Fast Clifford Fourier Transformation for Unstructured Vector Field Data

Michael Schlemmer [1]
Ingrid Hotz [2]
Vijay Natarajan [2]
Bernd Hamann [2]
Hans Hagen [1]

[1] Computer Graphics and Visualization
Department of Computer Science
University of Kaiserslautern
D – 67653 Kaiserslautern

[2] Institute for Data Analysis and Visualization (IDAV)
Department of Computer Science
University of California, Davis
Davis, Ca 95616

schlemmer@informatik.uni-kl.de

Abstract

Vector fields play an important role in many areas of computational physics and engineering. For effective visualization of vector fields it is necessary to identify and extract important features inherent in the data, defined by filters that characterize certain "patterns". Our prior approach for vector field analysis used the Clifford Fourier transform for efficient pattern recognition for vector field data defined on regular grids [1,2]. Using the frequency domain, correlation and convolution of vectors can be computed as a Clifford multiplication, enabling us to determine similarity between a vector field and a pre-defined pattern mask (e.g., for critical points). Moreover, compression and spectral analysis of vector fields is possible using this method. Our current approach only applies to rectilinear grids. We combine this approach with a fast Fourier transform to handle unstructured scalar data [6]. Our extension enables us to provide a feature-based visualization of vector field data defined on unstructured grids, or completely scattered data. Besides providing the theory of Clifford Fourier transform for unstructured vector data, we explain how efficient pattern matching and visualization of various selectable features can be performed efficiently. We have tested our method for various vector data sets.

Keywords: Fourier transformation, unstructured grids, scattered data, Clifford algebra

Introduction

The analysis and visualization of unstructured vector field data is a challenging task. Basically, two different approaches exist to visualize vector fields: visualization of an entire dataset, or reduction of the dataset by extracting features. The first class of visualization methods provides an overview of a dataset; the second class allows one to concentrate on certain features being of special interest. With increasing size of data sets, feature extraction becomes more and more important. Features of interest in vector fields include vortices and shock waves. Feature extraction as from scalar data, e.g. edge detection, is a well studied branch in image processing. Pattern recognition is performed by convolution of images with specially defined filter masks. For fast detection of such patterns the Fourier transformation plays an important role, since it enhances the convolution operation. A recently presented method for the application of Fourier transformation to vector fields is using the properties of Clifford algebra [1,2]. For a fast calculation, the Clifford fast Fourier transformation (FFT) has been developed, operating on uniformly distributed data [1]. Our main contribution is the combination of this Clifford FFT for vector fields with methods for a non-uniform FFT, operating on arbitrarily distributed scalar data, as proposed by Fourmont [6] and Kunis/Potts [14].

In the following sections, we present the theory for the non-uniform fast Clifford Fourier transformation (NFCFT) and show its application to unstructured vector data.

Related work

Besides direct visualization of vector fields using hedgehogs, for example, a feature-based approach is divided into two steps. The first step is to find patterns of interest, the second visualizes this preprocessed and simplified data. An example for a feature-oriented method is the algorithm of Sujudi and Haimes [18], which extracts vortex core lines by analyzing the eigenvalues and eigenvectors of the velocity gradient tensor. More feature-based visualization methods are discussed by Post et al. [19].

Another possibility for feature-based visualization of vector fields uses signal and image processing techniques for pattern recognition. Prior work introduced a convolution operator for pattern recognition applied to uniform vector field data, see Heiberg et al. [17], Granlund/Knutson [16], and Ebling/Scheuermann [3]. The latter method is based on Clifford algebra and was also applied to non-uniform data [4]. Expensive convolution in spatial domain is reduced to a multiplication in frequency domain. In signal processing it is common to filter the data in frequency domain. To devise a similar method for vector fields we adapted a continuous and discrete Fourier transformation for multi-vector field data by using a Clifford algebra approach [1,2]. We implemented the discrete CFT using the FFT for regular grids. Unfortunately, this method is based on a regular grid structure and cannot be used for arbitrary meshes.

