

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/290465603>

A Graphical Open Source Tool for Preprocessing Satellites Images

Article · January 2014

DOI: 10.7763/IJIT.2014.V4.445

CITATIONS

0

READS

268

5 authors, including:



Reginaldo Santos

Federal University of Pará

29 PUBLICATIONS 45 CITATIONS

[SEE PROFILE](#)



Ana Siravenha

Federação das Indústrias do Estado do Pará

24 PUBLICATIONS 36 CITATIONS

[SEE PROFILE](#)



Damares Crystina Oliveira de Resende

VALE

10 PUBLICATIONS 4 CITATIONS

[SEE PROFILE](#)



Evaldo Pelaes

Federal University of Pará

33 PUBLICATIONS 90 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Modeling and development devices and sensor networks based on optical fiber [View project](#)



Enhancements for artificial intelligence algorithms [View project](#)

A Graphical Open Source Tool for Preprocessing Satellites Images

Reginaldo Filho, Ana Siravenha, Damares Resende, Danilo Souza, and Evaldo Pelaez

Abstract—The digital image processing is widely studied in the scientific literature. Techniques for image treatment derived from satellites are continuously developed and improved. Although most of these techniques are readily available to the researchers, it is known that a minimum knowledge in programming should be purchased by those who wish to use these techniques. In this context, this paper proposes an open source graphical application which refers to preprocessing of images coming from satellites. The main focus is to become simple the use of preprocessing techniques without requiring the user to have knowledge about programming. To achieve this goal, technologies such as Qt framework (used for the development of graphical interface) and TerraLib (used for the development of preprocessing techniques) were used together and Façade design pattern was chosen to perform the communication between the technologies. This design pattern consists of developing a single interface to access resources developed and available in TerraLib library. The communication between the technologies is explained in detail and this work concludes with expectations about the proposal graphical application.

Index Terms—Façade design pattern, Qt framework, TerraLib library.

I. INTRODUCTION

Over the years the Digital Image Processing (DIP) area is growing and gaining more space. Due to the several available applications, this area have been developed many softwares that helps digital images manipulation. In remote sensing for example, the developed algorithms aims to correct imperfections generated during capture as well as inconveniences caused by characteristics of the imaged environment.

Thus, this paper presents the development of software directly related to DIP and also focused in remote sensing area.

This work aims to use open source technologies for academic purposes. Two such technologies are highlighted: Qt framework and TerraLib library.

The Qt framework is cross-platform application and user interface framework to develop the Graphical User Interface (GUI). Written in C++, Qt also provides support for others programming languages; it has extensive documentation and bibliography. It is maintained by Qt project involving groups from different countries and individual developers

Manuscript received August 24, 2013; revised December 17, 2013. This work was supported by the Amazon Research Foundation/Vale S/A [grant number 021/2010]; and the National Council of Technological and Scientific Development [grant number 142404/2011-0].

The authors are with the Technological Institute, Federal University of Para, Brazil (e-mail: reginaldo.filho@icen.ufpa.br, siravenha@ufpa.br, damares.resende@itec.ufpa.br, danilofraza@ufpa.br, pelaes@ufpa.br).

(www.qt-project.org).

The TerraLib is developed in the programming language C++ and offers multiple tools of Geographic Information System (GIS) [1]. Among various features of the library, it emphasizes the set of methods available for working with image manipulation from artificial satellites; this is an important feature for this work.

These technologies were chosen because they are open sources and work with the programming language C++. It also justifies the fact that Qt has an extensive range of classes that assist in the development of user interface and being multiplatform, i.e., the graphical application proposed in this paper is democratized between operating systems, avoiding rework.

Also written in C++, the proposed application is based on [2], [3]. It works in the preprocessing step of images, using sophisticated techniques involving mathematical morphology and processes of reconstructing regions in images.

To achieve this goal, a design pattern called Façade was chosen to facilitate the communication between the graphical interface developed in Qt and image preprocessing techniques developed in TerraLib. Based on the Façade design pattern described in [4], this work provides a unique interface to a set of resources developed and available in TerraLib.

The main focus of this work is hiding the complexity of the image preprocessing algorithms through a GUI application which facilitates the use of the most technical resources by the user. Some emphasis to the Façade design pattern will be given to demonstrate how it works in this proposed GUI application.

