3-D Image Registration Using Fast Fourier Transform, with Potential Applications to Geoinformatics and Bioinformatics

Roberto Araiza
Matthew G. Averill
G. Randy Keller
Scott A. Starks
NASA Pan-American Center for
Earth and Environmental Studies (PACES)
University of Texas at El Paso
El Paso, Texas 79968, USA
{raraiza,sstarks}@utep.edu

Chandrajit Bajaj
Center for Computational Visualization
Computer Sciences Dept.
University of Texas at Austin
Austin, TX 78712, USA
bajaj@cs.utexas.edu

Abstract

FFT-based techniques are actively used to register 2-D images, i.e., to find the shift, rotation, and scaling necessary to align one image with the other. It is desirable to extend these techniques to the problem of registering 3-D images. Registration of 3-D images is an important problem in areas such as bioinformatics (e.g., in protein docking) and geoinformatics (e.g., in earth modeling).

Keywords: Image Registration, 3-D Images, Geoinformatics, Bioinformatics, Fast Fourier Transform

1 Image Registration: A Practical Problem

Formulation of the problem. In many areas of science and engineering, we have two images $I_1(\vec{x})$ and $I_2(\vec{x})$ (2-D or 3-D) which represent the same object but viewed from different angles and positions, in a possibly different scale. Since these images represent the same object, they can be obtained from each other by an appropriate shift $\vec{x} \to \vec{x} + \vec{a}$, rotation $\vec{x} \to R\vec{x}$, and scaling $\vec{x} \to \lambda \cdot \vec{x}$, i.e., $I_2(\vec{x}) \approx I_1(\lambda \cdot R\vec{x} + \vec{a})$.

In many practical situations, we do not know the relative orientation of the two images. In such situations, it is desirable to *register* these images, i.e., to find the rotation and the shift after which the images match as much as possible.

A similar problem occurs when we have the images of two different objects whose shapes should match. For example, we may have images of two bioactive molecules. We know that in vivo, these molecules interact because one of these molecules "docks" to the other one, i.e., gets into a position where their surfaces match. In such situations, it is also important to find orientation and shift corresponding to this match.

Comment. Sometimes, the images also differ in lighting conditions, as a result of which we may have $I_2(\vec{x}) \approx C \cdot I_1(\lambda \cdot R\vec{x} + \vec{a})$ for some unknown factor C.

General image registration problem vs. its particular cases. In the *general* problem, images may differ by shift, by orientation, and by scale. In such a general case, we need to find all three groups of parameters: shift $\vec{a}$, rotation R, and scaling λ.

In practice, we sometimes do not need to find all three of these parameters. For example, we sometimes know that the images are already similarly oriented and they are in the same scale, so all we need to do is find the shift $\vec{a}$ between the two images.

In other practical problems, we know that the images are scaled right, but we need to find the shift $\vec{a}$ and the rotation R. For example, when a molecules docks into another molecule, it can shift and rotate, but it cannot seriously shrink or expand. Thus, in the protein docking problem, we must find the shift $\vec{a}$ and the rotation R – but not the scaling.

In short, in general, all three groups of parameters are unknown: the shift $\vec{a}$, the rotation R, and the scaling λ. In practice, in addition to this complex general image registration problem, we sometimes face (somewhat) simpler particular cases of the general image registration problem, cases in which we only need to determine some of these parameters.

2 First Case: Registration of Similarly Oriented 2-D and 3-D Images

Formulation of the Problem. We have two images $I_1(\vec{x})$ and $I_2(\vec{x})$, we know that they are obtained from each other by an (unknown) shift $\vec{a}$, i.e., that $I_2(\vec{x}) \approx I_1(\vec{x}+\vec{a})$, and we want to find this shift.

How an image is represented. In the 2-D case, each image on a 2-D $n \times n$ grid consists of n^2 intensity values at different grid points. In the 3-D case, an image on a 3-D $n \times n \times n$ grid consists of n^3 intensity values. In general, an image is represented by n^d intensity values, where $d = 2$ or 3 is the dimension of the image.

Registration in time $O(n^d)$. In some real-life problems, we can register the two similarly oriented images in the smallest possible time $O(n^d)$. One example if when we are registering the two images which have clear *landmarks*: e.g., points where the intensity have the largest possible value, or saddle points which are useful in describing the structure of the image.

