A SAR ALGORITHM DEVELOPMENT ENVIRONMENT USING MATLAB®

Dilia Beatriz Rueda S. *Domingo Rodríguez*
dilia@ece.upr.clu.edu *domingo@ece.upr.clu.edu*

University of Puerto Rico - Mayagüez Campus
Department of Electrical and Computer Engineering
Computational **S**ignal **P**rocessing **G**roup (cspg@exodo.upr.clu.edu)
Mayagüez, Puerto Rico, 00681-9042

ABSTRACT

A MATLAB® based environment is presented for signal processing algorithm development targeted to computational intensive image kernel formation in the area of Synthetic Aperture Radar (SAR).

Special attention is given to the development of algorithms for image formation from raw data. Kronecker algebra has been used, as a tool aid for reducing the computational effort in the MATLAB® implementation process of unitary operators, such as the multidimensional discrete Fourier transform, which form an integral part of some of these algorithms. The MATLAB® environment, named SARCSPE, is described in a finite dimensional multilinear algebra framework.

1. INTRODUCTION

The essence of remote sensing resides in the acquisition of information about remote targets for further processing [1]. Synthetic Aperture Radar (SAR) has evolved as a powerful tool that accomplishes the necessities of remote sensing plus some additional characteristics such as day-night all-weather operation and good resolution. These characteristics make SAR a very attractive tool in remote sensing but a very expensive operation from the point of view of computational processing and storage costs.

Based on advances reached in signal processing and image processing, for example, fast Fourier transform (FFT), correlation and convolution techniques, an environment for SAR processing has been developed and constitutes the work reported in this article. The environment includes various implementations of SAR algorithms. Many of them have been improved with the use of Kronecker products and have been implemented as matrix based operations [4]. For this reason, MATLAB® has been used as a development tool.

This paper is organized as follows: First, SAR imaging is explained; second, matrix and imaging array algebra concepts are given; third, image processing aspects are treated; fourth, some important operations implemented are presented; fifth, the conclusions are made and, finally, future work is proposed.

2. SAR IMAGING SYSTEM

SAR is a system that illuminates an area in order to extract information about it. Its nature of active system allows SAR to work all the time, during the day and during the night. Its operation in the microwave frequency range allows SAR to work under any climatic condition and to obtain images of higher resolution than those generated with other instrument [6]. The next figure shows an example of SAR operation and processing:

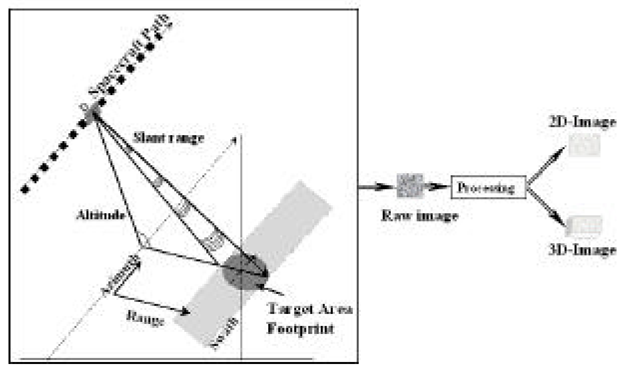

Figure 1. Example of SAR Processing.

After the sensor illuminates the target area, an image is generated; this image is called raw image because it is not processed completely. SAR imagery goal consists of recovering the original image or reflectivity function from raw data as shown in Figure 1. Two steps are necessary to recover the image:

In the first step, the raw data is generated using time-frequency processing techniques such as the ambiguity function, as illustrated in Figure 2. The process consists of computing the correlation between the transmitted and the reflected signal, which corresponds to the spectral density reflected by the target [1].

