

# Innovative Smart Healthcare and Bio-Medical Systems

AI, Intelligent Computing, and  
Connected Technologies



Edited by

**Abdel-Badeeh M. Salem**

# Innovative Smart Healthcare and Bio-Medical Systems



# Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

# Innovative Smart Healthcare and Bio-Medical Systems

AI, Intelligent Computing, and  
Connected Technologies

Edited by  
Abdel-Badeeh M. Salem



**CRC Press**

Taylor & Francis Group

Boca Raton London New York

---

CRC Press is an imprint of the  
Taylor & Francis Group, an **informa** business

First edition published 2021  
by CRC Press  
6000 Broken Sound Parkway NW, Suite 300  
Boca Raton, FL 33487-2742

and by CRC Press  
2 Park Square, Milton Park, Abingdon, Oxon OX14 4RN

© 2021 Taylor & Francis Group, LLC  
CRC Press is an imprint of Taylor & Francis Group, LLC

Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

Except as permitted under U.S. Copyright Law, no part of this book may be reprinted, reproduced, transmitted, or utilized in any form by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying, microfilming, and recording, or in any information storage or retrieval system, without written permission from the publishers.

For permission to photocopy or use material electronically from this work, access [www.copyright.com](http://www.copyright.com) or contact the Copyright Clearance Center, Inc. (CCC), 222 Rosewood Drive, Danvers, MA 01923, 978-750-8400. For works that are not available on CCC please contact [mpkbookspermissions@tandf.co.uk](mailto:mpkbookspermissions@tandf.co.uk)

*Trademark notice:* Product or corporate names may be trademarks or registered trademarks and are used only for identification and explanation without intent to infringe.

ISBN: 9780367490614 (hbk)  
ISBN: 9781003044291 (ebk)

---

# Contents

Contributors .....	vii
<b>Chapter 1</b> Smart Software for Real-Time Image Analysis.....	1
<i>Marius Popescu and Antoanela Naaji</i>	
<b>Chapter 2</b> A Best Practice for Establishing a Telemedicine Project .....	29
<i>Dina Ziadlou, Seyed Mojtaba Mir Hosseini, Nasim Nasiri, and Amin Saberinia</i>	
<b>Chapter 3</b> Intelligent Algorithms for the Diagnosis of Alzheimer’s Disease.....	51
<i>Sarah A. Soliman, Rania R. Hussein, El-Sayed A. El-Dahshan, and Abdel-Badeeh M. Salem</i>	
<b>Chapter 4</b> An Efficient Protein Structure Prediction Using Genetic Algorithm .....	87
<i>Mohamad Yousef, Tamer Abdelkader, and Khaled El-Bahnasy</i>	
<b>Chapter 5</b> Evaluating the Effectiveness of Healthcare Services by Using the Method of Data Envelopment Analysis .....	107
<i>Vitalina Babenko</i>	
<b>Chapter 6</b> Computer Simulation Study of Oscillation Mechanisms and Physical Properties of Nanosized Biostructures.....	125
<i>P.J. Kervalishvili and T.N. Bzhalava</i>	
<b>Chapter 7</b> Adapting Smartphone-Based Applications for Performance Improvement Metrics’ Tracking in Healthcare Facilities as a Managerial Tool.....	133
<i>Adriana Burlea-Schiopoiu and Koudoua Ferhati</i>	
<b>Chapter 8</b> Intelligent Approaches for Developing Knowledge-Based System for Diabetes Diet.....	149
<i>Ibrahim M. Ahmed, Marco Alfonse, and Abdel-Badeeh M. Salem</i>	

<b>Chapter 9</b>	An Approach for Designing of Domain Models in Smart Health Informatics Systems Considering Their Cognitive Characteristics.....	171
	<i>Olena Chebanyuk and Olexandr Palahin</i>	
<b>Chapter 10</b>	Medical Images of Breast Tumors: Classification with Application of Hybrid CNN–FNN Network .....	195
	<i>Yuriy Zaychenko and Galib Hamidov</i>	
<b>Chapter 11</b>	Artificial Intelligence-Empowered Mobile Healthcare: A Wireless Access Perspective .....	211
	<i>S. Lenty Stuwart and El-Sayed A. El-Dahshan</i>	
<b>Chapter 12</b>	Computational Intelligence in the Identification of COVID-19 Patients.....	229
	<i>Shaymaa Adnan Abdulrahman and Abdel-Badeeh M. Salem</i>	
<b>Index</b>	.....	245