This paper is presented as follow: in Section II is explained the theoretical information about the techniques available in the graphical application; in Section III is presented the methodology applied in this work, highlighting aspects of the used technologies; whereas in Section IV is shown the results and discussions about the developed application and how it was implemented the Façade design pattern; finally, Section V brings together all the focus of this paper and exposes future expectations for this work.

II. THEORETICAL INFORMATION

This section briefly explains the concepts and theoretical approaches regarding the techniques of image processing included in the proposed application.

III. ATMOSPHERIC COMPONENT DETECTION

This work considers the following atmospherics

components: clouds and shadows. The detection technique enables to identify and delimit them. The detection is fundamentally necessary because enables to direct the next preprocessing steps about detected objects in images. The detection technique basis was originally proposed in [5] and later refined in [2], [3], which were added the possibility to detect shadows and implemented clouds and shadows factors, allowing flexibility in detection technique.

IV. MORPHOLOGICAL OPERATIONS

Two primitive operations of morphological image processing are present in this paper: erosion and dilation. The erosion shrinks or compresses image objects, while dilation expands or thickens them. From the literature, it is highlighted two main combinations of morphological operations: opening and closing [6].

The opening consists of morphological erosion followed by dilation. This procedure removes the regions of an object that can not contain the structuring element (morphological operation mask which scans each pixel of the original image by extracting relevant information) and softens contours in the image. The morphological closing is the reverse of the opening process, i.e., it is applied a dilation followed by an erosion. This also tends to soften the contours of objects in the image, but unlike opening, closing generally fills narrow holes smaller than the structuring element chosen [6].

The combination of erosion and dilation operations, along with the shape and dimension of the structuring element, causes the changes in the result image. In this paper, the shapes and dimensions of the structuring elements are those implemented in [7] which aim to flexible even more combination possibilities of morphological operations.

V. REDEFINED REGION METHODS

Initially two methods of redefining region are discussed. The first is called *Inpainting* and it is used to reconstruct the regions that are missing or have damage that it has been detected by the atmospheric component detection technique.

The technique used in this paper is based on [8], [9], i.e., it works with information propagation and also utilizes a smoothing interpolation, based on the surrounding regions where the pixels need to be redefined, and finally estimating the contaminated region.

The second method is called *Texture Synthesis* used in the context of filling regions by bidimensional textural patterns. The Texture Synthesis used in this work is based on models, i.e., it is aimed to redefine large image objects by images consistent fragments with their surrounding regions.

VI. IMAGE DECOMPOSITION

The decomposition of images is used to separate the original image in two other images: cartoon (structure) and texture. The decomposition method used in this work is based on that described in [10] which it is proposed a solution using a fast and approximate variational problem, applying a pair of non-linear low and high-pass filters. For each pixel of the image the algorithm defines whether it belongs to the cartoon

or texture.

The decomposed image is transformed into a binary image with values 1 for heterogeneous areas of texture and 0 for homogeneous areas of cartoon. Then, a non-linear median filter can be applied to smooth areas where small gaps of a given class are surrounded by regions where dominates another class.

VII. METHODOLOGY

This section explains the process of developing the proposed application. The combination between Qt and TerraLib, elicited requirements and Façade design pattern are detailed below.

VIII. QT + TERRALIB

As previously mentioned, this application makes use of the Qt framework and TerraLib library. The purpose of this study comes up from the need to have an application with GUI to assist in the analysis and evaluation of DIP techniques, which are focused on detection, smoothing and removal of atmospheric components in images from artificial satellites.

In order to communicate the two technologies, it was interesting that the architectural design used design patterns to guarantee development in modules, i.e., separating the hard core code of preprocessing techniques and the graphical interface. To achieve this intention, the Façade design pattern [4] was used. This design pattern divided the project into three modules: Qt, Façade and TerraLibCore. The Fig. 1 shows a visual representation of the three modules over a simple perspective.