In such situations, to match the two images, it is sufficient to find, for each of the two images, the point $\vec{x}$ with the largest possible intensity value; this point can be found by searching over all n^d points $\vec{x}$ so it can be done in $O(n^d)$ steps. Once the corresponding landmark locations $\vec{d}_1$ and $\vec{d}_2$ are found, we can find the desired shift $\vec{a}$ as $\vec{a} = \vec{d}_2 - \vec{d}_1$.

For many important images, we cannot use this fast algorithm for image registration, because there are no easy-to-find landmark locations – namely, for every special point in the image, there are several similar points in the same image. For example, in multi-cellular biological images, whether we have a point on the border between several cells, or a special point of a special structure within a cell, there are many different points with the similar values of intensity at different cells. Similarly, in a satellite image, if the maximum of the image intensity corresponds to, e.g., road intersection or a river, we may have many road intersections and several rivers within the same image.

In some such situations, however, we can still use a fast ($O(n^d)$) algorithm for image registration. For example, in astronomical images, we often have an image of an object surrounded by empty space. In this case, we can use the moments to find out the shift between the images. It is easy to see that when we apply a shift to an image, i.e., replace the original image $I_1(\vec{x})$ with a shifted image $I_2(\vec{x}) = I_1(\vec{x}+\vec{a})$, then the 0-th order moment (overall intensity) $M_i = \int I_i(\vec{x})\,d\vec{x}$ does not change ($M_2 = M_1$), while the 1-st order moments $M_{ij} = \int I_i(\vec{x}) \cdot x_j \cdot d\vec{x}$ change to $M_{2i} = M_{1i} - a_i \cdot M_1$. Thus, we can find the desired shift $\vec{a}$ by comparing the corresponding moments: $a_i = (M_{1i} - M_{2i})/M_1$. An integral is, in effect, a sum over all n^d grid points, so computing each of the integrals M_1 and M_{ij} requires $O(n^d)$ steps. Thus, we still need computation time $O(n^d)$.

This method has a natural geometric meaning: the ratio M_{ji}/M_1 is the i-th coordinate of the image's center of gravity. Thus, in effect, this algorithm consists of the following three steps:

- first, we compute the center of gravity $d_{1i} = M_{1i}/M_1$ of the image $I_1(\vec{x})$;
- then, we compute the center of gravity $d_{2i} = M_{2i}/M_2$ of the image $I_2(\vec{x})$;
- finally, we find the desired shift as the difference between the centers of gravity: $\vec{a} = \vec{d}_1 - \vec{d}_2$.

Comment. This reformulation works also when the images are not only shifted, but also "re-scaled", i.e., $I_2(\vec{x}) \approx C \cdot I_1(\vec{x}+\vec{a})$ for some (unknown) constant $C > 0$, due, e.g., to different lighting conditions related to these two images.

Situations when $O(n^d)$ algorithms are not applicable. In many practical situations, none of the above $O(n^d)$ algorithms is applicable. For example, in satellite and multi-cellular images, not only we do not have landmark points, we also do not have empty space around an image. As a result, e.g., the shifted satellite image contains the values which are not present in the original image. When both images cover the same reasonably homogeneous area, then the intensity is of about the same value throughout both images, and hence, the center of gravity of each image is close to the geometric center of the image. Thus, if we apply the above center-of-gravity algorithm to detect the shift between the images, then, no matter what the actual shift is, the above algorithm will return (approximately) 0.

Straightforward approach: computational complexity. A natural idea is to look for a vector $\vec{a}$ for which the distance

$$\int (I_2(\vec{x}) - I_1(\vec{x}+\vec{a}))^2 \, d\vec{x} \tag{1}$$

attains the smallest possible value.

For this formula, a straightforward approach would mean that we compute the value of this scoring function for all possible shifts $\vec{a}$, and then find the shift for which this value is the smallest.

Each image on a 2-D $n \times n$ grid consists of n^2 intensity values at different grid points. Computing each integral requires time $O(n^2)$: for each pixel $\vec{x}$, we need one subtraction and one multiplication to compute the value $(I_2(\vec{x})-I_1(\vec{x}+\vec{a}))^2$, and then we must add all these values to get the integral.