---

# Contributors

**Tamer Abdelkader**

Faculty of Computer and Information  
Sciences  
Ain Shams University  
Cairo, Egypt

**Shaymaa Adnan Abdulrahman**

Department of Computer Engineering  
Imam Ja'afar Al-Sadiq University  
Baghdad, Iraq  
Ain Shams University  
Cairo, Egypt

**Ibrahim M. Ahmed**

Department of Information Technology,  
Faculty of Computer and Information  
Technology  
Karary University  
Khartoum, Sudan

**Marco Alfonse**

Department of Computer Science,  
Faculty of Computer and Information  
Sciences  
Ain Shams University  
Cairo, Egypt

**Vitalina Babenko**

Full professor  
International E-commerce and Hotel &  
Restaurant Business Department  
V.N. Karazin Kharkiv National  
University  
Kharkiv, Ukraine

**Adriana Burlea-Schiopoiu**

University of Craiova  
Craiova, Romania

**T.N. Bzhalava**

Georgian Technical University (GTU)  
Tbilisi, Georgia

**Olena Chebanyuk**

Software Engineering Department  
National Aviation University  
Kyiv, Ukraine

**Khaled El-Bahnasy**

Faculty of Computer and Information  
Sciences  
Ain Shams University  
Cairo, Egypt

**El-Sayed A. El-Dahshan**

Professor  
Computational Physics  
Ain Shams University  
Cairo, Egypt

**Koudoua Ferhati**

University of Craiova  
Craiova, Romania

**Galib Hamidov**

Information Technologies Department  
Baku, Azerbaijan

**Seyed Mojtaba Mir Hosseini**

Kerman University of Medical Sciences  
Kerman, Iran

**Rania R. Hussein**

Lecturer  
Computer Science department  
Higher Technological Institute  
Cairo, Egypt

**P.J. Kervalishvili**

Georgian Technical University (GTU)  
Tbilisi, Georgia

**Antoanela Naaji**

Vasile Goldis Western University  
of Arad  
Arad, Romania



**Nasim Nasiri**

Kerman University of Medical Sciences  
Kerman, Iran

**Olexandr Palahin**

V.M. Glushkov Institute of Cybernetics  
National Academy of  
Sciences of Ukraine  
Kyiv, Ukraine

**Marius Popescu**

Western University of Arad  
Arad, Romania

**Amin Saberinia**

Kerman University of Medical Sciences  
Kerman, Iran

**Abdel-Badeeh M. Salem**

Full professor  
Ain Shams University  
Cairo, Egypt

**Sarah A. Soliman**

Assistant lecturer  
Department of Computers and  
Information, Higher Technological  
Institute  
10th of Ramadan, Egypt

**S. Lentu Stewart**

Department of Electronics and  
Communication Engineering  
University College of Engineering,  
Nagercoil  
Tamil Nadu, India

**Mohamad Yousef**

Faculty of Computer and Information  
Sciences  
Ain Shams University  
Cairo, Egypt

**Yuriy Zaychenko**

Institute for Applied System Analysis  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

**Dina Ziadlou**

Colorado Technical University  
Colorado, USA

---

# 1 Smart Software for Real-Time Image Analysis

*Marius Popescu and Antoanela Naaji*

## CONTENTS

1.1	Introduction .....	1
1.2	Turning Pictures and Videos to Record/Evidence .....	3
1.3	Theoretical Background of the Application .....	4
1.4	Description of the Application .....	8
1.4.1	Application Architecture and Module Description .....	8
1.4.2	Functionalities and Scenario/Photoplay .....	13
1.4.3	Testing, Installing, and Using the Application .....	17
1.5	Validation of the Application.....	20
1.6	Conclusions .....	22