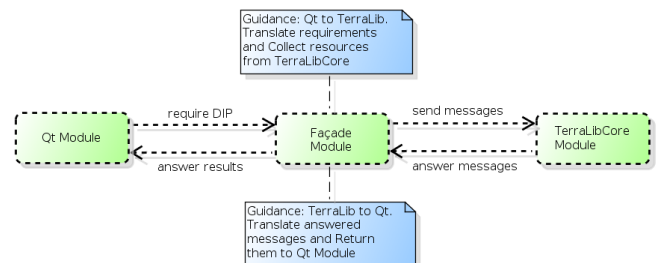


Fig. 1. Modules visual representation.

The Qt module is responsible for all graphical components (windows, buttons, dialogs, menus, etc.) and interactions that the user may have with the application. This module consists of several classes that implement all the features of the graphical application.

The Façade module is the design pattern chosen that aims to facilitate access and use of the TerraLibCore module, this one has several classes responsible for applying DIP. The Façade simplifies the use of the TerraLibCore module resources, including independence between Qt and TerraLibCore modules, i.e., there is no Qt code in TerraLibCore module and vice versa.

The TerraLibCore module is the one which contains all the codes that make the image processing. Everything that is related to changes in images that are loaded into the application, this module is responsible for performing corresponding operation.

IX. MAIN REQUIREMENTS ELICITED

More than 35 requirements (it is considered functional and non-functional requirements) were elicited for the development of this application, which makes it impossible to explain all them in this paper. There are simplest requirements, such as load an image in the application, to most complex requirements like being able to execute texture synthesis method in an image loaded in the application.

The method used for elicitate the requirements is called Introspection [11]. This technique requires the development of requirements based on what the analyst believes that are the demands and necessities of the system users and stakeholders. It represents an initial point for other requirements elicitation approaches and it is really effective when the reporter is at the same time familiar with the system goals and domain, and is familiar or even expert in the business processes performed by the users.

Next, it will be highlighted some requirements to characterize the main focus of this paper, and also a few others are highlighted by bold within topics (that does not feature the importance among them).

- *Apply DIP techniques*: a key requirement of this application is capable of performing DIP techniques on **images loaded in the application**. The main techniques that the application provides, initially, are those addressed in the theoretical. It follows: detection of atmospheric components (clouds and shadows); morphological operations (opening and closing); regions redefining methods (*Inpainting* and *Texture Synthesis*); image decomposition (cartoon and texture); applying filters (e.g. median).
- The application can easily integrate other techniques of DIP to their set of functions in the future. In addition, the application also offers image preview **zoom in for visual evaluation**, and even **statistical data extracted from the image**, and may even **save them in a portable document format** (PDF) for further analysis. The **image binarization** is customized according to the threshold chosen by the user and it can conclude the processing with image transformation in **gray scale method**.
- *Structuring elements choice*: this requirement is derived from an expansion of TerraLib held in [7]. It allows the user to choose the shape and dimension of the structuring element used in the process of erosion and/or dilation, allowing the **opening or closing operation**. The choice of the shape and dimension of the structuring element makes the resulting image suffers significant changes, thus in order to have a satisfactory opening or closing, attention should be given to this choice.
- *Preprocessing steps*: this means that a resulting image of a preprocessing technique available in the application can be used as input to another one. This allows the user to make different combinations contributing to the analysis of results. The main application window **shows the result of each preprocessing step**.

The Fig. 2 shows the use case diagram that the application user can perform. The use case diagram has been divided into three packages of image features: image configuration, image processing and image analysis.

Image configuration relates to simple configurations of an image. “Load image” is a fundamental use case to work with an image in the application. “View image” consists of observing the image already loaded in the application. This use case also involves the possibility of zooming in the loaded image, thus enabling display the details of the same. “Save image” allows the user to save the image being displayed in the application. This can be used to save the result of the processed images.

Image analysis corresponds to use cases related to statistical data that can be extracted from the image. “Perform statistical process” means extract statistical data for image analysis purpose. “Save statistical report” allows saving statistical data extracted from the image in a report format. This report contains the image that was worked on and their key information and statistical data of each band image.

