Efficient subdivision-based image and volume warping

Gady Agam and Ravinder Singh
Department of Computer Science
Illinois Institute of Technology
Chicago, IL 60616
{agam,singrav}@iit.edu

Abstract

Warping is fundamental to multiple algorithms in computer vision and medical imaging such as image and volume registration. Warping is performed by determining a continuous deformation map and applying it to a given image or volume. In registration the deformation map is determined based on correspondence between two images. It is often the case that the deformation map can only be determined at discrete locations and so has to be interpolated. The discrete locations where the deformation map is determined form irregular sampling of the unknown continuous deformation map. Thin-plate splines are commonly used to perform the interpolation and provide an optimal solution in the sense of bending energy minimization. Assuming N samples of the deformation map and n^2 image pixels, thin plate splines require solving a $N \times N$ dense linear system with $O(N^3)$ complexity for determining spline coefficients and N computations per pixel with $O(Nn^2)$ complexity for determining interpolated values. When N and n are large as in the case of volumetric medical image analysis this cost becomes prohibitive. The approach proposed in this paper is based on subdivision surfaces and is capable of achieving similar quality results with $O(N \log N)$ complexity for coefficient determination and $O(n^2)$ complexity for computing interpolated values. Experimental results demonstrate two orders of magnitude performance improvement on actual clinical data.

1. Introduction

Image registration is of primary importance in computer vision and medical image analysis. In its essence, image registration [1, 2] is a computational method for determining the point-by-point correspondence between two images or volumes. In multiple image registration techniques the deformation map between images is obtained only at a discrete set of locations thus requiring interpolation to produce a continuous deformation field. The given values of the deformation map are normally irregular samples of the continuous deformation map.

Thin plate splines (TPS) are commonly used for interpolating deformation maps based on irregular samples [3]. TPS are based on radial basis functions and are optimal in the sense of minimizing the bending energy of the map. TPS is capable of modeling arbitrary non-rigid deformations. In its regularized form the TPS model includes the affine model as a special case. The computational cost of TPS becomes prohibitive when the number of Samples is large. Let N be the number of samples of a deformation map and n^2 be the number of pixel in an image. Using TPS requires the solution of a $N \times N$ dense system with $O(N^3)$ complexity for determining interpolation coefficients and evaluation of a radial basis function at N locations per pixel for interpolation with $O(Nn^2)$ complexity.

The problem of interpolating between a set of irregularly sampled points has been successfully addressed in geometric modeling through the use of subdivision surfaces. Subdivision surfaces start with an initial surface triangulation and iteratively refine it by splitting triangles to obtain a smooth surface. While intermediate surface subdivisions are only piecewise linear, the limit surface obtained in this way has C^2 continuity. The proposed approach is based on the modified butterfly subdivision [4, 5] which is changed to support the interpolation of deformation maps. In addition we describe an efficient scanline algorithm to determine the value of the deformation map at all image locations which exploits coherence and is extremely efficient. It is shown that the approach we propose is capable of achieving similar quality results with $O(N \log N)$ complexity for coefficient determination and $O(n^2)$ complexity for computing interpolated values. Experimental results demonstrate two orders of magnitude perform ace improvement on actual clinical data.

The following sections describe our approach in greater detail. Section 2 reviews related work. Section 3 describes a novel subdivision-based approach for interpolation

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

and warping. Section 4 presents quantitative experimental results comparing the proposed approach to known techniques. Section 5 concludes the paper.