## 1.1 INTRODUCTION

A digital image is the representation of a real image, being constituted as a two-dimensional data structure, which can be natural numbers, complex numbers, or real numbers represented on a finite number of bits. An image element is called pixel, the whole image being composed of a string of octets. The color components of pixel are red, green, and blue (RGB). The image formats are known according to their file extension, such as bitmap (BMP), the gross form of the image; Joint Photographic Expert Group (JPG), intended to compress the images taken from reality; Graphic Interchange Format (GIF), a good quality format and very strong compression; Tagged Image File Format (TIF), used for professional photos due to its versatility and nondestructive compression; Digital Imaging and Communications in Medicine (DICOM), used for medical images; and raw, which contain the entire information that was captured by the sensor at the moment of the shooting.

In real and legal practice, there are situations where fraudulent images are used. Detecting a fake ordinary image or video edits is not an easy task. Until recently, only few software could determine whether an image was original or whether someone manipulated it. Currently, the image can be cracked by using a technique called Deep Video Portraits (DVP) [1], which allows the imitation of a great variety of movements, including a three-dimensional (3D) head view, its rotation, shifting it, and even blinking. This uses artificial intelligence by which movement data are taken

from the source and transmitted to the digital image that is to be integrated into the video. Through artificial intelligence, motion patterns are collected, analyzed, calculated, and predicted in order to create a more realistic video [2]. DVP is a very effective method for creating computerized animations, converting into realistic photo images, and is extremely easy to confound with a person's real image.

Some software that implement many filters that work automatically or semiautomatically are already available. Software developers have created 3D reconstruction algorithms over 50 smart filters, most of which are unique. These algorithms are part of the specialized software, and some of them are embedded into the visual video cameras. There are companies that patent at least five new technologies related to photo and video processing. Among these, there are algorithms that allow to perform 3D face and object reconstruction, measure the dimensions and distances between objects on photo and video, analyze the motion model of an object in a frame, etc. VOCORD VideoExpert is a powerful tool used for analyzing and improving digital photo and video materials [3]. The program restores indescribable objects on the original photos or videos – faces of people, license plates, or text, greatly improving the quality of the material.

Amped Authenticate is another software package used to authenticate forensic images and the incorrect detection of digital photos [4]. Current products and processes focus on one or several scientific instruments. With the help of authentication, *Amped* empowers law enforcement experts and legal medicine laboratories with an instrument capable of detecting any manipulation that has been done on an image and checking whether a digital photograph has been generated by a specific device. Previously, specialized applications have been highlighted in this domain, but there are other applications that can perform a similar analysis of the image (satellite), the best known being Peripheral Component Interconnect Geomatica [5], Environmental for Visualizing Images (ENVI), or ER Mapper [6].

Exelis and Environmental Systems Research Institute have joined to improve the interoperability between image processing and geographic information system (GIS) software through ENVI and ArcGIS. Practically, ENVI is a selection of workflows and tools that have been integrated into the ArcGIS ArcToolbox environment.

Most existing applications in this domain are professional applications that offer, in most cases, the “package” functionality at a fairly high cost. As a rule, they are marketed as a trial version, valid for a limited time, or as a simple editor or viewer that does not include any functionality. In addition to this impediment, many of them come as third parties, which require integration with other applications. An example is ENVI. If it is desired to use some of the processing or methods that it offers, another application, such as ArcGIS, which may be available at a cost, is required in order to enable the use of the ENVI functionalities.

These applications may also require a user guide or course as their complexity makes them inaccessible for some users. Few of them even require an Internet connection for certain packages and functionalities. Our software was developed in order to avoid these drawbacks of the existing applications and make it accessible to everyone.

This chapter is organized as follows: the second section presents some techniques related to image processing, the third section is about the theoretical background, the

fourth section presents the application and its implementation, and the fifth section evaluates and validates it. Finally, few conclusions are presented in the conclusion section.