According to the straightforward approach, we must compute the value of the L^2-norm for all possible shifts $\vec{a}$. For an $n\times n$ grid, it is reasonable to consider n^2 possible shifts. Computing each integral requires time $O(n^2)$, so overall, we need time $O(n^2) \cdot O(n^2) = O(n^4)$. For many real-life images, the size n is approximately 10^3, so $n^4 \approx 10^{12}$ computational steps require a large mount of time. It is therefore desirable to find faster algorithms for image registration.

It is known that many signal and image processing techniques can be made faster if we use Fast Fourier Transform (FFT); see, e.g., [12]. FFT of an image of size N requires $N \cdot \log(N)$ steps, so for an image of size $N = n^d$, we need $O(n^d \cdot \log(n))$ steps.

Using Fast Fourier Transform for shift detection. For a shifted image $I_2(\vec{x}) = I_1(\vec{x}+\vec{a})$, the Fourier transform has the form $F_2(\vec{\omega}) = \exp(\mathrm{i} \cdot \vec{\omega} \cdot \vec{a}) \cdot F_1(\vec{\omega})$. Thus, in the presence of measurement noise and inaccuracy, when we have $I_2(\vec{x}) \approx I_1(\vec{x}+\vec{a})$, we have $F_2(\vec{\omega}) \approx r(\omega) \cdot F_1(\vec{\omega})$, where $r(\omega) \approx \exp(\mathrm{i} \cdot \vec{\omega} \cdot \vec{a})$.

The complex number $\exp(\mathrm{i} \cdot \vec{\omega} \cdot \vec{a})$ has magnitude 1; so, it is reasonable, for every spatial frequency $\vec{\omega}$, to find the value $r(\vec{\omega})$ for which the approximation error $|F_2(\vec{\omega}) - r(\vec{\omega}) \cdot F_1(\vec{\omega})|^2$ is the smallest possible under the constraint that $|r(\vec{\omega})| = 1$.

By using the Lagrange multiplier method, we can show that the solution to this constrained optimization problem is given by

$$r(\vec{\omega}) = \frac{F_1(\vec{\omega}) \cdot F_2(\vec{\omega})^*}{|F_1(\vec{\omega}) \cdot F_2(\vec{\omega})|}. \tag{2}$$

In the ideal case, $r(\vec{\omega})$ is a sinusoidal wave, and its inverse Fourier transform is an impulse function $\delta(\vec{x}-\vec{a})$, i.e., a function which is only non-zero for $\vec{x} = \vec{a}$. It is therefore reasonable to find the shift $\vec{a}$ by applying the inverse FFT to the above function $r(\vec{\omega})$ and to find the shift as the point at which this inverse FFT attains the largest possible value.

We thus arrive at the following algorithm.

Resulting algorithm. We are given two images $I_1(\vec{x})$ and $I_2(\vec{x})$; we must find the shift $\vec{a}$ between these images. For that, we do the following:

- first, we apply FFT to both images $I_1(\vec{x})$ and $I_2(\vec{x})$, and compute the Fourier transforms $F_1(\vec{\omega})$ and $F_2(\vec{\omega})$;

- then, for each frequency $\vec{\omega}$, we compute the ratio (2);
- after that, we apply the inverse FFT to the resulting function $r(\vec{\omega})$ and obtain a new function $P(\vec{x})$.

Finally, we find the shift $\vec{a}$ as the value at which the function $P(\vec{x})$ attains its largest possible value.

Comment. This algorithm was first proposed in [26] (without the least squares justification); for 2-D images, it requires time $O(n^2 \cdot \log(n))$. This same algorithm can be used for 3-D images, it then requires time $O(n^3 \cdot \log(n))$.

Comment. The above method – based on traditional FFT techniques – is applicable when we have data on a rectangular grid. In many practical problems, e.g., in biomedical imaging, we often have areas with few grid points and areas with a large number of grid points. For such non-equispaced data, the traditional Fast Fourier Transform algorithm is not applicable.