2. Related Work

Thin plate splines are commonly used for representing flexible coordinate transformations due to the fact that they are parameter free, have a physical interpretation, and have a closed-form representation. Suppose z_i is the target function value at location (x_i, y_i) for $i = 1, \ldots, N$. Two such TPS models are used to describe a 2D image deformation, while three such models are used to describe a 3D volume deformation. While the remainder of this paper targets the 2D case it should be evident that extension to 3D is immediate. Let (u_i, v_i) be the deformation map sample at (x_i, y_i). We set z_i equal to u_i and v_i in turn to obtain one continuous transformation for each coordinate. The TPS interpolant $f(x, y)$ minimizes the bending energy [3]:

$$I_f = \int\int_{R^2} \left(\frac{\partial^2 f}{\partial x^2}\right)^2 + 2\left(\frac{\partial^2 f}{\partial x \partial y}\right)^2 + \left(\frac{\partial^2 f}{\partial y^2}\right)^2 dxdy \tag{1}$$

and has a solution of the form:

$$f(x, y) = a_1 + a_x x + a_y y + \sum_{i=1}^{N} w_i U(\| (x_i, y_i) - (x, y) \|) \tag{2}$$

where $U(r)$ is a radial basis function of the form of $U(r) = r^2 \log r^2$. The parameters of the TPS model w and a are the solution of the linear system:

$$\begin{bmatrix} K & P \\ P^T & 0 \end{bmatrix} \begin{bmatrix} w \\ a \end{bmatrix} = \begin{bmatrix} z \\ 0 \end{bmatrix} \tag{3}$$

where $K_{ij} = U(\| (x_i, y_i) - (x_j, y_j) \|)$, the i^{th} row of P is $(1, x_i, y_i)$, w and z are column vectors formed from w_i and z_i respectively, and a is a column vector with the elements a_1, a_x and a_y. To account for possible errors in the deformation map samples, regularization is used to trade off between exact interpolation. This results in minimizing the bending energy as follows:

$$H_f = \sum_{i=1}^{N} [z_i - f(x_i, y_i)]^2 + \lambda I_f \tag{4}$$

where λ is the regularization factor, controlling the amount of smoothing. The regularized TPS can be solved by placing K in (3) with $K + \lambda I$, where I is a $N \times N$ identity matrix. The computational complexity of TPS is given by $O(N^3) + O(Nn^2)$, where $O(N^3)$ is the complexity of solving a dense $N \times N$ linear system (3), and $O(Nn^2)$ is the complexity of computing the interpolated values at n^2 locations using (2). At each such location the radial basis function has to be computed for each of the N control points.

Hardy [6, 7] proposed a radial basis function known as Multiquadratic Interpolation (MQ) which is also obtained by solving a set of linear equations. When there are global geometric differences between the images, TPS performs better than MQ. When the images have local geometric differences, TPS and MQ performs similarly. The MQ method involves the determination of d^2 coefficients using a steepest descent algorithm, which makes MQ several times slower than TPS [8].

TPS and MQ are both interpolating techniques, mapping the control points exactly in both images. Several approximation methods have been proposed for faster computation of the transformation function. Donato and Belongie [9] proposed three approximation methods for TPS that address the computational problem through the use of a subset of corresponding control points. The first method is based on simple subsampling, which solves for TPS mapping between a randomly selected subset of the correspondence. The drawback of using this method was that some parts were not sampled at all and the mapping was poor in those areas. In the second method, an improved approximation was obtained by using a subset of the basis functions with all of the target values. The third method proposed an approach which is based on the Nystrom method [10]. The Nystrom Method provides means for approximating the eigenvectors used to compute the coefficients of TPS. Therefore, TPS coefficients were obtained by calculating matrix vector products in an appropriate order, instead of inverting a large matrix. The performance of methods 2 and 3 is similar.

The weighted Mean (WM) method [11] is another approximation method that map the corresponding control points to each other approximately by obtaining a weighted average of the control points, with the sum of the weights equal to 1 everywhere in the approximation domain. The 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 which has a $O(n^2 N)$ complexity. The WM method is preferred when having a large number of control points and the correspondences is noisy.

Piecewise Linear and Cubic transformations (PL) were proposed by Goshtasby [12, 13]. These methods have complexity of $O(N \log N) + O(n^2)$. PL transformations mainly involve the triangulation of the points in the reference image. Point correspondences in the source and target images are used to determine similar triangles. Triangular areas are then mapped using linear or cubic functions. As the transformation between different triangles are not identical, this approach results in discontinuities along triangle edges. Piecewise methods register image regions within the convex hull of the control points. PL mapping is continuous

but not smooth. When the regions are small or when local geometric differences between images are small, PL may be sufficient. When local geometric differences between images are large, tangents at the two sides of a boundary shared by triangles may become different thus resulting in large visible registration errors.