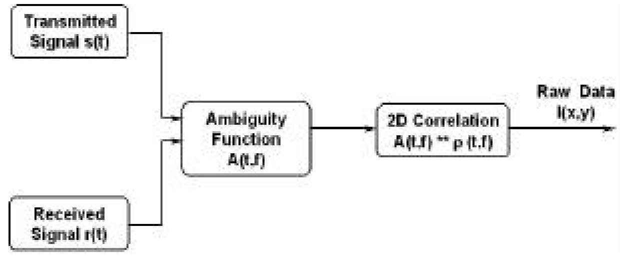

Figure 2. SAR Raw Data Generation

The cross-ambiguity function of the transmitted and received signal is given by:

$$A_{(x,y)}[m,k] = \sum_{n=0}^{N-1} x[n]y^{*}[n+m]e^{\frac{-j2pkn}{N}}$$

Here, $A[m,k]$ is the ambiguity function, m represents time delay, k is the frequency shift, $x[n]$ is the transmitted signal, y is the received signal, and N is the length of the transmitted signal.

In the second step the image is recovered from raw data. is to recover the image from the raw data. The algorithm shown in Figure 3, was proposed by Franceschetti et al [2]. The algorithm performs cyclic correlation operations between raw data and the scene spectral density.

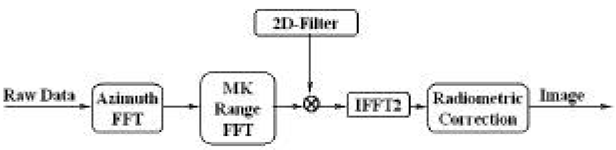

Figure 3. Image Formation Algorithm

Figure 4 illustrates an example of raw data generation.

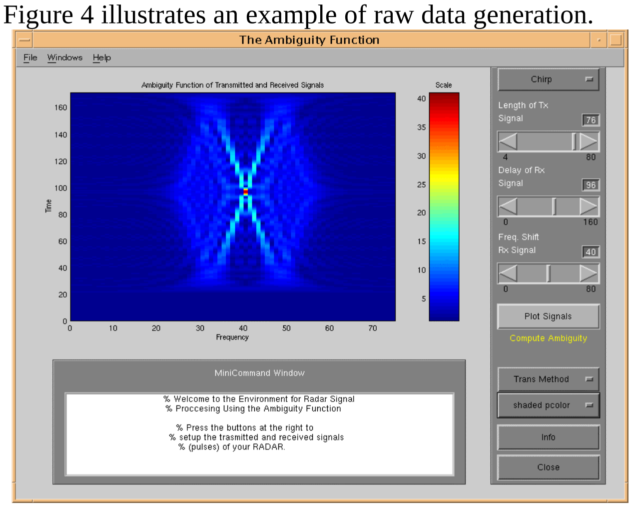

Figure 4. Point Spread Function Simulation-Chirp.

In order to illustrate the computational effort in SAR processing, a typical example is presented in a PC platform setting. Let an image of size 100 $\times$100 pixels occupy 30 MB of hard disk space. The convolution of two images, using FFT, can take as much as 90 seconds in a PC. The resultant image of 199 $\times$199 pixels occupy more or less double the space each of the original images. In almost all cases it becomes necessary to apply not only one convolution but a series of equivalent operations; consequently, the computational effort for processing SAR images is high enough and justifies the search for efficient algorithms.

3. MATRIX AND IMAGING ARRAY ALGEBRA

One of the main objectives of this environment is to assist in the analysis and design of SAR imaging algorithms in MATLAB® for mapping to hardware computational structures. This work presents some results of research efforts conducted in the area of computational signal processing algorithms for synthetic aperture radar (SAR) applications. The work emphasizes the use of computational Kronecker products (CKP) algebra, a branch of finite dimensional multilinear algebra, as a tool aid in analysis, design, and implementation of SAR signal processing algorithms. In particular, the work concentrates on the use of CKP algebra for the mathematical formulation of two-dimensional fast Fourier transform (FFT) algorithms and how they are used in the efficient computation of two-dimensional correlation operations prevalent in SAR signal processing algorithms for kernel image formation.

Working in identifying fast two-dimensional correlation methods for SAR kernel image formation from raw data, using finite discrete ambiguity functions as point spread functions or unit impulse response functions, has induced us to develop new variants of two-dimensional FFT algorithms. The advantage of these new FFT variants over conventional formulations is that additive group theoretic properties of the two-dimensional input/output indexing data sets are used for the mathematical formulation of the algorithms themselves, reducing in this way their computational complexity.