## 1.2 TURNING PICTURES AND VIDEOS TO RECORD/EVIDENCE

Image processing is a complex and dynamic domain with numerous applications in many areas. New technologies used in the image processing industry allow the application of new algorithms and methods for extracting useful information from the image: parallel algorithms, neural network-based classifiers, etc. There are also real-time systems implementations based on the image processing, systems such as those used in medicine, or the ones for the automated process control. Many improvements have also been made for the systems that process fingerprints, face, or writing recognition, etc.

The bias of the imaging can be performed by transforming the color components of each pixel.

The brightness of an image can be changed by adding a value to the color components of each pixel so that it does not come out of the 0.255 range. If a positive value is used, a lighter luminosity will be obtained, and if a negative value is used, a darker luminosity will be obtained.

The rotation of an image can be achieved by mapping an input pixel with the output position, determined by rotating with an angle, around an origin. Translating an image means moving the pixels from the original image, with a value that the user specifies, into the new positions.

The reflection or mirroring converts the original image so that the pixels are reflected from an axis specified in the new position in the target image. The reflection may be relative to a horizontal axis of order or to a vertical axis of abscissa. Mirroring may be after an axis pointing in an arbitrary direction also, which passes through the point of origin. In general, image processing involves two essential procedures: fusion of images and pseudo-coloring of images. Image fusion is a process of combining relevant information from a set of images into a single image, where the resulting image will contain more information than any of the images used before the merging [7]. Various methods have been developed to achieve the fusion of images. Some regions of the images, which are focused, have pixels of greater intensity than the rest.

Thus, the arithmetic mean algorithm is a way to get an image with all the focused regions. The algorithm consists in calculating the pixel average of the input images, and the resulting value is assigned to the pixel in the output image. The operation is repeated for each pixel in the image [8].

The algorithm of the maximal selection chooses the focus regions of each input image by electing the highest value of each pixel, resulting in an image with highly focused regions.

Discrete wavelet transform (DWT) is a process of image decomposition in low-high, high-low, and high-high space frequency bands on different scales, and the low-low band at the largest scale [9]. The low-low band contains the average of the information in the image, while the other bands contain information regarding the direction due to the spatial orientation.

*Principal component analysis (PCA)* is a mathematical *tool* that converts a number of correlated variables into a number of uncorrelated variables [10]. The algorithm is often used in compressing and classifying images. The uncorrelated transformed variables are called main components. The first main component is considered to be along the direction with the maximum variation. The second main component is constrained to stretch in the subspace that is perpendicular to the first one. The third main component is taken in the direction of the maximum variation in the subspace that is perpendicular to the first two and so on.

Pseudo-coloring, in the image processing [11], represents the color assignment for gray values according to a specific criterion. The term “pseudo-coloring” emphasizes the fact that these colors are artificially assigned in opposition to the true colors [12]. As in the fusion of images, pseudo-coloring can also be performed through different techniques.

Intensity slicing is a technique that can be better explained by interpreting the grayscale image as a 3D function that is cut by a plane that is parallel to the plane of the image. The intensity-color transformation is a method that involves performing three independent transformations on the intensity of any input pixel. The results are placed separately on the RGB channels, producing a composite image that has its colors modulated by the transformation functions.

In image acquisition, noise is the result of errors and is expressed by pixel values that do not reflect the intensity of the true image. Thus, if the image is a photocopy of a photo, the granulation of the film or photographic paper is a source of noise. If the image is purchased directly via satellite/digital camera, then the acquisition equipment can bring noise to the image.

Also, satellite image transmission can produce noise. Some image components or noise reduction are favored by filtering, which is a technique used for modifying or improving the image in order to favor certain components or remove other components. The noise of the image can be reduced by replacing the value of each pixel with the median of neighboring pixel values.