3. Proposed Approach

The proposed approach is an efficient alternative to TPS with $O(N \log N + n^2)$ complexity compared to $O(N^3 + Nn^2)$ complexity of TPS. The proposed approach is superior to the PL model in several aspects: the handling of extrapolation, the utilization of subdivision and local TPS to reduce mapping discontinuities, and the exploitation of coherence to speed up performance.

3.1. Overview

Given N samples of a continuous deformation mapping

$$\{(x_i, y_i), (dx_i, dy_i); i = 1, \ldots, N\} \tag{5}$$

we would like to determine an approximation of the continuous deformation function $f(x, y)$ with components $f_x(x, y)$ and $f_y(x, y)$ that satisfy:

$$dx_i = f_x(x_i, y_i) \quad ; \quad dy_i = f_y(x_i, y_i) \tag{6}$$

The coordinates of corresponding control points and deformation map samples are arranged as two sets of 3D points as follows:

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

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

Using this notation, the deformation maps $f_x(x_i, y_i)$ and $f_y(x_i, y_i)$ describe height surfaces and may be obtained using interpolation.

The proposed approach starts by triangulating the point set (7) (and similarly the point set (8)). This triangulation forms an initial approximation of the deformation map coordinates and is refined later through subdivision. It also supports the scanline algorithm described later which exploits coherence for extremely efficient determination of map values. Since the convex hull of these points may not cover the complete image, additional points are added at image corners and at regular intervals on the image boundaries. The z coordinate of the added points is extrapolated based on the values of a local neighborhood of the original points. The extrapolation process is discussed further in Section 3.2).

To improve the accuracy of the mapping interpolation, the initial triangulation is subdivided iteratively using the modified butterfly subdivision scheme. This is an interpolating scheme that guarantees that the interpolated surface will continue to pass through the initial control points. The modified butterfly subdivision interpolates all coordinate values in the same way. In the proposed approach the accuracy of the z coordinate is more important compared with the value of the other coordinates and so the modified butterfly subdivision scheme is changed to interpolate the z coordinate using a local TPS computation. This local TPS computation is performed only once before subdivision begins and has low computational cost. Experimental results show that the use of local TPS to interpolate the z coordinate in subdivision improves accuracy while not affecting performance in a meaningful way. The local TPS-based subdivision is discussed in greater details in Section 3.3.

Given the refined (subdivided) deformation map, it is necessary to produce specific values of the deformation map at all pixel locations within the image so that the deformation can be employed to warp the image. This part is normally very time consuming in other interpolation schemes. In the proposed approach, due to the triangulated structure of the deformation map, we describe a highly efficient scanline algorithm for producing the deformation map values at each location. The scanline algorithm exploits coherence within triangles to produce a simple incremental computation at each location. This process is described in Section 3.4.

3.2. Deformation Extrapolation

Given the point set (7) (and similarly the point set (8)), it is necessary to add points on the image boundaries so that the deformation map covers the complete image. Setting the (x, y) coordinates is simple and can be done in an arbitrary way (e.g. at regular intervals). The z coordinate of the added points needs to be computed to produce extrapolation of deformation map values. The extrapolation is done using points neighboring the convex hull of the given point set. Extrapolation is performed in a linear form by computing a plane equation using points in a local neighborhood near the convex hull. This extrapolation is attenuated based on the coherence of the points in the local neighborhood that produced the plane equation. It is also attenuated based on the distance of the extrapolation from the convex hull. To perform the extrapolation we assume that the point set was triangulated as described later.