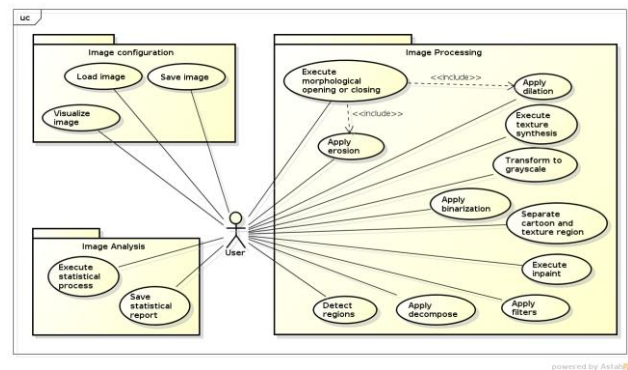


Fig. 2. Use case diagram.

Image processing includes use cases effectively related operations that make changes in the input image. In this package, there is the use case “Perform morphological opening or closing”, if the user runs this method it is necessary to apply other two use cases “Apply erosion” and “Apply dilation”. Depending upon the order that the latter are applied, it has the effect of opening or closing.

X. FAÇADE

Due to the amount of written codes for image preprocessing (TerraLibCore module) and because of the main focus of this paper, a GUI preprocessing tool (Qt Module), the Façade design pattern was the most suitable for the development of the application.

- Some advantages of using the Façade design pattern:
- Simplifies the use of TerraLibCore module, it allows to apply techniques of preprocessing just passing some parameters in the corresponding function;
- Facilitates the future maintenances of the application as if it is necessary to carry out corrective maintenance, separation into modules of the application helps to quickly find the location of the problem;
- Improves the code readability, considering that there is no TerraLib code in Qt module nor Qt code in TerraLibCore module, clearly separating what belongs to graphical interface from what belongs to preprocessing techniques;
- Covers modules, because the Façade is responsible for communication between Qt and TerraLibCore module. Façade only knows the extremities. If it is necessary to

make significant changes in Qt module, these changes will be seen only by the Façade, but not by TerraLibCore module, the opposite is also true;

- Because the project is open source and developed by a group, simplifying the application development allows others to participate in the project so that it does not have many setbacks to understand and improve it;
- The modularity of the system, achieved by Façade, allows the user to direct the development of application parts, i.e., the divide and conquer paradigm, indirectly, is used in this sense. It can direct the development of the graphical interface or the development of more preprocessing techniques or even in both parts simultaneously (by different teams) without compromising the application.

The Fig. 3 shows the representation of the Façade class in this work. This representation resembling the syntax used to make UML class diagram, however, it was simplified and modified for explanation purpose.

It is noticed that the class constructor is private, represented by the sign (-), this implies that any object of the class can be instantiated. Therefore, to use any existing public method in this class it is necessary being static, represented by outlined in the signature method. To simplify the representation, no argument was made clear, it is used a (...) symbol instead. Except for the two methods of image conversion Qt for TerraLib and vice versa, respectively *convertQImage2Raster()* and *convertRaster2QImage()* returning void, all existing methods in the class return a pointer of type QImage (Qt class used to work with images). This happens because who sends messages to the “FacilitiesFacade” is the Qt module, so the response operations in the class “FacilitiesFacade” must attend the way the Qt module works, i.e., through the class QImage.

The objective of each method was discussed on the sections above, and the names of the methods are highly suggestive. The additional methods is found in *applyAbs()*, *applyGrayScale()* and *applyBinary()*. They are used in intermediate steps, the first generates absolute values for each pixel, the second transforms into a gray scale image and the third executes binarization.

XI. GPL V3 LICENSE

GNU General Public License (GPL), in its most current version 3 released on June 29, 2007, is the name given to free software license. This license was chosen to be used by the software developed; it ensures the philosophy of change and public use of the application. This work is in full agreement with the freedoms present in the license.

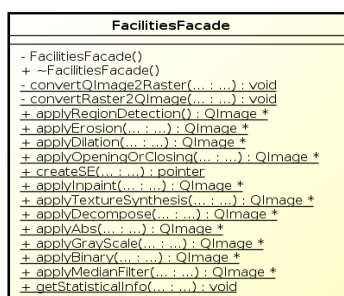


Fig. 3. Simplified façade class.

XII. RESULTS AND DISCUSSIONS

The Fig. 4 shows the main application window. The figure is divided into 4 parts. In one, there is the main menu of the application where all the application features are shown. In two, it has a toolbar that provides quick access to some application features. In three, there is a stage, where the image is viewed by the user. And in four, there is a dock that is responsible for displaying the statistical data of the image. For now the application works with the composition of satellite images of up to three bands.