There has been some work concerning the development of fast algorithms for the Fourier transformation on irregular grids (NFFT). We extended this work to CFT. Our work is mainly based on a method by Fourmont [6] and Kunis/Potts [14] for calculating a fast and accurate FFT for non-uniformly spaced data. Our implementation of the fast Clifford Fourier transformation uses a NFFT library developed by Potts et al. [13].

Basics

We start with a brief review of the basics and motivate our work. After an introduction of the CFT we discuss existing methods for NFFTs.

Feature-based Visualization of Vector Fields

Convolution was modified to be applicable for vector valued data. Scientists have defined convolution for vector fields, e.g., Heiberg et al. [17] or Granlund and Knutson [16] using component-wise convolution. A very elegant approach using Clifford algebra was provided by Ebling and Scheuermann [3], introducing the Clifford convolution (CFT). In contrast to other methods, Clifford multiplication and Clifford convolution preserve the full information, magnitudes as well as directions of a vector dataset.

Clifford algebra operates on multi-vectors. These can be regarded as an extension of the complex numbers to vector fields, completed by a complex scalar part. Regarding vectors in three-dimensional Euclidian vector space, we obtain an eight-dimensional algebra G_3 with the basis $\{1, e_1, e_2, e_3, e_2e_3, e_1e_3, e_1e_2, e_1e_2e_3\}$ using the rules of the 3D-Clifford algebra, i.e.,

1. $1e_k = e_k,\ k = 1,2,3$,
2. $e_ke_k = 1,\ k = 1,2,3$,
3. $e_ke_l = -e_le_k,\ k \neq l$,

the Hodge-duality can be derived:

$$e_1e_2 = e_1e_2e_3e_3 = i_3e_3, e_3e_1 = i_3e_2, e_2e_3 = i_3e_1,$$

where

$$i_3 = e_1e_2e_3 \text{ and } i_3^2 = i^2 = -1 \quad .$$

Further information regarding Clifford algebra can be found in Scheuermann [5].

The Clifford product of two vectors is a combination of the inner and outer product and therefore contains angular information as well as the relation of vector lengths. Thus, the so-called Clifford convolution is a suitable approach for pattern matching in vector field data. According to [2] it is defined as

$$c_n(r) = \int\int\int_\Omega P_n(\xi)U(r-\xi)|d\xi|$$

for a multi-vector field ***P*** and filter mask ***U*** in direction ***n***. Since the Clifford product is only commutative for odd dimensions, one has to consider that there is a difference when applying a filter from the left or the right side for even dimensions.

Clifford Fourier Transformation

Clifford convolution can be enhanced by a transformation into frequency space. We have developed the Clifford Fourier transformation as an extension of the common Fourier transformation for vector fields. It can be defined continuously for a three-dimensional multi-vector valued function ***f***: $E^3 \rightarrow G_3$ as

$$\mathcal{F}\{f\}(u) = \int_{I\!R^3} f(x) e^{(-2\pi i_3 <x,u>)} |dx|$$
,

where i_3 is an extension of the imaginary number i in the Clifford algebra [1,2]. The vectors ***x*** and ***u*** indicate position in spatial and frequency domain, respectively. It can be generally defined for any dimension d. This definition varies from the original one only in the fact that we use multi-vectors instead of scalars and that it is defined to be multidimensional.

Especially important for our application is the linearity property of the Fourier transformation. Using the Hodge duality, any three-dimensional multi-vector field ***f***: $E^3 \rightarrow G_3$ can be written as four complex signals, i.e.

$$\begin{aligned} f(x) &= [f_0(x) + f_{123}(x) i_3]1 \\ &+ [f_1(x) + f_{23}(x) i_3] e_1 \\ &+ [f_2(x) + f_{31}(x) i_3] e_2 \\ &+ [f_3(x) + f_{12}(x) i_3] e_3 . \end{aligned}$$

Considering linearity of the Fourier transformation, one obtains

$$\begin{aligned} \mathcal{F}\{f\}(u) &= [\mathcal{F}_c\{f_0(x) + f_{123}(x) i_3\}(u)]1 \\ &+ [\mathcal{F}_c\{f_1(x) + f_{23}(x) i_3\}(u)] e_1 \\ &+ [\mathcal{F}_c\{f_2(x) + f_{31}(x) i_3\}(u)] e_2 \\ &+ [\mathcal{F}_c\{f_3(x) + f_{12}(x) i_3\}(u)] e_3 . \end{aligned}$$

This separation applies to multi-vector fields of arbitrary dimension d, thus Clifford Fourier transformations can be computed by calculating several common Fourier transformations. In our context, we require two transformations for a two-dimensional and four transformations for a three-dimensional Clifford transformation.

