Fast and Exact Fourier Volume Rendering

Syed Alam Abbas, *Student Member, IEEE*, Ulas Bagci and Hassan Foroosh, *Senior Member, IEEE*

***Abstract*—Fourier volume rendering (FVR) is a significant visualization technique that has been used widely in digital radiography. It plays a significant role in medical imaging as it allows exploring the internal structures of volumetric data acquired by different imaging modalities. This exploration allows accurate diagnosis and consequently effective treatment. It relies on a sound *Fourier projection-slice theorem*, where using the spectral representation of a 3D spatial data and taking a central 2D slice generates attenuation-only projections that looks like *X-ray radiographs*. We propose an algebraically exact and fast algorithm to compute such a central slice with arbitrary orientation (angle) which has the special feature that it involves only 1D equispaced FFTs and no interpolation or approximation at any stage. The result is we get the fast solution within machine precision without any oversampling and devoid of any artifacts. We also propose an exact and fast algorithm to simultaneously compute multiple planar projections that are angularly equispaced using 3D discrete Cylindrical Fourier transform.**

***Index Terms*—Fourier Volume Rendering, Projection slice theorem, 2D projections, Volumetric data, Cylindrical Fourier transform.**

I. Introduction

Volume visualization or rendering is an essential tool for exploring and analysing the anatomy of complex structures and phenomena, it is used in various scientific and engineering arenas such as medical imaging, geoscience, microscopy, mechanical engineering, and others [1]-[3]. An extensive review of the classical theory behind FVR can be found in [4],[5],[6]. Volume rendering algorithms can be classified into spatial-domain and other-domain-based techniques such as frequency domain, compression domain or the wavelet domain. In this paper we discuss a new way to perform Fourier domain based volume rendering. This technique has considerable significance in the medical discipline for the following reasons: 1. It produces attenuation renderings that look like the *X-ray radiographs* which the medical professionals are well trained to interpret, 2. It can be used to get a sliced view of the full volume data along any direction for medical analysis, 3. Due to its fast implementation using FFTs it can be suitable for visualizing of large scale data sets.

FVR uses a 3D spectral representation of the volume to compute an image that looks like *X-ray radiograph* relying on the projection-slice theorem. It works by transforming the spatial volume into frequency domain. Then it reconstructs 2D projection at any viewing angle by resampling an extracted projection-slice along a perpendicular plane to the viewing direction followed by backtransforming this resampled slice to the spatial domain [5],[7],[8] and [9]. However practical implementation of the FVR technique is complicated by three main factors which arise when the projection slice theorem is applied to discrete sampled data, [7] and [9]. In a typical FVR algorithm a projection slice is extracted from the spectrum at certain angle specified by user. This stage is followed by an optional resampling stage to reduce the aliasing and ghosting artifacts from the final projection. The interpolated slice is then back-transformed by inverse 2D FFT operation, and the resulting projection image is wrapped-around to set its origin at the center of the image. First when the projection slice is desired along an arbitrary direction conventional FFT algorithm cannot be applied to yield a plane in the desired angle so interpolations must be used. To reduce the aliasing and ghosting artifacts in the reconstructed image, this projection slice is processed in a further step to have it resampled. The resampling stage is only mandatory if the desired projection was not orthogonal. Second the FFT algorithms yield frequency-domain output data which is not ideally structured for resampling. To mitigate this effect, it is necessary to add high performance multidimensional fft-shift stages to the rendering pipeline to rearrange the data. Moreover, sampling the frequency domain is equivalent to the replication of the signal in the spatial domain that ultimately accounts for the appearance of ghosting artifacts in the reconstructed images. This issue is resolved by zeropadding the volume in the spatial domain and using highorder interpolation filters in the frequency domain. Finally there is no straightforward way for direct extraction of the complex projection slice from the 3D spectrum [10] so the entire set of operations must be performed two times one for real part and one for complex part of the spectral 3D data.