Median filtering is an effective image smoothing technique for noise abatement; however, it has the disadvantage of affecting the contours of the objects included in the image. Contours are critical in an image; therefore, we need to eliminate the noise. For a moderate noise level, the median is the ideal way to remove the noise, preserving the contours. Digital image filters are used to edit digital images in order to transform or improve them. They can also be used to modify or retouch photos, apply special effects, or make unique transformations using light and deformation effects. For example, the filters provided by Photoshop or ImageJ are organized by categories according to the desired result: stylistic and artistic effects, geometric distortions (creating 3D effects or other reforming effects), pixel displacements, etc.

### 1.3 THEORETICAL BACKGROUND OF THE APPLICATION

Analyzing a specific object within an image requires its contour to be highlighted with the help of the image segmentation procedure. Segmentation and contour

extraction are important steps in image analysis. The segmentation process is considered as a process of classifying objects in an image. The purpose of segmentation is to simplify the representation of images in useful information by partitioning into areas. Although there are high complexity algorithms, image segmentation remains application dependent, with no unique solution identified. The segmentation method is used to detect certain objects or contours in an image. The contour of an object is given by the set of pixels of the object that have at least one neighbor outside the object. Segmenting images by this method involves locating pixels corresponding to contours of objects present in the image. After the acquisition of the image (e.g., through satellite telecommunications networks), it follows the analysis of the characteristics of a particular area/object in the image, which is a complex process having the following general processing stream: preprocessing, improvement and highlighting, and result processing.

The filters are used within each layer of the application flow, where algorithms and drivers are implemented, such as the salt and pepper filter, the low-pass filter, the high-pass filter, *contrast stretching*, *grayscale* processing, fusion, and pseudo-coloring of images.

The *salt and pepper filter/noise* is generally caused by the flaws of the photo/video sensor, memory location faults, or sync errors in the digitization process [13]. The *salt and pepper* is represented in an analytical form as follows:

$$\text{Histogram}_{\text{Salt\&Paper}} = \begin{cases} A & g = a(\text{paper}) \\ B & g = b(\text{salt}) \end{cases} \quad (1.1)$$

In the noise model type of salt and pepper, there are only two possible values. The likelihood of occurrence of each of them is generally less than 0.1. At higher values than these, the noise will dominate the image. For an 8-bit image, the typical intensity value for the pepper noise is approximately zero, and for the salt noise it is around 255.

The *low-pass filter* attenuates or blocks the passage of high frequencies, and allows only the unattenuated passage of the space low frequencies. They are usually used to reduce noise from images. The ideal low-pass filter [14] blocks all frequencies that are higher than the cut-off frequency  $\omega_0$ ,

$$H(k,1) = \begin{cases} 1 & (k^2 + l^2) \leq \omega_0^2 \\ 0 & \text{else} \end{cases} \quad (1.2)$$

and the Gaussian low-pass filter brings to the attenuation of the high frequencies without completely removing them as the ideal filter.

$$H(k,1) = e^{\left(\frac{-\omega}{\omega_0}\right)^2}, \omega^2 = k^2 + l^2 \quad (1.3)$$

The effects that result from this filter are capable of smoothing the image and without effects of ringing wave. It usually applies before segmentation (based on regions or edges) to eliminate noise. Other filters are the *butterworth*

$$H(k,1) = \frac{1}{1 + \left(\frac{\omega}{\omega_0}\right)^n} \quad (1.4)$$

and *trapezoidal filters* (which introduce a circular wave effect stronger than the Gaussian or butterworth, but weaker than the ideal one).

$$H(k,1) = \begin{cases} 1 & \text{for } \omega < \omega_0 \\ \frac{\omega - \omega_1}{\omega_0 - \omega_1} & \text{for } \omega_0 \leq \omega \leq \omega_1 \\ 0 & \text{for } \omega > \omega_1 \end{cases} \quad (1.5)$$

The *high-pass filter* attenuates or blocks the passage of low frequencies, allowing only the unattenuated passage of space high frequencies. The ideal filter passes high blocks all frequencies smaller than a cut-off frequency. The resulting image after applying this filter has a circular wave pattern. The Gaussian high-pass filter performs a gradual reduction of the low frequencies so that the high frequencies could pass unaltered through the filter.

