.

**7 Detecting the Wrong Correspondences**

Transformation functions contain information about the 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 scene and may be significant in interpretation of the scene. Figs. 9a and 9b show the $X$ and $Y$ components of the ideal transformation obtained for registering images in Figs. 1f and 1a. The sinusoidal geometric difference between the images is 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. 9a and 9b. This is observed when all the correspondences are accurate.

In an experiment, two of the control points in Fig. 2a were displaced horizontally, two were displaced vertically, and one was displaced both horizontally and vertically, each by 15 pixels. The displacements are reflected in the obtained transformation as depicted in Figs. 9c and 9d. 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.

Figure 9: (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.

Note that horizontal displacements appear only in the $X$-component of the transformation and vertical displacements appear only in the $Y$-component of the transformation. Displacement in other directions will appear in both the $X$-component and the $Y$-component of the transformation. An inaccurate correspondence, therefore, could appear in one or both components of a transformation. A transformation not only provides information about the geometric difference between two images, it contains valuable information about the
accuracy 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 will point to the inaccurate correspondences.

8 Concluding Remarks

Images representing different views of a 3-D scene contain nonlinear geometric differences. To register such images, first a number of corresponding control points in the images must be determined and then a transformation function that can accurately represent the geometric difference between the images must be found. Image registration accuracy is, therefore, controlled by the accuracy of the control-point correspondences and the accuracy of the transformation function that represents the geometric difference between the images. This paper compared the performances of four popular transformation functions used in nonrigid image registration.

If information about the geometric difference between the images is available, that information should be used to select the transformation. If no such information is available, a transformation function that can adapt to the local geometric difference between the images must be chosen.

Use of TPS, MQ, WM, and PL transformations in the registration of images with nonlinear geometric differences was discussed. Experiments show that among the four transformations, TPS and MQ are least suitable for registration of images with local geometric differences. This is because of three main reasons. First, the basis functions from which a component of a transformation is determined are radially symmetric. When spacing between the control points is nonsymmetric, large errors will be obtained in areas where the arrangement of control points is nonuniform. Second, because the radial basis functions used in TPS and MQ are monotonically increasing, the process is global and it cannot adapt well to the local geometric difference between the images. Third, a system of equations has to be solved to find each component of a transformation, and when spacing between the control points varies greatly the system of equations to be solved becomes ill-conditioned. In registration cases where a small number of widely spread control points are provided and the local geometric difference between the images is not large, TPS and MQ are actually very effective in registering the images and are preferred over WM and PL. This is because they are interpolating functions and map corresponding control points to each other exactly and also because they extend beyond the convex hull of the control points.

PL interpolation is most suitable for the registration of images with local geometric differences because a control point affects a small neighborhood surrounding it. This property not only makes the computations very efficient, it keeps inaccuracies in the correspondences local without spreading them over the entire image domain. Recent subdivision schemes [37, 49] provide efficient means for fitting piecewise smooth patches over triangular regions in such a way that adjacent patches patches join smoothly, creating continuous and smooth transformation functions.

WM approximation is preferred over TPS and MQ in nonrigid image registration for four main reasons. First, it does not require the solution of a system of equations. A transformation is immediately obtained from the coordinates of corresponding control points in the images. Second, a transformation is obtained from a weighted mean of the control points and the averaging process that is involved smoothes noise in the correspondences. Third, the rational weights adapt to the density and organization of the control points by automatically stretching toward large gaps among the control points and their widths increase or decrease with the density of the control points. Fourth, the width of all weight functions can be controlled globally by changing the neighborhood size n of the transformation.

References


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

<table>
<thead>
<tr>
<th></th>
<th>Translation</th>
<th>Rotation</th>
<th>Scaling</th>
<th>Linear</th>
<th>Nonlinear</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform, Dense</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Uniform, Sparse</td>
<td>0.0</td>
<td>1.4</td>
<td>1.4</td>
<td>1.4</td>
<td>27.0</td>
</tr>
<tr>
<td>Random</td>
<td>0.0</td>
<td>1.4</td>
<td>1.4</td>
<td>1.4</td>
<td>14.1</td>
</tr>
<tr>
<td>Varying Density</td>
<td>0.0</td>
<td>2.2</td>
<td>2.0</td>
<td>1.4</td>
<td>20.0</td>
</tr>
</tbody>
</table>

Table 2: Registration accuracy of MQ using the control points shown in Fig. 2 and geometric differences between images in Figs. 1b–f versus the image in Fig. 1a. Numbers inside parentheses show errors when optimal $d_2$ was used, while numbers not in parentheses are errors obtained with $d_2 = 0$.