There exists no fast and exact solution to the FVR technique. In literature, several volume rendering algorithms have been presented either to improve the rendering speed of large datasets or to enhance the rendering quality of their reconstructed images. By and large, the rendering speed and quality are traded-off and no single complete algorithm that can deliver the optimum quality associated with maximum speed exists [11]. Recently the authors introduced a fast and exact solution to the discrete polar and spherical polar transform [12]. We use a similar strategy by using 1D Fractional Fourier transform (FrFT) but tailored specifically for our FVR algorithm which gives us the only known solution that is both fast and algebraically exact.

All the techniques proposed in the literature are approximate and based on interpolations. In this paper we propose an exact FVR scheme of low complexity and we show the parallel implementation of our algorithm that is fast, exact and fully vectorized. This paper is organized as follows: Section II discusses the mathematical tools required, the algorithm

The authors are with the Department of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL, 32816, USA. e-mails: (syedalamabbas@knights.ucf.edu, syedalamabbas@gmail.com, foroosh@eecs.ucf.edu,).

Manuscript received May 16, 2016.

implementations and its computational complexity. Section III has the experimental section where we demonstrate the superiority of the exact projection scheme over the interpolated ones, followed by conclusions in section IV.

II. Proposed Method

A. Preliminaries: 1D Fractional Fourier Transform

Given a discrete 1D signal $f(n)$ over the support $-N/2 \leq n \leq N/2$, where N is even, and given an arbitrary scalar value α, the definition of 1-D fractional Fourier transform (FrFT) (also known as Chirp-Z transform) [13] is given by,

$$F^{\alpha}(k) = \sum_{n=-N/2}^{N/2} f(n) e^{-i\frac{2\pi k\alpha n}{N+1}}, -N/2 \leq k \leq N/2 \tag{1}$$

There is a fast algorithm for this that computes the above 1D FrFT in the order of $(N+1)\log_2(N+1)$ [13], [14]. Consider the factor in the exponent, $2kn = k^2 + n^2 - (k-n)^2$, we have,

$$\begin{aligned} F^{\alpha}(k) &= \sum_{n=-N/2}^{N/2} f(n) e^{-\frac{i\pi\alpha}{N+1}(k^2+n^2-(k-n)^2)} \\ &= e^{-\frac{i\pi\alpha}{N+1}k^2} \sum_{n=-N/2}^{N/2} f(n) e^{-\frac{i\pi\alpha}{N+1}n^2} e^{\frac{i\pi\alpha}{N+1}(k-n)^2} \end{aligned}$$

Let us define a sequence,

$$E(n) = e^{-\frac{i\pi\alpha}{N+1}n^2}, -N/2 \leq n \leq N/2 \tag{2}$$

then the above equation reduces to,

$$F^{\alpha}(k) = E(k) \sum_{n=-N/2}^{N/2} f(n)E(n)E(k-n), \quad -N/2 \leq k \leq N/2 \tag{3}$$

To evaluate the transform (1) using FFT for convolution in (3), we multiple $f(n)$ by $E(n)$ and compute its FFT, then we take FFT of $E(n)$ and multiply it to compute the convolution. Then we take 1 more IFFT to the product of 2 FFTs and post multiply it with $E(k)$ which gives us the desired sequence. As noted in [14], the FFT should be computed with zero padding to length $2(N+1)$ in order to avoid cyclic effects. A $(N+1)$ point 1-D FFT can be evaluated with operational cost $5(N+1)\log_2(N+1)$. Since three 1D-FFT's are needed, and since the zero padded sequence length is $2(N+1)$, we have a total operation cost of evaluating (1) as $30(N+1)\log_2(N+1)$ floating point operations (flops).