We have implemented a fast discrete Clifford Fourier transformation. It is applicable to uniform grids [1], providing a possibility for fast convolution in frequency domain. It also provides insight into the structure of the frequency domain of a vector field. We have used this approach to apply a variety of different filters, e.g., low pass, high pass, band pass, and vector valued filters (i.e. rotations, divergences) and have obtained satisfying results. Unfortunately, this technique is limited to uniform grids. An example for a Clifford Fourier transformed vector data set is presented in Figure 1, whereas examples for vector valued filters and their frequency representation are illustrated in Figure 2.

The two most obvious ways to treat data on irregular grids is either resampling or defining the filter mask according to the local grid structure, compare Ebling and Scheuermann [4]. We present the NFCFT, to enhance these spatial domain approaches by transforming unstructured vector data into frequency domain.

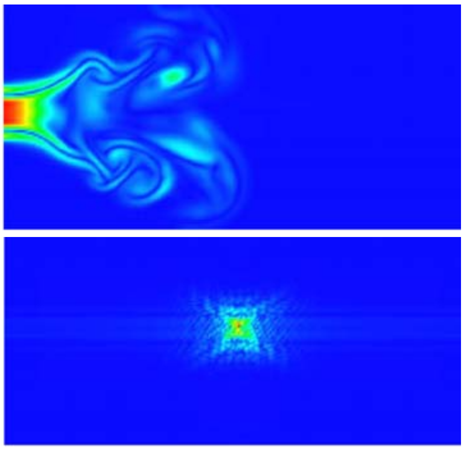

Figure 1: [1]
Top: Magnitude representation of swirling jet entering a liquid at rest (vector data, rectilinear, resampled to uniform grid)
Bottom: Magnitude frequency representation (transformed by a CFT)

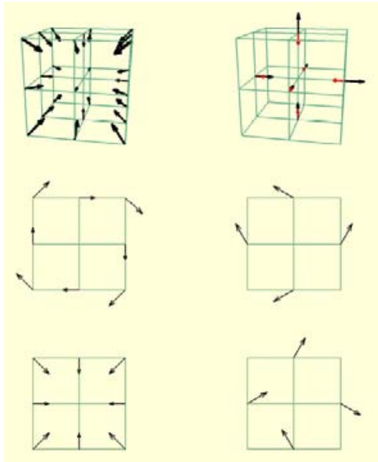

Figure 2: [2]
Left: examples for vector valued filters in spatial domain, two- and three-dimensional.
Right: The corresponding Clifford frequency domain representation.

Non-uniform FFT for scalar data:

Starting in the mid 90s with investigations by Dutt and Rohklin [7,8], it is still an active research area. There are basically two types of non-uniform Fourier transforms. The non-uniform discrete Fourier transform (NDFT) is defined as transformation from N evenly distributed data points evaluated at M arbitrary positions in frequency domain [6], i.e.,

$$\hat{z}_l = \sum_{k=-N/2}^{N/2-1} e^{\frac{-2\pi i x_l k}{N}} z_k, \quad l = 1, ..., M \quad .$$

Since one only has to recalculate the Fourier basis location, there is no need for an approximation of the data. The approximate inverse transformation is defined similarly, using interpolation to calculate the correct Fourier modes:

$$\hat{z}_k = \sum_{l=1}^{M} e^{\frac{-2\pi i x_l k}{N}} z_l, \quad k = -\frac{N}{2}, ..., \frac{N}{2} - 1 \quad .$$

Kunis and Potts presented an adjustable algorithm for a high-accuracy approximation to this problem [16].