Let $p_e = (x_e, y_e, dx_e)$ be an extrapolated point. The (x, y) coordinates are set and the z coordinate has to be computed. Let $p_h = (x_h, y_h, dx_h)$ be the closest point to p_e on the convex hull. The coordinates of p_e are all known. Let n_i be the vertex normal at p_i computed by averaging the normals of all the triangles that intersect at p_i. Let $n_h = (n_x, n_y, n_z)$ be the average neighborhood normal at p_h computed by averaging the normals of the first ring

neighbors of p_h (i.e. neighbors who share an edge with p_h). The point p_h on the extrapolation surface need to satisfy the plane equation and so: $(p_e - p_h) \cdot n_h = 0$. Thus we have:

$$dx_e = dx_h + \frac{(x_h - x_e)n_x + (y_h - y_e)n_y}{n_z} \quad (9)$$

The coordinate dx_e is attenuated by multiplying it by an exponential attenuation factor ξ depending on coherence of the local neighborhood and distance from the convex hull.

Let $r_e = p_e - p_h$ and $\hat{r}_e$ be a unit vector in the direction of r_e. A measure for the planar coherency of the local neighborhood is given by the cosine of the angle between n_h and r_e, that is: $|n_h \cdot \hat{r}_e|$. This coherence measure is zero in a coherent (i.e. planar) neighborhood and different from zero (but below one) in an incoherent neighborhood. A measure for the distance from the convex hull is given by $|r_e|$. Using these terms the attenuation factor ξ is given by:

$$\xi(n_h, r_e) = e^{-|n_h \cdot \hat{r}_e| \alpha |r_e|^2} \quad (10)$$

where $\alpha = 1/\bar{e}_h^2 \log(0.9)$ is a constant, and $\bar{e}_h$ is the average length of edges intersecting at p_h. The constant α is selected so that attenuation at the average edge distance is small (0.9). It should be noted that in linear mappings where the deformation map is a planar, the attenuation factor becomes one thus resulting if planar extrapolation.

The augmented point set containing the extrapolated points is triangulated to form the initial approximation of the deformation map. This approximation is later subdivided to increase the smoothness of the interpolation. The triangulation is performed in 2D by projecting all the points onto the (x, y) plane. In this work we use the Delaunay triangulation due to the fact that it tends to maximize the minimum angle in triangles and prevents slivers. The proposed approach does not depend on the triangulation scheme and may be employed using different triangulation schemes. We use an incremental insertion algorithm for Delaunay triangulation, which have an advantage of generalizing to arbitrary dimensionality. Incremental insertion algorithms operate by maintaining Delaunay triangulation, into which vertices are inserted one at a time. The algorithm used in our implementation is described by Bowyer/Watson [14, 15].

3.3. Local TPS-based Subdivision

The triangulated surface obtained in the previous steps forms an approximation of the desired continuous deformation map. To improve the accuracy of the mapping interpolation, the initial triangulated is subdivided iteratively using the modified butterfly subdivision scheme. The modified butterfly subdivision scheme is an interpolating scheme and so guarantees that the interpolated surface will continue to pass through the initial control points.

In the original butterfly scheme [4, 5], at each iteration, every triangle is subdivided into four triangles by connecting approximated midpoints along its edges. To guarantee smoothness, the approximated midpoints are computed using a weighted average in a local neighborhood. Assuming a regular mesh where the valence (affinity) of regular vertices is 6, the approximated midpoint is computed using a butterfly mask with weights of $W = [-1/16, 1/8, -1/16, 1/2, 1/2, -1/16, 1/8, -1/16]$. That is, given points p_i and p_j the approximated mid point between them p_{ij} is given by:

$$p_{ij} = \frac{1}{2}(p_i + p_j) + 2w(p_k + p_l) - w(p_m + p_n + p_o + p_p) \quad (11)$$

where $w = 1/16$, p_k and p_l are the remaining vertices of the triangles $(t_{ij}^{(1)}, t_{ij}^{(2)})$ that share the edge between p_i and p_j, and p_m, p_n, p_o, p_p are the remaining vertices of the triangles that are adjacent to $t_{ij}^{(1)}$ and $t_{ij}^{(2)}$. Note that the combination coefficients add to one thus resulting in an affine combination. As locality is used to determine new vertices, there is no need to solve a global system of equations. In mesh surfaces where not all the vertices are regular (i.e. having a valance of 6), there is a need to set the weights of the affine combination for defining new vertices differently. A modification of the basic butterfly subdivision scheme was developed by Zorin et al. [16] to include semiregular and irregular settings in which the vertices defining the edge have a valance other than 6. This is known as the modified butterfly subdivision.