<table>
<thead>
<tr>
<th></th>
<th>Translation</th>
<th>Rotation</th>
<th>Scaling</th>
<th>Linear</th>
<th>Nonlinear</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform (Dense)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Uniform (Sparse)</td>
<td>3.4 (0.0)</td>
<td>6.2 (1.4)</td>
<td>6.4 (1.4)</td>
<td>2.4 (1.4)</td>
<td>30.4 (21.2)</td>
</tr>
<tr>
<td>Random</td>
<td>3.5 (0.0)</td>
<td>5.4 (1.4)</td>
<td>5.4 (1.4)</td>
<td>5.3 (1.4)</td>
<td>16.1 (13.1)</td>
</tr>
<tr>
<td>Varying Density</td>
<td>4.2 (0.0)</td>
<td>6.8 (2.0)</td>
<td>6.5 (2.0)</td>
<td>20.9 (1.4)</td>
<td>33.2 (18.3)</td>
</tr>
</tbody>
</table>

12
Table 3: Registration accuracy of the rational Gaussian transformation when using the control points in Fig. 2 and the geometric differences shown in Fig. 1. The errors are for \( n = 9 \). When reducing \( n \) to 5, the errors shown inside the parentheses are obtained.

<table>
<thead>
<tr>
<th>Type of Transformation</th>
<th>Translation</th>
<th>Rotation</th>
<th>Scaling</th>
<th>Linear</th>
<th>Nonlinear</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform (Dense) MAX RMS</td>
<td>0.0 (0.0)</td>
<td>1.0 (0.0)</td>
<td>0.0 (0.0)</td>
<td>1.0 (1.0)</td>
<td>2.2 (1.4)</td>
</tr>
<tr>
<td>Uniform (Sparse) MAX RMS</td>
<td>0.0 (0.0)</td>
<td>1.4 (0.0)</td>
<td>1.0 (0.0)</td>
<td>1.4 (1.0)</td>
<td>9.0 (7.1)</td>
</tr>
<tr>
<td>Random MAX RMS</td>
<td>0.0 (0.0)</td>
<td>1.4 (1.4)</td>
<td>1.4 (1.4)</td>
<td>1.4 (1.4)</td>
<td>10.2 (9.0)</td>
</tr>
<tr>
<td>Varying Density MAX RMS</td>
<td>0.0 (0.0)</td>
<td>1.4 (1.4)</td>
<td>1.4 (1.4)</td>
<td>1.4 (1.4)</td>
<td>15.1 (15.0)</td>
</tr>
</tbody>
</table>

Table 4: Registration accuracy of PL using the control points shown in Fig. 2 and the geometric differences between Fig. 1b–f and Fig. 1a.

<table>
<thead>
<tr>
<th>Type of Transformation</th>
<th>Translation</th>
<th>Rotation</th>
<th>Scaling</th>
<th>Linear</th>
<th>Nonlinear</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform (Dense) MAX RMS</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Uniform (Sparse) MAX RMS</td>
<td>0.0</td>
<td>1.4</td>
<td>1.4</td>
<td>1.4</td>
<td>15.0</td>
</tr>
<tr>
<td>Random MAX RMS</td>
<td>0.0</td>
<td>1.4</td>
<td>1.4</td>
<td>1.4</td>
<td>16.1</td>
</tr>
<tr>
<td>Varying Density MAX RMS</td>
<td>0.0</td>
<td>1.4</td>
<td>1.4</td>
<td>1.4</td>
<td>16.0</td>
</tr>
</tbody>
</table>

Table 5: Computational complexities of various transformation functions when \( N \) corresponding control points are used and the images are of size \( n \times n \) pixels.

<table>
<thead>
<tr>
<th>Type of Transformation</th>
<th>Computational Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPS Interpolation</td>
<td>( O(N^2) + O(n^2N) )</td>
</tr>
<tr>
<td>MQ Interpolation</td>
<td>( O(N^2) + O(n^2N) )</td>
</tr>
<tr>
<td>WM Approximation</td>
<td>( O(n^2N) )</td>
</tr>
<tr>
<td>PL Interpolation</td>
<td>( O(N \log N) + O(n^2) )</td>
</tr>
</tbody>
</table>