There exist many formulations of fast algorithms for computing the two-dimensional discrete Fourier transform (DFT). In this work CKP algebra is used as a mathematical language to identify similarities and differences among the various FFT algorithm variants as well as for the creation of new variants. Each two-dimensional DFT computation is expressed in matrix form for analysis. The two-dimensional DFT matrix, in turn, is decomposed into a set of factors, called functional primitives, which are individually identified computational software procedures and eventually mapped to an underlying computational structure. It is in this mapping process where the language of CKP algebra becomes instrumental. For a given computational structure and multidimensional DFT matrix, there are many FFT algorithm variants which can map to this computational structure. The mathematical language of CKP algebra aids in this mapping effort by identifying the more computational efficient FFT variants and thus reducing the computational effort.

4. SARCSPE OPERATION

An environment is a system composed of a set of tools available to reach a goal, in this case, the efficient processing of multidimensional signals. An environment is usually characterized by its components and their relationship: *Operators* are combined according to *composition rules* for applying *action rules* over an *input set (operands set)* in order to obtain an *output set.* A *user* can make use of all those elements, normally through an *interface,* which, is a **G**raphical **U**ser **I**nterface *(GUI)* most of the times.

Algorithms are used as tools in the implementation of this environment. They accomplish actions over input for producing an output. Performance of a computational environment depends greatly on the implementation of algorithms. In this work, an attempt has been made to produce an algorithm development environment for SAR processing. SARCSPE (Synthetic Aperture Radar Computational Signal Processing Environment) groups in one environment, matrix and imaging array algebra, operators, operations, and tools necessary for SAR processing, it was programmed in MATLAB® [5].

Components are organized according to their roll in the environment. Different alternatives have been presented to the user; for example, different computation methods for executing the same task have been programmed, trying always to minimize the number of computations. Important functions included were ambiguity function, fast Fourier transform, cyclic correlation operation, and cyclic convolution operation.

Kronecker products algebra was used to improve algorithms, this tool allows to eliminate redundant operations, and, since it is based in matrix vector product operations [3], it allows to take advantage of MATLAB® as programming tool.

Figures 7 and 8 illustrate an example of how the environment has been conceived. An example has been presented in which, the cyclic correlation between two images B and A has been computed. The steps for computing the result were: First, apply cyclic reflection operation to image B; second compute cyclic convolution between the result and image A. The same result could be obtained by applying any of the available methods provided in the GUI menu for computing cyclic correlations.

As it can be appreciated, the environment is organized following the concepts explained above. On the left side of the interface are the computational utilities. There are different computational operations, for example, filtering operations, raw data generation, and basic operations. On the right are visual tools. The operations are divided in unary and binary operations, and in the middle appear operands and the results as figures. Other examples of the environment are presented in Figure 5 and Figure 6 below.