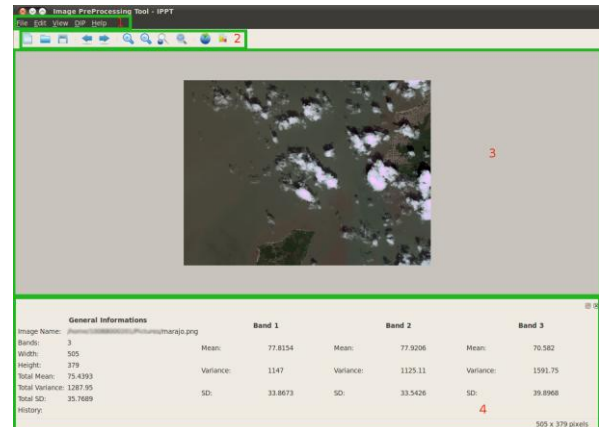


Fig. 4. Main window.

As a matter of exemplification, it is shown a process of decomposition (the choice of this use case was arbitrary) followed by a binarization of a satellite image in order to explain how is the communication between the Qt and TerraLibCore module through Façade module. The Fig. 5 shows the sequence diagram about the details of use case “Apply decompose”.

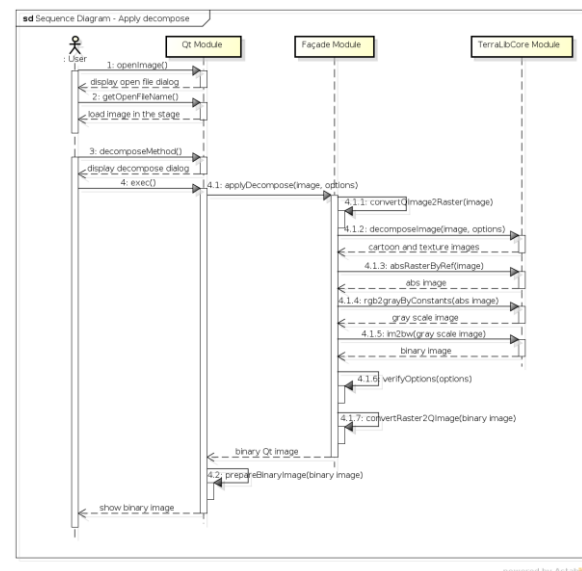


Fig. 5. Sequence diagram-apply decompose.

Initially at 1, the user requests to load an image in the application. Qt module provides a dialog for the user to navigate through directories and search the image. When it is chosen, Qt module loads the image through the method 2. With the image loaded, the user has the possibility to use the resources of the application, in this case it was chosen the decomposition method 3. Qt module is responsible for

showing a dialog in order to set the decomposition parameters. The user sets these parameters and finally runs 4. Qt module passes the method chosen, along with the parameters entered by the user, for the Façade module. This module converts the image in Qt format to TerraLib format in 4.1.1 and then passes to TerraLibCore module all options chosen by the user, in this example applies the decomposition 4.1.2, the absolute value in 4.1.3, transformed into gray scale in 4.1.4 and executes binarization in 4.1.5. Façade module is also responsible for checking the options chosen by the user in 4.1.6 and converts the resulting TerraLib image format in Qt image format in 4.1.7.

It is important to realize that without the Façade module, it would be require that Qt module could communicate directly with TerraLibCore module to use resources. It is emphasized that the complexity of implementing a solution without the Façade design pattern would significantly increase, particularly decreasing code readability and harming future maintenance. Also, without Façade design pattern, TerraLib codes would mix with Qt codes and/or vice versa, which would strengthen the interdependence between technologies. And this is a feature which is not desired in this work.

XIII. CONCLUSION

This work aimed to expose the internal aspects of the implementation of a Qt module over TerraLib implementation. As means of process optimization (among other reasons previously mentioned), it was used Façade design pattern. This pattern is responsible for the communication between Qt and TerraLibCore independent modules.