What can we do in this situation? The idea behind Fourier transform is that instead of representing each image $I(\vec{x})$ by its values at different pixels, we use one of the possible bases – namely, the basis of sinusoid functions – and describe the image by the coefficients of this expansion:

$$I(\vec{x}) = \frac{1}{2\pi} \cdot \int F(\vec{\omega}) \cdot \exp(\mathrm{i} \cdot \vec{\omega} \cdot \vec{x}) \, d\vec{\omega}.$$

For non-equidistant data, we cannot use only this expansion to get an efficient algorithm. Thus, it is natural to use the expansion over other bases – e.g., polynomials – to help computations.

Several fast algorithms for nonequispaced Fourier transform have been proposed; see, e.g., [2, 6, 15, 16, 17, 34, 35, 43, 44, 47]. The main difference between the standard FFT and the nonequidistant FFT (NFFT) is that the standard FFT produces the *exact* Fourier transform, while the nonequidistant FFT only compute the values of the Fourier transform with a given accuracy.

Many of the NFFT algorithms use (local) polynomial approximations as an intermediate computational step; for example:

- in [44], an interval domain is divided into subintervals, and Chebyshev polynomials are used in each subinterval;
- in [2], Taylor polynomials are used instead of the Chebyshev ones;
- splines (a special class of locally polynomial functions) are used in [17, 34, 43].

In [4, 5, 9, 42], the spline-based methods from [17, 34, 43] and several other similar techniques have been successfully applied to the docking problem; the comparison seems to indicate that the spline-based methods are indeed the fastest.

3 Registration of General 2-D and 3-D Images

Formulation of the problem. We have two d-D images $I_1(\vec{x})$ and $I_2(\vec{x})$; we must find the shift $\vec{a}$ and the rotation R for which $I_2(\vec{x}) \approx I_1(R\vec{x} + \vec{a})$.

Registration in time $O(n^d)$. To find the shift between the two images, it is sufficient to find one landmark point. To find both shift and rotation, we can use two different landmarks:

- we can use the first landmark to find the shift between the two images, and then
- we can compare the location of the second landmark in both images, and thus find the corresponding rotation.

Finding the landmarks in the image requires time $O(n^d)$.

If there are no easily detectable landmarks, but the images are objects surrounded by an empty space, then we can use the moments method to determine both shift and rotation. The values of the 0-th and 1-st order moment M_i and M_{ij} could only determine the shifts. So, to find the rotation, we must also use the second order moments $M_{ijk} = \int x_j \cdot x_k \cdot I_i(\vec{x})\,d\vec{x}$.

To compare the second order moments (also called moments of inertia) of the two images, it is necessary to first shift both images to the same point of origin, e.g., to the center of gravity. As a result, we replace the original value M_{ijk} with the new value $M_{ijk} - d_j \cdot d_k \cdot M_i$, where $d_j = M_{ij}/M_i$ is the j-th coordinates of the center of gravity.

Second order moments of each image form a symmetric non-negative definite matrix. Thus, this matrix has orthogonal eigenvectors. When the image rotates, these eigenvectors rotate as well. So, we can find the rotation angle by comparing the orientations of the eigenvectors.

New algorithm for the general case: motivation and description. In general, the images may not have clear landmarks and they may not be surrounded by an empty space, so the above $O(n^d)$ methods may not be applicable. Let us show that in these case, we can use FFT-based techniques.

It is known that if the two images $I(\vec{x})$ and $I'(\vec{x})$ differ not only by shift but also by a rotation R and a scaling λ, the absolute values $M(\vec{\omega})$ and $M'(\vec{\omega})$ of their Fourier transforms $F(\vec{\omega})$ and $F'(\vec{\omega})$ differ from each other only by the corresponding rotation and scaling.

While the original images were not surrounded by an empty space, in many practical situations, the corresponding magnitudes of Fourier transforms are actually rapidly decreasing as the spatial frequency ω increases; see, e.g., [39, 40].

Comment. For smooth images, the magnitudes of their Fourier transforms do decrease with frequency. However, for non-smooth images, e.g., for a point source $I(\vec{x}) = \delta(\vec{x} - \vec{a})$ located at a point $\vec{a}$, its Fourier transform is equal to $\exp(-\mathrm{i} \cdot \vec{\omega} \cdot \vec{a})$. So, the magnitude of this Fourier transform is equal to 1 for all the frequencies and thus, does not decrease with frequency. Thus, if both given images contain a bright spot on a reasonably uniform background, then we cannot use this moments technique to find the rotation that registers these images. However, the case of bright spots (landmarks) is exactly the case when we can use landmark algorithms.