B. Algorithm I: Single Planar Projection

We can define a plane oriented at an arbitrary angle ϕ in the X-Y plane see Fig 1. Then the 1-D frequency domain scaling of a 3-D image along X-axis can be computed using 1D FrFT along X-axis such that,

$$\begin{gathered} F_x^{\alpha}(r,c,d) = \sum_{n=-N/2}^{N/2} f(n,c,d) e^{-j\frac{2\pi r\alpha n}{N+1}} \\ \alpha = \cos\phi, \\ r,c,d = -\tfrac{N}{2}, \cdots, \tfrac{N}{2} \end{gathered} \tag{4}$$

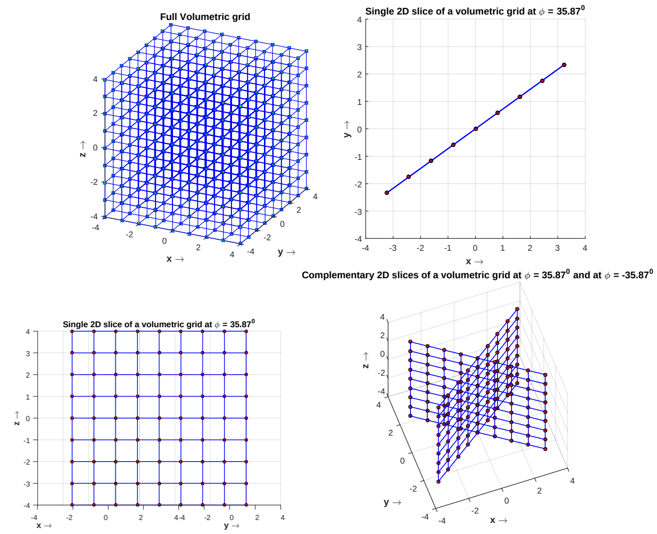

Fig. 1. Direct and exact 2D slices computation of a full 3D medical volumetric data. Clockwise from top left: (a)3D Volume data such as CT scan (b) Top view or X-Y view of the 2D slice (c) 2D grids of complementary slices with respect to X-axis (d) Orthographic view of the slice which is a 2D equi-spaced normal grid.

Next we can scale the image f, in Y-axis, we can compute 1D FrFT and the equation is given by,

$$\begin{gathered} F_y^{\alpha,\beta}(r,c,d) = \sum_{n=-N/2}^{N/2} F_x^{\alpha}(r,n,d) e^{-j\frac{2\pi c|c|\beta n}{(N+1)}} \\ \beta = \sin\phi, \\ r,c,d = -\tfrac{N}{2}, \cdots, \tfrac{N}{2} \end{gathered} \tag{5}$$

Finally we scale the transformed image in Z-axis,

$$\begin{gathered} F_z^{\alpha,\beta,\gamma}(r,c,d) = \sum_{n=-N/2}^{N/2} F_y^{\alpha,\beta}(r,c,n) e^{-j\frac{2\pi d\gamma n}{(N+1)}} \\ \gamma = 1, \\ r,c,d = -\tfrac{N}{2}, \cdots, \tfrac{N}{2} \end{gathered} \tag{6}$$

Using the (4) and (5) we gather points falling on the 2D plane which is the 2D projection-slice passing through the origin of the spectral volume at an angle ϕ. This carefully extracted slice represents the 2D FFT of the desired projection and thus, the reconstructed image can be directly obtained by a 2D inverse FFT operation.

The last equation (6) is nothing but centered FFT along Z-axis. By simply rotating the entire volume, and performing the operations using the above equations we can compute central slices along any direction. Thus any arbitrary plane with angle ϕ, can be computed using systematic frequency domain scaling of the spatial volume grid.

This gives us the true 2D projection solution that is algebraically exact. It only involves the use of 1D FFT's and hence it is fully vectorizable, memory efficient and fast. It does not involve any design parameter choice for user, it involves no preprocessing, oversampling or padding of any kind. The computation of the 1D FrFT of the volumetric data is also completely independent, and it is easy to compute each vector in a parallel way on a GPU, multi-CPU, multi-GPU and grid computing (CPU/GPU) systems.