$$H(k,1) = 1 - e^{-\left(\frac{\omega}{\omega_0}\right)^2}, \omega^2 = k^2 + l^2 \quad (1.6)$$

In practice, this filter is combined with a Gaussian low-pass filter, and thus results in the *Gaussian difference*. Another filter is the *butterworth* high-pass filter.

$$H(k,1) = 1 - \frac{1}{1 + \left(\frac{\omega}{\omega_0}\right)^n} \quad (1.7)$$

*Contrast stretching* is a simple technique for image improvement that improves the contrast of the image by “stretching” the range of pixel intensities so that it covers a range of desired values, for example, the entire range of values supported by the image to which it applies. This type of improvement is more sophisticated than the histogram equalization, where only a linear scaling function applies to the pixel intensities [15]. Before “stretching” can be performed, it is necessary to set the upper limit and lower limit of pixel value over which the image is normalized. Often, these limits are the minimum and maximum pixel values that the image supports. For example, for 8-bit *gray level* images, the lower and upper limits are 0 and 255, respectively.

*Grayscale processing* [16] is a very simple technique and consists in equalizing the three color components of each pixel. A commonly used method is the mediation of the three components.

*Image fusion* is a process of combining relevant information from a set of images belonging to the same scene into a single image [17]. This process has some requirements, namely: the merged image must contain all the input image information and the fusion of images must not introduce other artifacts which may lead to wrong analysis.

Image fusion methods can be divided into two groups: spatial domain-based fusion, which directly works on the pixels from the input image (in this group, we have algorithms such as minimum selection, maximum selection, arithmetic mean, or the analysis of the main component), and domain transformation-based fusion, which relies primarily on the image transfer in a frequency domain (in this group, we have algorithms such as DWT). In the case of the minimum selection, the result of the fusion is obtained by selecting the pixel with the minimum intensity in the fused images.

$$F(i,j) = \sum_{i=0}^m \sum_{j=0}^n \min(A(i,j), B(i,j), \dots) \quad (1.8)$$

where  $A(i,j)$ ,  $B(i,j)$  are input images and  $F(i,j)$  is the resulting image.

In the case of the arithmetic mean, the result of the fusion is obtained by calculating the arithmetic mean of the pixel intensity from the input images.

$$F(i,j) = \frac{A(i,j) + B(i,j) + \dots}{n} \quad (1.9)$$

*The main component analysis* is a subspace method that reduces multidimensional sets of data into smaller dimensions for analysis. This method determines the balance/ponderosity of each source image using its own proper vector at the matrix's own covariance value, for each source image.

*DWT* is a multi-resolution image decomposition technique that offers a variety of channels that represent the image characteristics in different frequency sub-bands. When the decomposition is accomplished, the approximation and detail components can be separated. DWT converts the image from the spatial domain to the frequency domain. The image is divided by both vertical and horizontal lines, representing the first DWT order. Let  $s(n_1, n_2)$  be the input image of size  $N_1 \times N_2$ , then the wavelet and scaling functions are:

$$\begin{aligned} \omega_{\Phi}(j_0, k_1, k_2) &= \frac{1}{\sqrt{N_1 N_2}} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} s(n_1, n_2) \Phi_{j_0, k_1, k_2}(n_1, n_2) \\ \omega_{\epsilon}(j_0, k_1, k_2) &= \frac{1}{\sqrt{N_1 N_2}} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} s(n_1, n_2) \epsilon_{j_0, k_1, k_2}(n_1, n_2) \end{aligned} \quad (1.10)$$



## 1.4 DESCRIPTION OF THE APPLICATION

The operating principle behind the application consists in taking a stream of image-type data and processing them along a series of modules and filters to get the final result that highlights some features. Each module represents a general step in the processing line and plays a well-defined role in the processing of information, with various algorithms, data structures, and implementation technologies. These modules are:

- The image acquisition module, which takes over high-resolution TIFF format images from an external source and transforms them into a data structure and stores them externally into easily accessible (JPEG) formats.
- The preprocessing module, which is an optional one but often very necessary. If the images on which the actual processing is desired require some “adjustments,” this module provides a number of filters that may prove to be useful.
- The module of image enhancement and highlighting the features of the images, which is considered to be the “heart” of the entire application, having the role of improving the features of the image by image fusion methods, and highlighting or mapping the regions of the image using pseudo-coloring techniques.
- The result processing module, which is the last in the execution thread and deals with the analysis of the data obtained in the previous step, comparing the resulting image and the areas highlighted with a legend that will determine the characteristics of interest.