In situations when the magnitude of the Fourier transform decreases with frequency, we can use the moments method to find the appropriate rotation. Namely, based on the function $M_1(\vec{\omega})$, we compute the second moments $M_{1ij} \stackrel{\text{def}}{=} \int M_1(\vec{\omega}) \cdot \omega_i \cdot \omega_j\,d\vec{\omega}$; similarly, based on the function $M_2(\omega)$, we compute the second moments

$$M_{2ij} = \int M_2(\vec{\omega}) \cdot \omega_i \cdot \omega_j\,d\vec{\omega}.$$

For each of these matrices, we compute the eigenvectors corresponding to the largest eigenvalue; by comparing the orientations of these eigenvectors, we can then find the desired rotation R in time $O(n^d)$.

Comment. Images are usually given on a 2-D or 3-D grid, in the shape of a box. When we rotate the images an transform them back into a box shape, the corner information disappears. The only values which are preserved after all rotations are the values inside a circle (sphere) subscribed into the box. Since the corner values cannot match anyway, it is reasonable, before computing the moments, to only consider values $M(\vec{\omega})$ inside this circle (sphere).

Once we know the rotation R, we can align the images – e.g., by rotating the Fourier transform $F_1(\vec{\omega})$ of one of the image $I_1(\vec{x})$ to $F_1(R\vec{\omega})$. Now, the images $I_1(R\vec{x})$ and $I_2(\vec{x})$ differ only by shift, so we can use the above $O(n^d \cdot \log(n))$ method to find this shift.

Our preliminary results show that the resulting method indeed works well for registering 3-D images.

New algorithm: computational complexity. Overall, in the d-D case, we thus need $O(n^d \cdot \log(n))$ time to compute the original Fourier transforms, $O(n^d)$ time to find the rotation, and then $O(n^d \cdot \log(n))$ time to find the shift – to the total of

$$O(n^d \cdot \log(n)) + O(n^d) + O(n^d \cdot \log(n)) =$$
$$O(n^d \cdot \log(n))$$

computational steps.

Acknowledgements

This work was supported in part by NASA under cooperative agreement NCC5-209, NSF grants EAR-0225670, ACI-022037, EIA-0325550, NIH grants 3T34GM008048-20S1, 0P20 RR020647, R01 GM074258, and Army Research Lab grant DATM-05-02-C-0046. The authors are thankful to the anonymous referees for valuable suggestions.

References

[1] J. Aczel, *Lectures on Functional Equations and Their Applications*, Academic Press, New York, 1966.

[2] C. Anderson and M. D. Dahleh, "Rapid computation of the discrete Fourier transform", *SIAM J. Sci. Computing*, 1996, Vol. 17, pp. 913-919.

[3] R. Araiza, H. Xie, S. A. Starks, and V. Kreinovich, "Automatic Referencing of Multi-Spectral Images", *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, Santa Fe, New Mexico, USA, April 7–9, 2002, pp. 21–25.

[4] C. Bajaj, J. Castrillon-Candas, V. Siddavanahalli, and Z. Xu, "Compressed Representations of Macromolecular Structures and Properties Structure", *Structure*, 2005, Vol. 13, No. 3, pp. 463–471.

[5] C. Bajaj and V. Siddavanahalli, *Fast Error-bounded Surfaces and Derivatives Computation for Volumetric Particle Data*, The University of Texas at Austin, Department of Computer Sciences, Technical Report TR-06-06, January 2006, available as http://www.cs.utexas.edu/ftp/pub/techreports/tr06-06.pdf

[6] G. Beylkin, "On the fast fourier transform of functions with singularities", *Appl. Comput. Harmon. Anal.*, 1995, Vol. 2, pp. 363-381.

[7] R. N. Bracewell, *The Fourier Transform and its Applications*, McGraw-Hill, New York, 1965.

[8] L. G. Brown, A survey of image registration techniques, *ACM Computing Surveys*, 1992, Vol. 24, No. 4, pp. 325–376.