Equation (11) can be applied for introducing new surface points by treating the x, y, z coordinates equally. Note, however, that the accuracy of the z coordinate is more important as it has direct influence over the value of the interpolated deformation map. Small inaccuracies in the x, y coordinates have little influence on accuracy. Due the importance of accuracy in the z coordinate we use local TPS interpolation to compute it whereas the x, y coordinates of added points are interpolated as above (Equation 11). To perform local TPS interpolation, TPS coefficients are computed for each triangle based on the triangle vertices and its first ring neighbors. The local TPS coefficients are computed and stored only once before subdivision begins and so do not affect much the overall computational cost. Given an edge $p_i p_j$ shared by the triangles $t_{ij}^{(1)}$ and $t_{ij}^{(2)}$, a new vertex is added along it using a two step process. First the x, y coordinates of the new vertex are computed using Equation (11) to yield (x_{ij}, y_{ij}). Then we set the z coordinate to the mean of the value interpolated based on the two neighboring triangles. Experimental results show that the use of local TPS to interpolate z coordinates improve accuracy while not affecting performance.

3.4. Scanline Processing

The subdivided mesh surface of the deformation map coordinates $f_x(x_i, y_i)$ (and similarly $f_y(x_i, y_i)$) forms an ap-

proximation of the continuous deformation map. To be able to use this deformation map to perform warping it is necessary to determine the value of the deformation mat at each image pixel. Given that the number of pixels in the image is large it is necessary to perform this step in an efficient manner. Using the triangulation of the surface, following algorithms for smooth polygon shading, the proposed approach employs an incremental scanline algorithm to exploit coherence within triangles which is highly efficient.

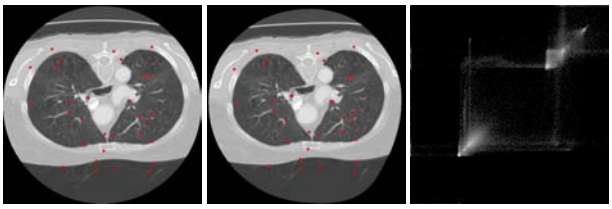
In the scanline algorithm each triangle is projected onto the x, y plane and then decomposed into a set of horizontal scanline intervals. Consider the i-th scanline of the k-th triangle. This scan line intersects exactly two triangle edges. The starting and ending points of this scanline are determined in integer coordinates by computing the intersection between the scanline and the respective triangle edge. The scanline equation is given by $y = y_i$ whereas the edge equation is given by $y = ax + b$. The intersection point between a triangle edge and the scanline is thus given by $y_i = ax_i + b$ from which we have $x_i = (y_i - b)/a$. The starting and ending points of a horizontal edge ($a = 0$) are known directly.

Given the computed starting and ending point of a scanline after rounding, we get a set of image pixels with integer coordinates belonging to the scanline. When moving from one pixel to the next on the same scanline the x coordinate of the pixel is simply incremented by one. The intersection of subsequent scanlines with the edge can be computed incrementally. Given an intersection point of (x_i, y_i) on the i-th scanline, the intersection point on the $i + 1$ scanline is given by: $(x_{i+1}, y_{i+1}) = (x_{i+1}, y_i + 1)$ where $x_{i+1} = (y_i + 1 - b)/a = x_i + 1/a$, thus resulting in a simple incremental computation.