Typical implementations of the non-uniform fast Fourier transformations (NFFT) use a windowing function to approximate the Fourier modes for fast calculation. Various authors proposed different possibilities for these windowing functions. While Beylkin [9] uses a B-spline window, Dutt and Rohklin [7,8] use a Gaussian window, which was further optimized and by Steidl [12]. Ware [10] compared these methods. Further improvements and windowing approaches were also proposed by Duijndam and Schonewille [11]. Our work is based on the ideas of Fourmont [6], using Kaiser-Bessel windowing. He showed the effectiveness for these window approximations resulting in very small errors. Using Shannon's theorem for band-limited functions, it can be shown [6] that

$$e^{-ix\xi} = \frac{1}{\sqrt{2\pi}\phi(\xi)} \sum_{m\in\mathbb{Z}} \hat{\phi}(x-m)e^{-im\xi}, \quad |\xi| < \frac{\pi}{c} \quad ,$$

for an interpolation function $\phi \in \mathcal{C}_0^\infty, \phi > 0$, with support in $[-\pi, \pi]$. Thus, the NFFT with non-uniform result is given by inserting this equation into the transformation [6]:

$$\hat{z}_l = \frac{1}{\sqrt{2\pi}} \sum_{m\in\mathbb{Z}} \hat{\phi}(cx_l - m) \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} e^{\frac{-2\pi imk}{cN}} \frac{z_k}{\phi(\frac{2\pi k}{cN})}, \quad l = 1, ..., M \quad .$$

Considering non-uniformly spaced data, the (simple) transformation is defined as

$$\hat{z}_k = \frac{1}{\sqrt{2\pi}} \frac{1}{\phi(\frac{2\pi k}{cN})} \sum_{l=1}^{M} \sum_{m\in\mathbb{Z}} z_l \hat{\phi}(cx_l - m)e^{\frac{-2\pi imk}{cN}}, \quad k = -\frac{N}{2}, ..., \frac{N}{2} \quad .$$

The quality of the windowing function ϕ depends on its concentration in spatial and frequency domain. It is impossible to find a function for exact reconstruction, since any band-limited function has to be infinite in the spatial domain and vice versa. The Gaussian window seems to be the best choice, since it is similar, or even equal, in spatial as well as in frequency domain and minimizes error in both domains. Fourmont's Kaiser-Bessel window turns out to be a better choice. It provides compact support over the window span in the spatial domain, contributing no error, and minimizes the error when limiting the infinite frequency representation. For more information on Kaiser Bessel windows, we refer to Kaiser [15], for their application to the NFFT algorithm, we refer to Fourmont [6].

Non-uniform fast Clifford Fourier Transformation

Our main contribution is the combination of our discrete fast Clifford Fourier transformation [1] developed for uniformly distributed data with an approach for non-uniform fast Fourier transformation, following the ideas of Fourmont [6]. We define the NFCFT transformation as a transformation of non-uniformly distributed data in the spatial domain to evenly spaced data in frequency domain:

$$f_l = \frac{1}{\sqrt{2\pi}} \frac{1}{\phi(\frac{2\pi l}{cN})} \sum_{u=1}^{M} \sum_{m\in\mathbb{Z}^3} \hat{f}_u \hat{\phi}(cx_l - m) e^{\frac{-2\pi i_3 <l,m>}{cN}} ,$$

for $v \in \mathbb{R}^d$: $\phi(v) = \phi(v_1)...\phi(v_d) \quad and \quad \hat{\phi}(v) = \hat{\phi}(v_1)...\hat{\phi}(v_d)$.