[9] J. Castrillon-Candas, C. Bajaj, and V. Siddavanahalli, *An Adaptive Irregularly Spaced Fourier Method for Protein-Protein Docking*, The University of Texas at Austin, Department of Computer Sciences, Technical Report TR-05-34, July 2005, available as http://www.cs.utexas.edu/ftp/pub/techreports/tr05-34.pdf

[10] Q. Chen, M. Defrise, and F. Deconnick, "Symmetric phase-only matched filtering of Fourier-Mellin transform for image registration and recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1994, Vol. 16, pp. 1156–1168.

[11] A. V. Cideciyan, "Registration in occular fundus images", *IEEE Engineering in Medicine and Biology*, 1995, Vol. 14, pp. 52–58.

[12] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, MIT Press, Cambridge, Massachusetts, 2001.

[13] R. A. Crowther, Procedures for three-dimensional reconstruction of spherical viruses by Fourier synthesis from electron micrographs, *Phil. Trans. Roy. Soc. London*, 1971, Vol. 261, pp. 221–270.

[14] E. De Castro and C. Morandi, "Registartion of translated and rotated images using finite Fourier transforms",

IEEE Transactions on Patten Analysis and Machine Intelligence, 1986, Vol. 9, No. 5, pp. 700–703.

[15] A. Dutt and V. Rokhlin, "Fast Fourier transform for nonequispaced data", *SIAM J. Sci. Computing*, 1993, Vol. 14, pp. 1368-1393.

[16] A. Dutt and V. Rokhlin, "Fast Fourier transform for nonequispaced data. II", *Appl. Comput. Harmon. Anal.*, 1995, Vol. 2, pp. 85-100.

[17] B. Elbel, "Fast Fourier transform for non-equispaced data", In: C. K. Chui and L. L. Schumaker (eds.), *Approximation Theory*, Vanderbilt University Press, 1998.

[18] P. C. Fishburn, *Utility Theory for Decision Making*, John Wiley & Sons Inc., New York, 1969.

[19] P. C. Fishburn, *Nonlinear Preference and Utility Theory*, John Hopkins Press, Baltimore, Maryland, 1988.

[20] H. Foroosh, J. B. Zerubia, and M. Berthod, "Extension of phase correlation to subpixel registration", *IEEE Transcations on Image Processing*, 2002, Vol. 11, pp. 188–200.

[21] S. Gibson, V. Kreinovich, L. Longpré, B. Penn, and S. A. Starks, "Intelligent Mining in Image Databases, With Applications to Satellite Imaging and to Web Search", In: A. Kandel, H. Bunke, and M. Last (eds.), *Data Mining and Computational Intelligence*, Springer-Verlag, Berlin, 2001, pp. 309–336.

[22] L. K. Grover, A fast quantum mechanical algorithm for database search, *Proceedings of the 28th Annual ACM Symposium on the Theory of Computing STOC'96*, 1996, pp. 212–219.

[23] L. K. Grover, From Schrödinger's equation to quantum search algorithm, *Aerican Journal of Physics*, 2001, Vol. 69, No. 7, pp. 769–777.

[24] O. Kosheleva, V. Kreinovich, and H. T. Nguyen, "On the Optimal Choice of Quality Metric in Image Compression", *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, Santa Fe, New Mexico, USA, April 7–9, 2002, pp. 116–120.

[25] V. Kreinovich, A. Lakeyev, J. Rohn, and P. Kahl, *Computational complexity and feasibility of data processing and interval computations*, Kluwer, Dordrecht, 1997.

[26] C. D. Kuglin and D. C. Hines, The phase correlation image alignment method, *Proc. IEEE International Conference on Cebrnetics and Society*, New York, Sept. 1975, pp. 163–165.

[27] T. M. Lehmann, "A two stage algorithm for model-based registration of medical images", *Proceedings of the International Conference on Pattern Recognition ICPR'98*, Brisbane, Australia, 1998, pp. 344–352.

[28] D. R. Luce and H. Raiffa, *Games and Decisions, Introduction and Critical Survey*, John Wiley & Sons, Inc., New York, 1957.

[29] L. Lucchese, G. Doretto, and G. M. Cortelazzo, "A Frequency Domain Technique for Range Data Registration", *IEEE Transactions on Patten Analysis and Machine Intelligence*, 2002, Vol. 24, pp. 1468–1484.