### 1.4.1 APPLICATION ARCHITECTURE AND MODULE DESCRIPTION

The organization of the application under the “pipes and filters” pattern [18] is an ideal structure for the systems that process a data stream, each step being encapsulated into a filter that represents a *pipeline* processing unit.

The advantages of organizing the application under this pattern include flexibility in changing filters (changing between filters and the interface and functionality is easy), flexibility by recombination (ease of reconfiguring a *pipeline* for including new filters, or using the existing filters in another sequence), reuse of filters (small filters are easy to reuse if the environment allows them to be connected in a simple way), and efficiency through parallel processing (filters run on different threads, and from this point of view, pipes-and-filters systems can benefit from the advantage of a multiprocessor).

The disadvantages include expensive or inflexible information exchange (the information must be encoded, transmitted, and then decoded), gaining efficiency by paralleling processes is just an illusion (the cost of data and synchronization can be high), and error handling (the disadvantage of the structure lies in the dependence of the current module on the previous one and the fact that the occurrence of a possible data processing error in one of the modules from the first steps may propagate an error along the entire process line). Despite these disadvantages, this architectural pattern fits the application developed by us and presented in this chapter. The implementation of the application by using the “pipes and filters” pattern was done

in the Java programming language. The image pickup module has the role of uploading the image from an external source to the application (hard disk, CD/DVD), for processing. The application allows the processing of a single image as well as the simultaneous processing of two images. Thus, the user interface will have dedicated areas for uploading these images. Once the external files are taken in, they are intermediary saved in an easily accessible (JPEG) file format. This will be done after each processing, each step being seen as a filter that represents a processing unit.

Preprocessing prepares the information for the next stage, the processing step, by applying filtration operations. This step in the execution flow can be optional, depending on the user's requirements. From the previous module, data ready for processing reach this module in which it is decided to preprocess them. The steps that run here are similar to those of the image acquisition. The user can dynamically add the filter(s) and the order in which the image(s) passes. This is done with the ease of a JPanel where each process is represented by a *drop-down* menu called JComboBox, which contains the entire list of processes available in the application.

Improving and highlighting the image features is the main application module, including processing, such as fusion of images and pseudo-coloring. On the fusion of images, algorithms for minimum and maximal selection, fusion by calculating the arithmetic mean, and DWT were implemented, as well as an intensity slicing algorithm for pseudo-coloring, which uses seven coloring levels. The algorithms were implemented without using any external bookstore, or the Application Programming Interface (API). The image is read by using its location and then transformed into a BufferedImage-type object. The transformation into this object will allow an easy extraction of the image pixel matrix.

Thus, the following steps will represent the pure processing for accomplishing the above-mentioned methods. Image fusion requires two or more images. The application will only fuse for two images. Typically, of the two images, one will be a panchromatic image and the other one will be a different frequency band image (Table 1.1 [19]).

Once the two images are available as BufferedImage objects, the actual processing may start. The fusion technique consists in selecting the minimum pixel value from the same position in both images and creating a new image with them. This theory