The inverse transformation (INFCFT) is an extension of the NFFT:

$$\hat{f}_u = \frac{1}{\sqrt{2\pi}} \sum_{m\in\mathbb{Z}^3} \hat{\phi}(cx_l - m) \sum_{l_1=-\frac{N_1}{2}}^{\frac{N_1}{2}-1} \sum_{l_2=-\frac{N_2}{2}}^{\frac{N_2}{2}-1} \sum_{l_3=-\frac{N_3}{2}}^{\frac{N_3}{2}-1} f_l \frac{1}{\phi(\frac{2\pi l}{cN})} e^{\frac{2\pi i_3 <l,m>}{cN}} .$$

By considering the results of Ebling and Scheuermann [2], we can split up the calculation into four scalar-valued NFFTs for three-dimensional and into two for two-dimensional multi-vector fields.
Thus, we can use the NFFT library of Potts et al. [13] to calculate the scalar NFFTs. Moreover, for our NFCFT we used the simple inversion by Fourmont and the more accurate iterative approach developed by Potts and Kunis [16]. Both algorithms for the NFCFT have been compared, considering time and accuracy. We have applied the methods to several data sets, performing the transformation and its inverse. Comparing directions and magnitudes of the resulting vectors to the original ones, we have computed accuracy measurements for vector-valued data. A full reconstruction of a field is only possible when satisfying the Nyquist theorem, i.e., for an appropriate reconstruction at positions lying very closely to each other, we need to use a high over-sampling rate.

An important application of this Fourier approach is the convolution of vector-valued filters and non-uniformly distributed vector-valued data by performing a Clifford multiplication in frequency domain. We first transform a vector field onto a uniform grid in frequency domain, using the simple NFCFT, similar to Fourmont's definition of the INFFT [6], and the high accuracy method of Kunis and Potts [16]. Since our frequency representation is based on a uniform grid, we are able to use the frequency representation of non-interpolated convolution masks. Multi-vector field and filter mask are multiplied in frequency domain. INFCFT of the resulting multi-vector finally produces the filtered multi-vector field. In case of a vector-valued filter mask, we obtain a scalar-valued field, indicating the similarity of the field to the used filter mask at each position.

Results

The NFCFT has been implemented and tested, using a 2.6 GHz Pentium 4 processor with 512 MB RAM. The algorithm was applied to an unstructured vector data set (Figure 3), measuring accuracy (Tables 1, 2) and time dependency (Table 3), considering the number of iterations and over-sampling factors used for the transformation. The over-sampling factor indicates the number of positions in frequency respectively to the number of positions in spatial domain. Accuracy was measured by comparing the original data set with its inversely transformed frequency representation. We distinguish between relative error in vector magnitude and the angular error.

The results show that combining over-sampling and the iterative improvement of Kunis and Potts [14] leads to high reconstruction quality, whereas Fourmont's method (equivalent to performing just one iteration) does not lead to accurate results. The tables show, that the angular error decays faster than the magnitude error. This is of advantage for the application of vector pattern recognition, since rotations or divergences are defined by directions of vectors. For practical considerations, the performance of the inverse transformation is more important, since a data set can be transferred into frequency domain once, and can then be filtered with various filters. The results of these filtering operations are all transformed with the inverse transformation. The transformation into frequency domain can be regarded as preprocessing for an efficient filtering in frequency domain.

OF	1 it.	3 it.	5 it.	10 it.	15 it.	20 it.	25 it.	30 it.
1	37.26	34.23	30.68	26.20	24.30	22.96	22.12	21.46
4	22.51	15.45	9.79	5.34	3.51	2.58	1.71	1.32
16	10.92	4.55	2.78	1.21	0.77	0.42	0.32	0.16
64	4.66	1.68	0.71	0.34	0.09	0.05	0.02	0.0006

Table 1: Relative medium error in magnitude depending on over-sampling factor and no. of iterations [%] for test data set with n=2500 vectors. Formula: $100 * \sum_n (| |v|-|v_t| | / n)$

OF	1 it.	3 it.	5 it.	10 it.	15 it.	20 it.	25 it.	30 it.
1	23.46	18.21	16.18	13.76	12.58	11.81	11.35	11.00
4	11.72	5.57	3.95	1.90	1.15	0.78	0.52	0.41
16	4.97	1.42	0.67	0.30	0.22	0.11	0.08	0.05
64	1.94	0.531	0.227	0.067	0.026	0.011	0.0005	0.0001

Table 2: Relative medium angular error depending on over-sampling factor and no. of iterations [%] for test data set, n=2500. Formula: $100 * \sum_n \cos^{-1}(\langle v,v_t \rangle / |v|*|v_t|) / (\pi * n)$.