[30] R. B. Myerson, *Game theory. Analysis of Conflict*, Harvard University Press, Cambridge, MA, 1991.

[31] H. T. Nguyen and V. Kreinovich, *Applications of Continuous Mathematics to Computer Science*, Kluwer, Dordrecht, 1997.

[32] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press, Cambridge, Massachusetts, 2000.

[33] F. Pavelcik, J. Zelinka, and Z. Otwinowski, Methodology and applications of automatic electron-density map interpretation by six-dimensional rotational and translational serach for molecular fragments, *Acta Crystallografica D*, 2002, Vol. D58, pp. 275–283.

[34] D. Potts, G. Steidl, and M. Tasche, "Fast Fourier transforms for nonequispaced data: A tutorial", In: J. J. Benedetto and P. J. S. G. Ferreira (eds.), *Modern Sampling Theory: Mathematics and Applications*, Birkhaser, Boston, Massachusetts, 2001, pp. 249-274.

[35] D. Potts, G. Steidl, and A. Nieslony, "Fast convolution with radial kernels at nonequispaced knots", *Numer. Math.*, 2004, Vol. 98, No. 2, pp. 329-351.

[36] W. K. Pratt, *Digital Image Processing*, John Wiley & Sons, New York, 1978.

[37] M. Radermacher, Radon transform techniques for alignment and three-dimensional reonstrution from random projections, *Scanning Microscopy*, 1997, Vol. 11, pp. 171–177.

[38] B. S. Reddy and B. N. Chatterji, "An FFT-based technique for translation, rotation, and scale-invariant image registration", *IEEE Transactions on Image Processing*, 1996, Vol. 5(8), pp. 1266–1271.

[39] C. G. Schiek, *Terrain change detection using aster optical satellite imagery along the Kunlun fault, Tibet*, Master Thesis, Department of Geological Sciences, University of Texas at El Paso, 2004.

[40] C. G. Schiek, R. Araiza, J. M. Hurtado, A. A. Velasco, V. Kreinovich, and V. Sinyansky, "Images with Uncertainty: Efficient Algorithms for Shift, Rotation, Scaling, and Registration, and Their Applications to Geosciences", In: M. Nachtegael, D. Van der Weken, and E. E. Kerre (eds.), *Soft Computing in Image Processing: Recent Advances*, Springer Verlag, Berlin (to appear).

[41] H. Shekarforoush, M. Berthod, and J. Zerubia, "Subpixel image registration by estimating the polyphase decomposition of cross power spectrum", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition CVPR'96*, 1996, pp. 532–537.

[42] C. Bajaj, P. Djeu, V. Siddavanahalli, and A. Thane, "Interactive Visual Exploration of Large Flexible Multicomponent Molecular Complexes", *Proc. of the Annual IEEE Visualization Conference*, October 2004, Austin, Texas, IEEE Computer Society Press, pp. 243–250.

[43] G. Steidl, "A note on fast Fourier transforms for nonequispaced grids", *Advances in Computational Mathematics*, 1998, Vol. 9, pp. 337-352.

[44] E. Suli and A. Ware, "A spectral method of characteristics for hyperbolic problems", *SIAM J. Numer. Anal.*, 1991, Vol. 28, pp. 423–445.

[45] P. Suppes, D. M. Krantz, R. D. Luce, and A. Tversky, *Foundations of Measurement. Vol. II. Geometrical, Threshold, and Probabilistic Representations*, Academic Press, San Diego, CA, 1989.

[46] S. A. Vavasis, *Nonlinear Optimization: Complexity Issues*, Oxford Science, New York, 1991.

[47] A. F. Ware, "Fast approximate Fourier transforms for irregularly spaced data", *SIAM Rev.*, Vol. 40, No. 4, pp. 838–856.

[48] H. Xie, N. Hicks, G. R. Keller, H. Huang, and V. Kreinovich, "Implementation, test, and analysis of an automatic image registration algorithm based on FFT and IDL/ENVI", *Computers and Geosciences*, 2003, Vol. 29, No. 8, pp. 1045–1055.

[49] B. Zitová and J. Flusser, Image registration methods: a survey, *Image and Vision Computing*, 2003, Vol. 21, pp. 977–1000.