It is expected that the proposed application contributes, at least in the academic environment to facilitate the use of techniques for preprocessing of images, including users who are unfamiliar with the area to use programming techniques without prior programming knowledge. In addition to open source, the application allows it to be used on different operating systems; this contributes to the decrease the barrier between the application and the users. It is also expected that new techniques of image processing application can be added and it is not limited to only preprocessing step.

REFERENCES

- [1] INPE. (2002). Funcate, and Tecgraf. *Biblioteca geográfica open source desenvolvida em C++*. [Online]. Available: <http://www.terralib.org>.
- [2] A. Siravenha, "Um método para classificação de imagens de satélite usando Transformada Cosseno Discreta com detecção e remoção de nuvens e sombras," *Dissertação de mestrado*, Universidade Federal do Pará 2011.
- [3] D. Sousa, A. C. Siravenha, and E. Pelaes, "A hybrid approach to clouds and shadows removal in satellites images," *Computational Modeling of Objects Represented in Images III*, CRC Press, pp. 153-158, August 28, 2012.
- [4] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns. Elements of Reusable Object-Oriented Software*, Addison-Wesley Professional Computing Series, 1995.
- [5] C. Y. Hau, C. H. Liu, T. Y. Chou, and L. S. Yang, "The efficacy of semi-automatic method," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2008.
- [6] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Addison-Wesley Publishing Company, 2008.
- [7] D. Resende, R. Filho, A. Siravenha, D. Souza, and E. Pelaes, "A influência do elemento estruturante no processo de detecção de nuvens em imagens de sensoriamento remoto," *Simpósio Brasileiro de Sensoriamento Remoto (SBSR)*, 2013.
- [8] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpaint," in *Proc. the 27th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co, 2000, pp. 417-424.
- [9] D. Garcia, *Robust smoothing of gridded area in one and higher dimensions with missing values*, Elsevier, vol. 54, pp. 1167-1178, 2010.
- [10] A. Buades, T. Le, J.-M. Morel, and L. Vese, *Cartoon + texture image decomposition*, 2011.
- [11] M. Jirotko and J. Goguen, *Requirements engineering, social and technical issues*, Academic Press Limited, 2004.



Reginaldo C. S. Filho was born in Para, Brazil on October 18, 1988. He graduated in computer networks at the Faculty of Amazon Technology, Para, Brazil in 2008. He is currently majoring in computer science from the Federal University of Para and is linked to the Institutional Program for Scientific Initiation Scholarship, working as a programmer in the Laboratory of Signal Processing (LaPS) in the area of Digital Image Processing (DIP). His research interest includes DIP, computer networks, desktop, web and mobile developing, defensive programming, algorithm analysis, database and genetic algorithms.



Ana Carolina Siravenha received her B.S. degree in computer engineering from Federal University of Para (2008) and the MSc. degree in electrical engineering from the Federal University of Para (2010). She is currently a Ph.D. student in electrical engineering at the Federal University of Para. She has experience in electrical and computer engineering with emphasis in signal processing, mainly in the following areas: digital image processing, remote sensory images segmentation and classification, image restoration and Discrete Cosine Transform applied to classification.



Damares de Resende was born in Para, Brazil on October 25, 1992. She is currently undergraduate in computer engineering from the Federal University of Para and is linked to the Institutional Program for Scientific Initiation Scholarship, working as a programmer in the Laboratory of Signal Processing (LaPS) in the area of Remote Sensory and Digital Image Processing (DIP). Her research interest includes DIP, algorithm analysis, database and genetic algorithms.



Danilo Sousa received his B.S. degree in computer engineering from Federal University of Para, Belem, Brazil, in 2011. He is currently a MSc. Student in the Post-Graduate Program in Electrical Engineering. His research interests are in digital image processing (image restoration, inpainting, segmentation and classification), remote sensing and Pattern Recognition.



Evaldo Pelaes received his B.S. degree in electrical engineering from Federal University of Para (1977), Full undergraduate degree in physics from the Federal University of Para (1977), MSc. degree in electrical engineering from the Catholic University of Rio de Janeiro (1982) and Ph.D. in Electrical Engineering State University of Campinas (1998). He's currently an associate professor at the Federal University of Para and has experience in Electrical Engineering with emphasis in signal processing, mainly in the following areas: wavelet transform, image coding, propagation models, and mobile power quality.