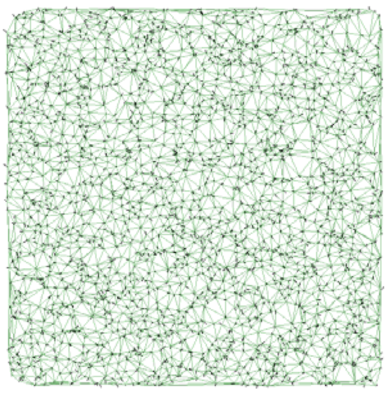

Figure 3: Hedgehog representation of completely unstructured test data

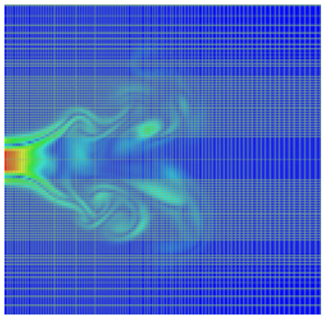

Figure 4: Structured non-uniform grid and magnitude representation of swirling jet entering a liquid at rest (vector data, rectilinear, non-uniform spacing)

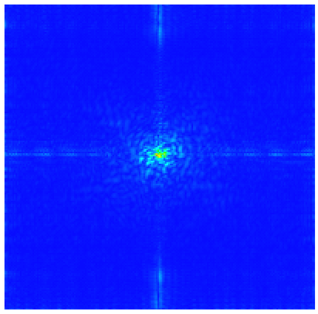

Figure 5: Frequency representation of swirling jet data set, low frequency magnitudes stronger (center)

OF	1 it.	3 it.	5 it.	10 it.	15 it.	20 it.	25 it.	30 it.	Inv
1	0.03	0.3	0.45	0.85	1.25	1.71	2.03	2.48	0.03
4	0.08	0.75	1.16	2.19	3.26	4.3	5.33	6.41	0.07
16	0.25	2.81	4.32	7.93	12.1	15.95	19.67	22.57	0.26
64	1.06	11.31	17.31	32.23	47.15	62.09	77.16	94.5	1.07

Table 3: Calculation times [sec] depending to over-sampling factor (OF) and number of iterations for test data set consisting of 2500 vectors

We applied the algorithm to a real world data set, a two-dimensional slice of a swirling jet vector field, entering a fluid at rest (Figure 4). With an over-sampling factor of approx. 5, mapping 12524 vectors in spatial domain to 256*256 in frequency domain, the computation with 100 iterations took 106 seconds, while inverse transform took 0.35 seconds. This data set is not unstructured, but it is not defined on a uniform grid. The frequency representation of the swirling jet data set shows the expected larger magnitude values in the lower frequency spectrum (Figure 5).

Conclusions and Future Work

We have presented a generalization of the fast Clifford Fourier transformation and compared the accuracy and efficiency of two different implementations of our approach. This method, based on the discrete fast Clifford Fourier transformation for uniform grids [1] and the NFFT for scalar data of Fourmont and Kunis/Potts [6,14], was developed to provide an alternative to other methods for pattern matching for unstructured vector field data [4].
Our method performs well for most unstructured vector fields, but we need to develop approaches to assess uncertainty. Having large numbers of vectors concentrated in specific sub-areas, it can happen that outliers cause high similarity values for specific filters. There is need for proper uncertainty measures to indicate the significance of matches. A possible way for improving errors is to perform some preliminary segmentation and compute the transformation on each segment.