4. Results and Discussion

The performance of the proposed approach is evaluated both qualitatively and quantitatively and compared to that of TPS. The evaluation uses 512 thoracic CT scans and is performed using 2D registration. Quantitative performance evaluation requires knowing the true deformation field. For this purpose we synthesize various deformation fields, both linear (e.g. rotation (T_r), translation (T_t), and scale (T_s)) and non-linear. The non-linear deformations are produced by generating random deformation vectors at discrete locations with random directions and lengths. These are then interpolated to produce a continuous deformation map. To avoid bias toward any of the evaluated methods, the interpolated deformation field is produced separately using both TPS (T_{tps}) and subdivision (T_{sub}).

The accuracy of the different approaches was evaluated by comparing the obtained results to expected (known) results. Accuracy was measured in both image and transformation space. In image space the error was measured by registering the images and measuring the average intensity



(a) (b) (c)

Figure 1. Sample test images. **(a)** Reference image. **(b)** Deformed target image. **(c)** Co-occurrence histogram between the target and reference images.

difference. Intensity errors are on a scale of $[0..255]$ where low intensity difference indicates high accuracy. In transformation space the error was measured as the average distance (in pixels) between the obtained and expected mapping of each pixel. Low pixel differences indicate high accuracy. In our evaluation intensity errors generally corresponded to transformation errors. Note that while the above error measures provide a quantitative measures they do not quantify the effects of discontinuities. The proposed subdivision approach is capable of reducing both intensity/transformation errors as well as discontinuities. The errors and time measurements in the evaluation where computed as the average of multiple, randomly generated, deformations.

Example of a typical thoracic CT section and its deformation used in our evaluation is shown in Figure 1. Figure 1-(a) shows the original section image, Figure 1-(b) shows the deformed section, and Figure 1-(c) shows the co-occurrence histogram of the two images. A diagonal co-occurrence histogram is an indication of successful registration.

Registration obtained using TPS and subdivision is presented in Figure 2. The first and second rows show respectively results obtained using TPS and subdivision with four iterations. The first column shows the registered images after warping, the second column shows the differences between the warped and reference image, and the third column shows the co-occurrence histogram between the reference and warped images. As can be observed TPS and subdivision produce similar results.

The effect of subdivision on the interpolated deformation map is shown in Figure 3. The first column displays a coordinate of the deformation map interpolated using TPS. The second, third, and fourth columns show the results after 0, 1, and 2 subdivision iterations respectively. The first row displays the x coordinate of the deformation map $f_x(x, y)$ whereas the second row shows the underlying triangulation. For clarity reasons, the values in this image were mapped to $[0..255]$. As can be observed, subdivision smooths the deformation map by removing the visible discontinuities between triangles.

Time performances comparison of TPS and subdivision

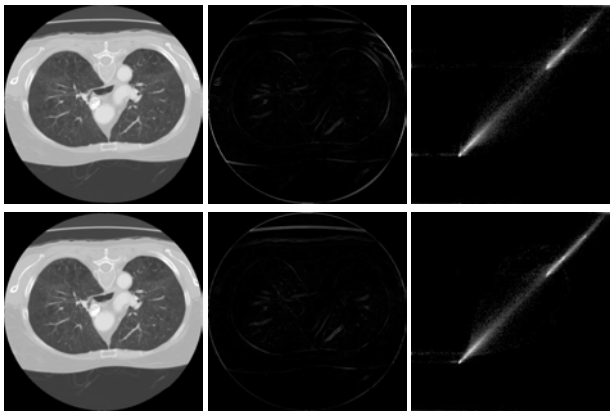


Figure 2. Comparison of TPS and subdivision warping.

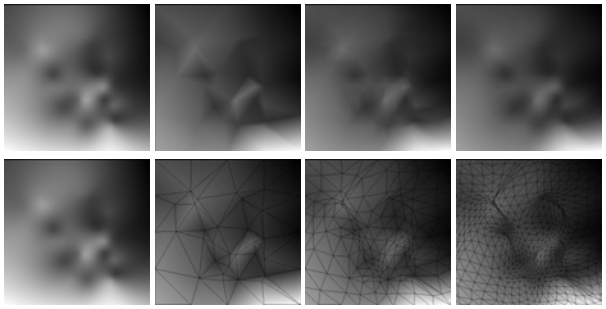


Figure 3. Effect of subdivision on interpolated displacement maps.