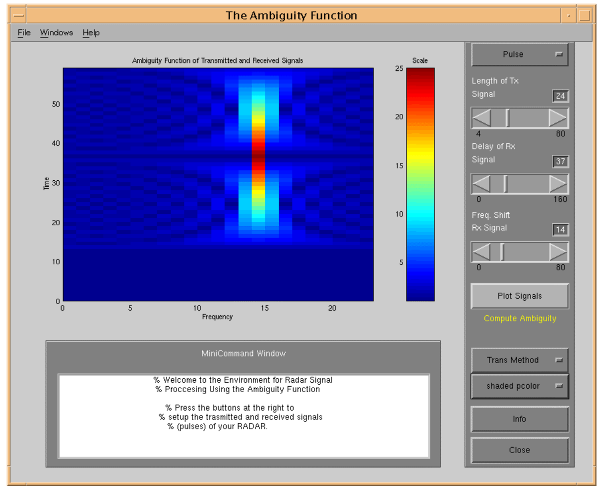

Figure 5. Point Spread Function Simulation – Pulse

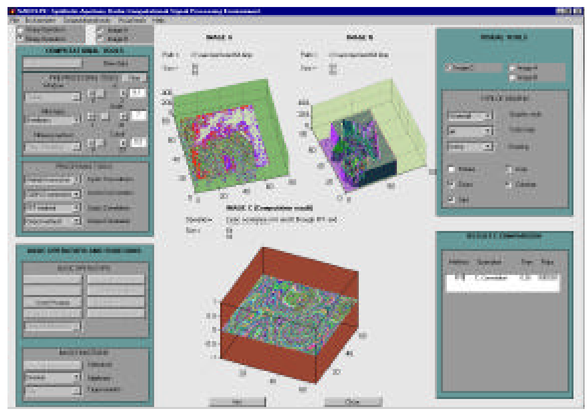

Figure 6. Example of Cyclic Correlation Computation

5. CONCLUSIONS

An environment has been developed for fast SAR processing. The design of the environment was done considering the main components of the concept of environment and image algebra. Some of the algorithms have been implemented using different approaches in order to compare the performance of the different methods. The most important functions exposed were the ambiguity function, used to simulate raw data generation, and the cyclic correlation, which was also implemented in terms of cyclic convolutions in order to improve algorithms performance. Kronecker products were presented as the optimization tool.

6. FUTURE DIRECTION

It is necessary to work in the mapping of different algorithms with Kronecker products to different architectures and to consider real time data acquisition implementations. Other step in this work is to extend all the algorithms implementations to the multidimensional case.

7. REFERENCES

[1] R. E. Blahut, W. Miller Jr., C.H. Wilcox, "Radar and Sonar", *The IMA volumes in Mahematica and its Applications*, part 1, vol. 32

[2] G. Franceschetti, R. Lanari, E.S. Marzouc, "Efficient and High Precision Space-Variant Processing of SAR Data", *IEEE Transactions on Aerospace and Electronics Systems*, Vol. 31, No. 1, January, pp. 227-237, 1995.

[3] J. R. Johnson, R. W. Johnson, D. Rodríguez, R. Tolimieri, "A Methodology for Designing, Modifying, and Implementing Fourier Transform Algorithms on Various Architectures", *IEEE Transactions on Circuits, Systems, and Signal Processing*, Vol. 9, No. 4, 1990.

[4] D. Rodríguez, "Computational Kronecker Products for Multidimensional SAR Signal Processing Algorithms", *IGARSS* , 1998.

[5] The MathWorks Inc. v. 5.2, "MATLAB® the Language of Technical Computing", *http://www.mathworks.com*. 1998.

[6] A.W. Rihaczec, "Principles of High Resolution Radar", *Artech House Boston*, London, 1996.