1) Computational Complexity: The most expensive operation is the first uniform scaling by α along X axis. We have the complexity for X axis scaling of the full 3D volumetric grid using real data,

$$(N+1)^3/2 \times \log_2(N+1)$$

Next scaling is in Y-axis and is differential scaling. Since we are interested in just point on the 2D plane oriented along a specified direction this operation simply reduces to a Hadamard element wise multiplication, hence we have the reduced complexity,

$$(N+1)^3$$

Finally we compute 1D FFT on a 2D plane along Z -axis which is uniform and requires no scaling, the computational complexity of 1D-FFT is then,

$$(N+1)^2 \times \log_2(N+1)$$

Hence the total computational complexity is dominated by the first uniform scaling for N large, so the total computational complexity for each slice at an arbitrary angle ϕ is,

$$(N+1)^3/2 \times \log_2(N+1) \tag{7}$$

Note that during this computation the complementary slice planes at angles $180 \pm \phi$ for X-axis orientation or basically horizontal planes or $90 \pm \phi$ for Y-axis orientation or basically vertical planes can also be computed without any overhead, which bring us on to the topic of multiple simultaneous projections.

C. Algorithm II: Multiple Planar Projections

Given a rotation axis $\vec{n}$ that is parallel to one of the principle axis X,Y and Z, we can define the multiple planar projections conveniently using cylindrical coordinates representation, see figure 2. Consider first the Z-axis, this arrangement of cylindrical coordinates could be considered as exact 2D polar slices stacked along the height of the cylinder. We define the cylindrical FT of a volume f, denoted by F^C, given by,

$$F^C(f) = F^P(F_z(f)) \tag{8}$$

where F_z is the centered 1D FT in the z direction and F^P is the 2D polar FT which operates on each plane that is perpendicular to the z axis. To compute the DFT for 3D cylindrical coordinates,we compute 1-D centered FFT in the Z direction, then we compute 2-D Polar FFT for each plane perpendicular to the Z-axis. In our case since its a unified volume, $N+1$ is the height of cylinder and let M be the even number of desired equidistant projections, then taking a 2D discrete polar Fourier transform (DPFT) for each plane perpendicular to complexity given by,

$$\frac{M}{4} \times (N+1)^2 \log_2(N+1) \tag{9}$$

See [12] for more details about fast and exact computation of 2D DPFT. Thus, we have the total computational complexity for full 3D discrete Cylindrical Fourier transform as,

$$(N+1) \times \frac{M}{4} \times (N+1)^2 \log_2(N+1) \tag{10}$$

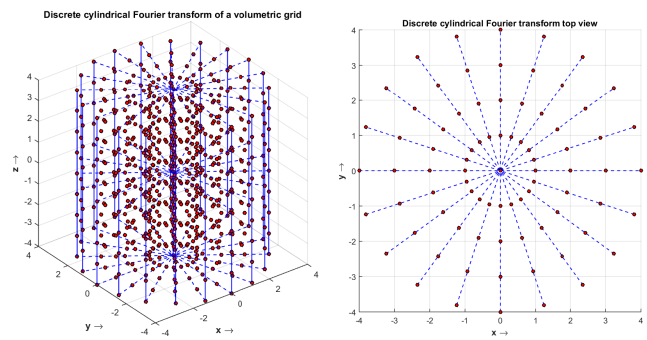

Fig. 2. Multiple projections using cylindrical Fourier transform of a full volumetric grid.

When multiple planar projections are desired and all the projections are angularly equispaced about an axis such as Z, this method gives a one shot highly optimized solution for the problem with complexity $\frac{M}{4} \times (N+1)^3 \log_2(N+1)$. This is just one time computation which can be preprocessed for the entire volume to compute 3D discrete Cylindrical Fourier transform, and all the subsequent computations of the projections is now greatly simplified, since any projection can then be retrieved using an inverse 2D FFT of complexity, $(N+1)^2 \log_2(N+1)$.