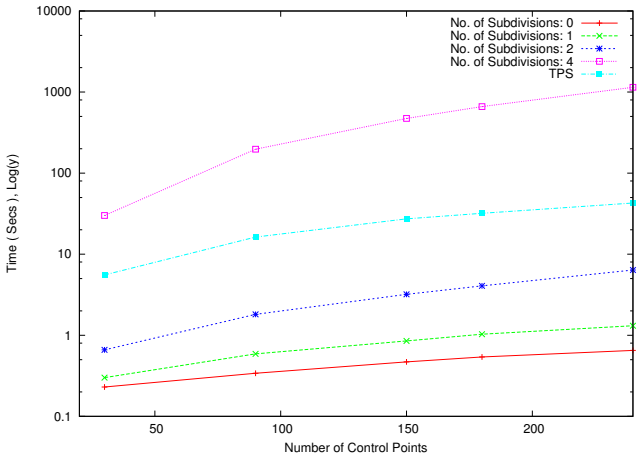


Figure 4. Time performance of TPS and subdivision as a function of the number of control points.

as a function of the number of control points is presented in Figure 4. As can be observed, subdivision is substantially faster than TPS. For i subdivision steps the number of triangles increases by a factor of 4^i and so eventually subdivision becomes less efficient. As shown later quantitatively, there is no much gain in subdivision beyond two subdivision steps. As expected, increasing the number of control points increases the time it takes to perform interpolation in both TPS and subdivision interpolation.

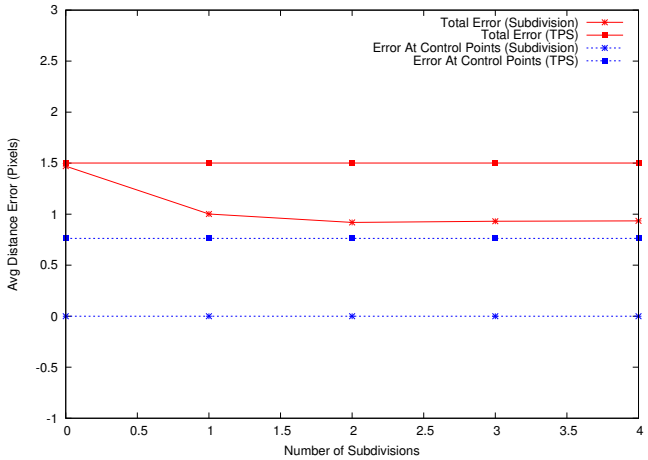


Figure 5. Comparison of subdivision and TPS for 50 control points.

Quantitative performance evaluation is presented in Figure 5 where transformation error in pixels is measured as a function of the number of subdivisions. As can be observed the error in subdivision is not improved much after one or two subdivision iterations. Note that the deformation used in generating this figure is T_{sub} and so subdivision produces smaller errors. When using T_{tps}, TPS obtains smaller errors with a comparable margin to the one in this Figure. Thus we conclude that the results obtained using TPS and subdivision are comparable.

Evaluation of interpolation error as a function of the range of control points in the image is shown in Figure 6. As expected, when the range of control points in images (512×512) increases the interpolation error decreases. This due to the fact that extrapolation is less accurate than interpolation. As can be observed, when the control points are sufficiently spread, a single subdivision step has the largest influence on accuracy.

5. Conclusion

We present a novel subdivision interpolation scheme that is suited for image and volume warping and registration. The advantage of the proposed approach is improved performance. Compared to TPS, the proposed approach produces similar results in terms of accuracy while reducing the processing time by up to two orders of magnitude. Using the proposed subdivision approach there is a tradeoff between accuracy and time performance. Experimental results show that after two subdivision iterations the computational cost increases with minor contributions to accuracy. While the proposed approach is described and evaluated in a 2D (image) context it can be easily extended to 3D (volume) warping.