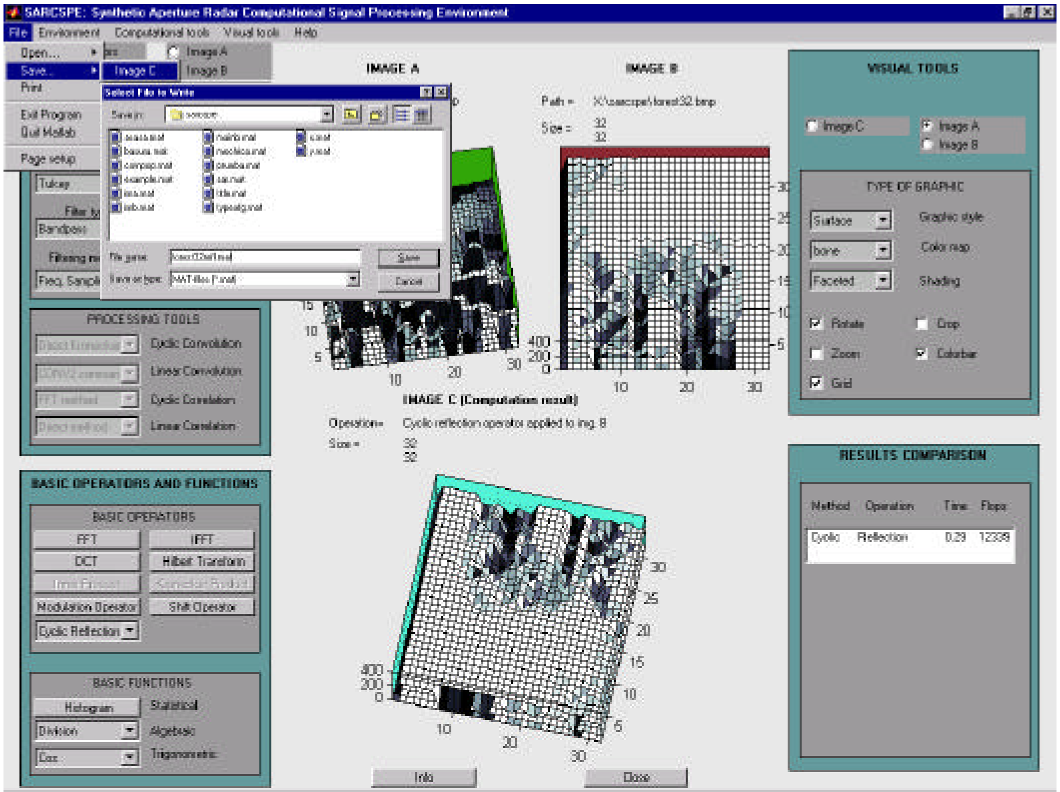

Figure 7. Results of Cyclic Reflection Operation with SARCSPE.

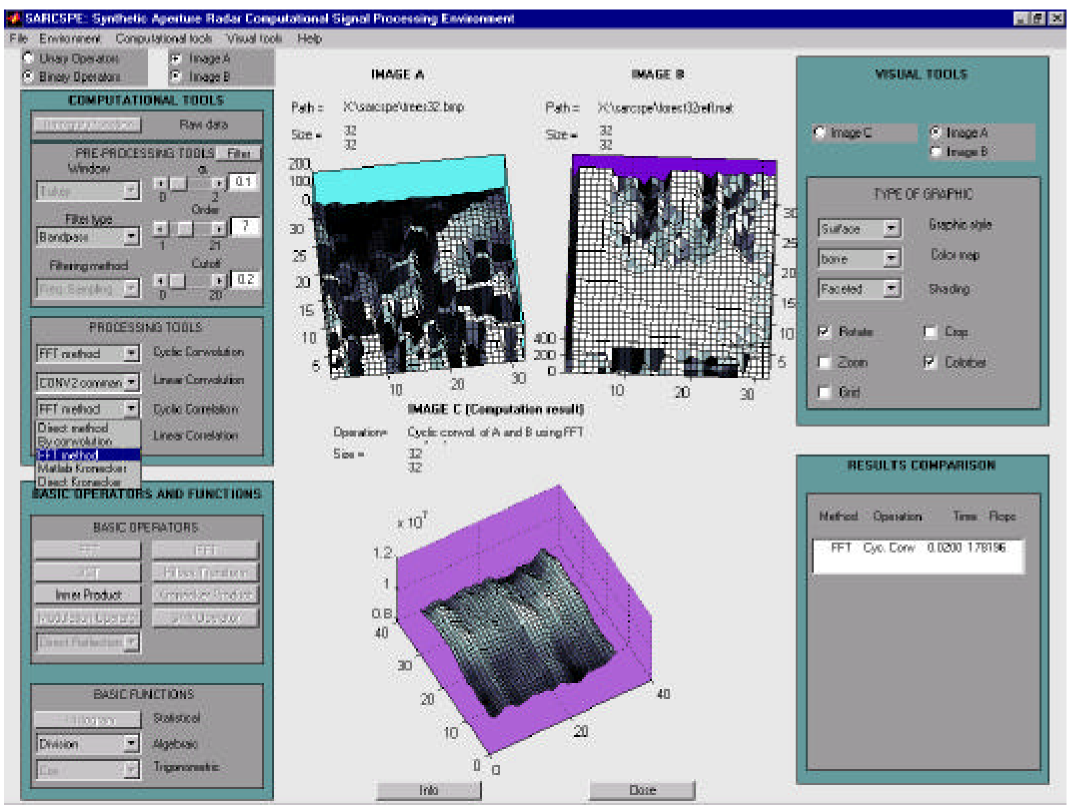

Figure 8. Cyclic Correlation between Image A and the Result of Reflection. Some Visualization Changes Were Applied Also.