Acknowledgements

We especially like to thank Gerik Scheuermann, University of Leipzig, for providing the FAnToM system to visualize our data and all persons involved in the development of this tool. We also thank Wolfgang Kollmann, Mechanical and Aeronautical Engineering Department, University of California, Davis, for providing the swirling jet data set. We thank Daniel Potts and Stefan Kunis, University of Lübeck, for supplying the NFFT library. We also would like to thank the members of the Computer Graphics and Visualization Research Groups in Kaiserslautern and Davis, as well as the members of the International Research Training Group (IRTG) for fruitful discussions, especially Torsten Bierz, Christoph Garth, Burkhart Lehner, Ariane Middel and Marco Schneider. The IRTG is supported by the German Research Foundation (DFG) under contraction.

References

[1] M.Schlemmer, *Fourier Transformation and Filter Design for Clifford Convolution*, Diplomarbeit (Master Thesis), University of Kaiserslautern, 2004. (Available at http://www-hagen.informatik.uni-kl.de/~schlemmer)

[2] J.Ebling and G.Scheuermann, *Clifford Fourier Transformation on Vector Fields*, in "Proceedings of Parallel Visualization and Graphics 2003", 2003

[3] J.Ebling and G.Scheuermann, *Clifford convolution and pattern matching on vector fields*. In "Proceedings of IEEE Visualization 2003", 2003.

[4] J.Ebling and G.Scheuermann, *Clifford Convolution and Pattern Matching on Irregular Grids*. In "Proceedings of Dagstuhl 2003", 2003

[5] G.Scheuermann, *Topological Vector Field Visualization with Clifford Algebra*. PhD thesis, University of Kaiserslautern, Germany, 1999.

[6] K.Fourmont, Non-*Equispaced Fast Fourier Transforms with Applications to Tomography*, Journal of Fourier Analysis and Applications Vol.9, Issue 5, 2003.

[7] A.Dutt and V.Rohklin, *Fast Fourier transforms for nonequispaced data*. SIAM Joumal on Scientific and Statistical Computing 1993;14: 1368-1393.

[8] A.Dutt and V.Rohklin, 1995. *Fast Fourier Transforms for non-equispaced data, II*: App. Comp.Harm. Anal., 2, 85-100.

[9] G.Beylkin, *On the fast Fourier transform of functions with singularities*, Appl. Comput. Harmon. Anal., 2 (1995), pp. 363-381.

[10] A.Ware, *Fast approximate Fourier transforms for irregularly spaced data*. SIAM Rev., 40:838 - 856, 1998.

[11] A.Duijndam and M.Schonewille, *Nonuniform fast Fourier transform*. *Geophysics*, 64:539 - 551, 1999.

[12] G.Steidl, *A note on fast Fourier transforms for nonequispaced grids*. Adv. Comput. Math., 9:337 - 353, 1998.

[13] D. Potts, G. Steidl, and M. Tasche, Fast Fourier transforms for nonequispaced data: A tutorial. In "*Modern Sampling Theory: Mathematics and Applications*, pages 247 - 270, Boston, 2001. Birkhäuser.

[14] S. Kunis and D. Potts. *Stability results for scattered data interpolation by trigonometric polynomials*. Preprint, Univ. Lübeck, A-04-12, 2004.

[15] Kaiser, J.F. (1966), *Digital filters*, in "System Analysis by Digital Computers", Kuo, F. and Kaiser, J.F., Eds., 218–285.

[16] G.Granlund and H.Knutsson, 1995. *Signal Processing For Computer Vision*. Kluwer Academic Publishers, Dordrecht, The Netherlands.

[17] E.Heiberg, T.Ebbers, L.Wigström, M.Karlson 2003. *Three Dimensional Flow Characterization Using Vector Pattern Matching*. In "IEEE Transactions on Visualization and Computer Graphics 9 (3)", David S. Ebert (ed.) 319.

[18] D.Sujudi and R.Haimes, *Identification of Swirling Flow in 3-D Vector Fields*, Technical Report, Department of Aeronautics and Astronautics, MIT, Cambridge, MA, 1995

[19] F.H.Post, B.Vrolijk, H.Hauser, R.S.Laramee, H.Doleisch, 2002. *Feature Extraction and Visualization of Flow Fields.* In Eurographics 2002 State of the Art Reports, The Eurographics Association, Saarbrücken, Germany, pp. 69-100.