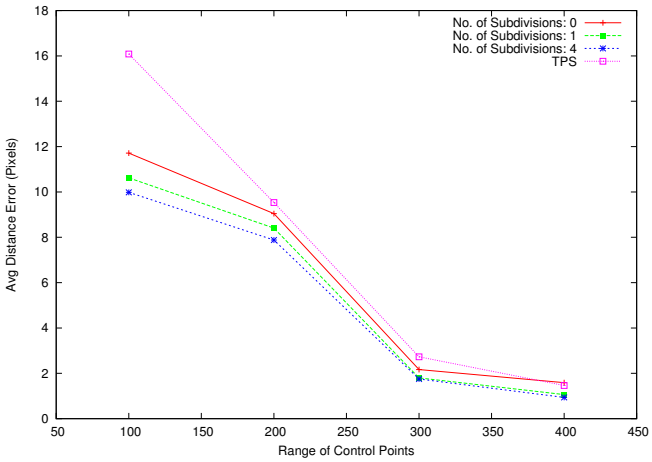


Figure 6. Evaluation of interpolation error as a function of the spread of control points in the image.

Acknowledgments

This work was supported in part by a grant from The Pritzker Institute of Biomedical Science and Engineering.

References

[1] B. Zitova and J. Flusser, "Image registration methods: A survey," *Image and Vision Computing*, vol. 21, pp. 977–1000, 2003.

[2] L. G. Brown, "A survey of image registration techniques," *ACM Comput. Surv.*, vol. 24, no. 4, pp. 325–376, 1992.

[3] F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Transaction On Pattern Analysis and Machine Intelligence*, vol. 2, no. 6, pp. 567–585, June 1989.

[4] N. Dyn, D. Levin, and J. A. Gregory, "A butterfly subdivision scheme for surface interpolation with tension control," *ACM Transactions on Graphics*, vol. 9, no. 2, pp. 160–169, 1990.

[5] N. Dyn, D. Levin, and C. A. Micchelli, "Using parameters to increase smoothness of curves and surfaces generated by subdivision," *Computer Aided Geometric Designs*, vol. 7, pp. 129–140, 1990.

[6] R. L. Hardy, "Multiquadric equations of topography and other irregular surfaces," *J. Geophys. Res*, vol. 76, no. 8, pp. 1905–1915, 1971.

[7] ——, "Theory and applications of the multiquadric-biharmonic method - 20 years of discovery - 1969 to 1988," *Computers Math. Appl.*, vol. 19, no. 8/9, pp. 163–208, 1990.

[8] L. Zagorchev and A. Gostasby, "A comparitive study of transformation functions for nonrigid image registration," *IEEE Transaction On Image Processing*, vol. 15, no. 3, pp. 529–538, March 2006.

[9] G. Donato and S. Belongie, "Approximate thin plate spline mappings," in *European Conference on Computer Vision (ECCV) (3)*, 2002, pp. 21–31.

[10] C. T. H. Baker, "The numerical treatment of integral equations," in *Oxford Clarendon Press*, 1977.

[11] A. Goshtasby, "Image registration by local approximation methods," *Image and Vision Computing*, vol. 6, no. 4, pp. 255–261, 1988.

[12] ——, "Piecewise linear mapping functions for image registration," *Pattern Recognition*, vol. 19, no. 6, pp. 459–466, 1986.

[13] ——, "Piecewise cubic mapping functions for image registration," *Pattern Recognition*, vol. 20, no. 5, pp. 525–533, 1987.

[14] A. Bowyer, "Computing dirichlet tessellations," *The Computer Journal*, vol. 2, no. 24, pp. 162–166, 1981.

[15] D. F. Watson, "Computing the n-dimensional delaunay tessellation with apllication to voronoi polytopes," *The Computer Journal*, vol. 2, no. 24, pp. 167–172, 1981.

[16] D. Zorin, P. Schroder, and W. Sweldens, "Interpolating subdivision for meshes with arbitrary topology," in *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, August 1996, pp. 189–192.