III. Experiments

We use 3 real world, publicly available, standard datasets of *visible male*, *skull* and *Foot*, all the datasets are organized in regular 3D Cartesian grids [15],[16]. The solutions rendered are fast and exact.

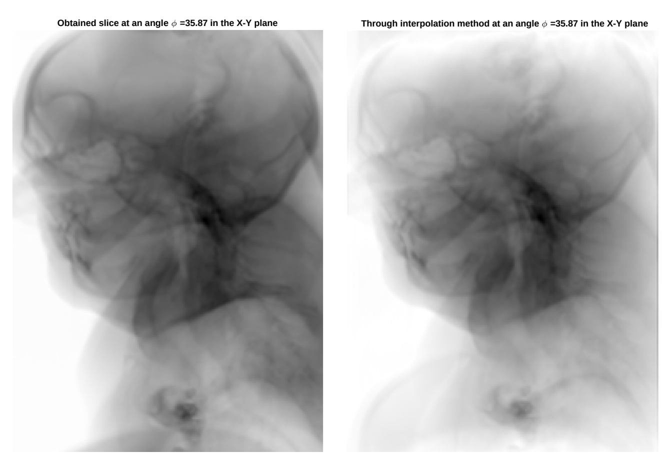

Fig. 3. Single slice at a specified angle, used the complement of the image to enhance the effects of distortion. See the interpolated slice has poor reconstruction at the slice.

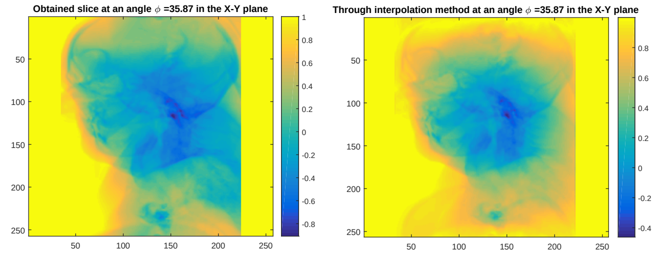

Fig. 4. Ghosting artifacts and problem along sharp edges, while the exact reconstruction directly from the volume data using the proposed method is dense, clean and sharp.

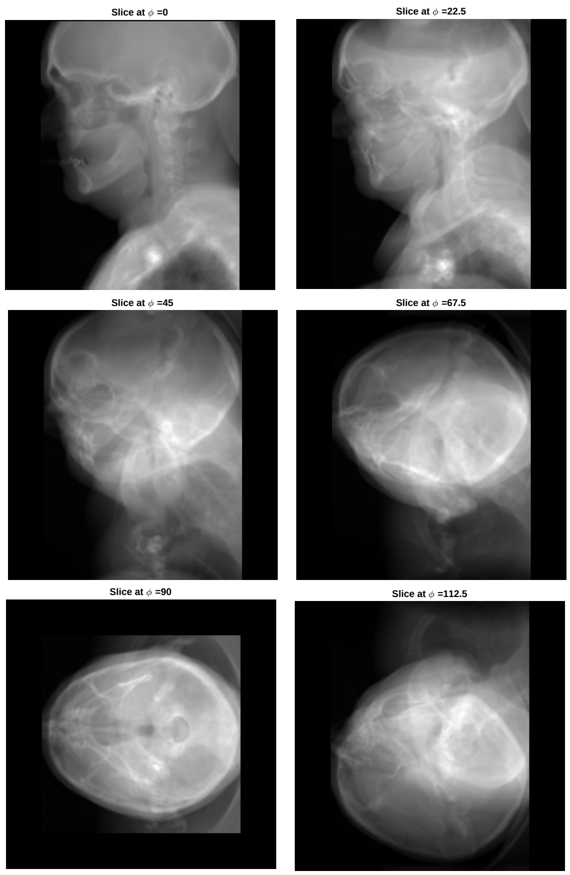

Fig. 5. Full volumetric data analysis using the proposed FVR method on *visible male* dataset, starting from top left at increasing angles of slices.

IV. Conclusions

The main contributions of the paper are:

1. We have designed a highly specialized method to find the exact and fast 2D projection at an arbitrary angle, given the image in 3D spatial domain cartesian coordinates respectively.

2. The algorithm I for single planar projection proposed is algebraically exact and fast without any oversampling, and it doesn't require interpolation in spatial or frequency domain or any approximation stage.

3. The solution proposed is highly flexible since any projection at an arbitrary angle specified by the user can be retrieved with full precision without any artifacts.

4. The algorithm II for multiple planar projection is also algebraically exact and fast. An advantage of using this technique is that only preprocessing is computationally expensive, but the retrieval of angularly equispaced planar projections is very fast with complexity of order $(N+1)^2 \log_2(N+1)$.

5. Due to the preselected nature of planar angles depending on the total number of projection desired M, algorithm II is not very flexible however it has reduced complexity. On the other hand algorithm I for single planar projection is very flexible but has relatively higher complexity. So algorithm I can be used for fine level resolution and algorithm II can be used for coarse level resolution, and a combination of two can be very useful for medical analysis.

6. The fully parallel algorithm can take full advantage of commodity GPGPU processing hardware. Most importantly each of the components used in our solution has a parallelisable structure and scales naturally with processing and memory resources.

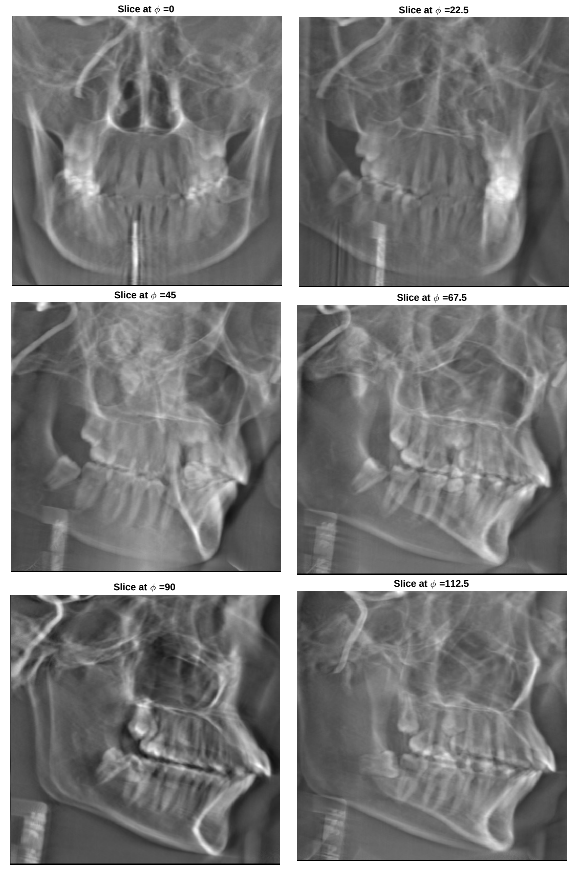

Fig. 6. Full volumetric data analysis using the proposed FVR method on *skull* dataset, starting from top left at increasing angles of slices.

Acknowledgment

The authors would like to thank...

References

[1] Elvins, T. Todd. "A survey of algorithms for volume visualization." ACM Siggraph Computer Graphics 26, no. 3 (1992): 194-201.

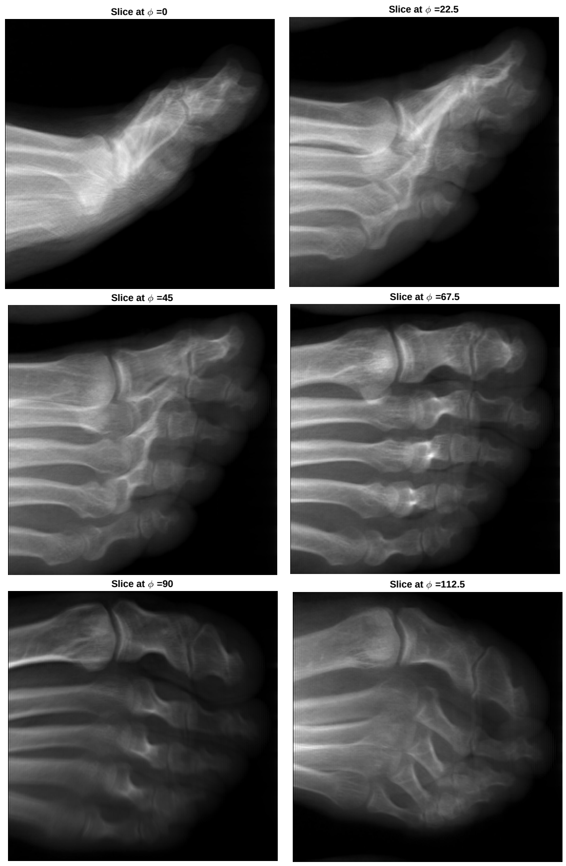

Fig. 7. Full volumetric data analysis using the proposed FVR method on *Foot* dataset, starting from top left at increasing angles of slices.

[2] Ikits, Milan, Joe Kniss, Aaron Lefohn, and Charles Hansen. ”Volume rendering techniques.” GPU Gems 1 (2004).

[3] Preim, Bernhard, and Dirk Bartz. Visualization in medicine: theory, algorithms, and applications. Morgan Kaufmann, 2007.

[4] Levoy, Marc. Volume rendering using the fourier projection-slice theorem. Computer Systems Laboratory, Stanford University, 1992.

[5] Malzbender, Tom, “Fourier volume rendering.” ACM Transactions on Graphics (TOG) 12, no. 3 (1993): 233-250.

[6] Totsuka, Takashi, and Marc Levoy, “Frequency domain volume rendering.” In Proceedings of the 20th annual conference on Computer graphics and interactive techniques, pp. 271-278. ACM, 1993.

[7] Dunne, S., S. Napel, and B. Ruth, “Interactive display of volumetric data by fast fourier projection.” Computerized medical imaging and graphics 16, no. 4 (1992): 237-251.

[8] Abdellah, Marwan, Ayman Eldeib, and Amr Sharawi. “Matlab-based Fourier volume rendering framework.” In Biomedical Engineering Conference (CIBEC), 2014 Cairo International, pp. 55-58. IEEE, 2014.

[9] Abdellah, Marwan, Ayman Eldeib, and Amr Sharawi, “High Performance GPU-Based Fourier Volume Rendering.” International journal of biomedical imaging 2015 (2015).

[10] Abdellah, Marwan, Ayman Eldeib, and Amr Shaarawi, “Constructing a functional Fourier volume rendering pipeline on heterogeneous platforms.” In Biomedical Engineering Conference (CIBEC), 2012 Cairo International, pp. 77-80. IEEE, 2012.

[11] Hansen, Charles D., and Chris R. Johnson. Visualization handbook. Academic Press, 2011.

[12] Syed Alam Abbas, Q. Sun and H. Foroosh,“An exact and fast computation of Discrete Fourier Transform for Polar and Spherical grid”, IEEE Trans Signal Processing, Submitted.

[13] Bailey, David H., and Paul N. Swarztrauber,“The fractional Fourier transform and applications,” SIAM review 33, no. 3 (1991): 389-404.

[14] Averbuch, Amir, Ronald R. Coifman, David L. Donoho, Michael Elad, and Moshe Israeli,“Fast and accurate polar Fourier transform,” *Applied and computational harmonic analysis*, 21, no. 2 (2006): 145-167.

[15] “Online library for medical MR and CT datasets, January 2013, http://www.volvis.org/.

[16] The Volume Library, “Online library for Volume visualization datasets, http://lgdv.cs.fau.de/External/vollib/