**TABLE 1.1**  
**Wave Length Bands**

Band	Name	Band width( $\lambda$ , $\mu\text{m}$ )	Space resolution
1.	Blue	0.45–0.515	30 m
2.	Green	0.525–0.605	30 m
3.	Red	0.63–0.69	30 m
4.	Near infrared	0.75–0.90	30 m
5.	Shortwave IR-1	1.55–1.75	30 m
6.	Thermal IR	10.4–12.5	60 m/120 m
7.	Shortwave IR-2	2.09–2.35	30 m
8.	Panchromatic	0.52–0.9	15 m

applied in our situation undergoes some changes. As shown in Table 1.1, the eight-band image has a much higher resolution than the others, in which case the pixels in the same position will no longer match. However, whatever the second image will be, excluding the one of the eight bands, the following relationship will occur:

$$\text{panchromatic.dimension}() = \alpha * \text{secondImage.dimension}();$$

where  $\alpha$  represents how many times the eight-band image is larger than any of the other images. If the value of  $\alpha$  is set, then the selection can begin. A first step will be to find out which of the two images is larger. Thus, the dimensions of the two images will be calculated and will be compared in order to determine which one is the eight-band image. Once the dimensions are calculated, it is known which of the two images has the higher resolution, but the value of  $\alpha$  is still unknown. In the present case,  $\alpha$  will consist of two components:  $\alpha W$ , which represents how many times the eight-band image is larger in width than any other image, and  $\alpha H$ , which means the same thing, but in height. Finally, it will be determined how many pixels in the panchromatic image correspond to a pixel from any other image. Finding this, we can begin to scroll through the two images and select the minimum. The browsing is performed using two repetitive type *for*. structures, one in width and the other one in height. The pixels are extracted from the `BufferedImage` object as follows:

```
int pixelValue = bufferedImage.getRGB(x,y);
int redPixelValue = (pixelValue & 0x00ff0000) >> 16;
int greenPixelValue = (pixelValue & 0x0000ff00) >> 8;
int bluePixelValue = pixelValue & 0x000000ff;
```

where  $x$  and  $y$  are the coordinates from which we wish to extract the pixel, and `redPixelValue`, `greenPixelValue`, and `bluePixelValue` are, respectively, the intensity values for the red, green, and blue colors of that pixel.

In this case, the image will be gray scale, and all three values for red, green, and blue are equal; therefore, we only need one of them. Extracting the pixels in the above case is only valid for the lower resolution image, and for the panchromatic image we will have to extract the corresponding  $\alpha W \times \alpha H$  pixels. Having all the pixel values in the eight-band image and of the pixel in the other image, one can unanimously calculate these values, and that minimum will be the pixel from the current position of the final image. The maximum selection method consists in selecting the maximum pixel value from the same position in both images and creating a new image with them. This method is almost identical to the fusion method by selecting the minimum, the only thing that changes being the one from the final step.

Thus, instead of calculating the minimum value between all the pixels in the panchromatic image and the pixel in the different band image, the maximum will be calculated, and this will be a result in the final image. The fusion by calculating the arithmetic mean uses the same principle, instead, at the final step, for the pixels extracted from both images, the arithmetic mean is calculated. The value resulting from this calculation is the pixel value in the result image.

The DWT method is slightly different from the one presented above, due to the fact that it may contain them, plus high-pass and low-pass filters in the preprocessing module. Implementation of this method involves the following three steps:

- Decomposing the image by a coefficient, using high-pass and low-pass filters
- The fusion of the decomposed images using a simple method previously described
- Composing the resulting images until reaching one image, namely, the final image. Both loaded images will undergo the same decomposition transformations. The initial images will be applied by a high-pass and a low-pass filter, resulting in two images. The two resulting images will also be submitted to the same methods and filters. This transformation will be repeated by a number of times equal to the decomposition coefficient. The decomposition coefficient can be entered by the user from the keyboard. Finally, after decomposing, there will be  $2n$  (where  $n$  is the decomposition coefficient) for the panchromatic band and the other band of choice. Following each decomposition process, the result will be saved in the project directory. The images resulting from applying the high-pass and low-pass filters on the first image will fuse, using a simple fusion method, with the images resulting from applying the filters to the second image. As in the decomposition coefficient, the fusion modality can be selected by the user. Having the  $2n$  merged images, their composition may begin. This composition is also called *inverse discrete wavelet transform* and consists in adding the pixels from these images until a single image is reached, which is the final image. All the states of this process are presented in Figure 1.1 [19].

The last step of this module is the pseudo-coloring function. Once we have the enhanced image, we can apply the transformation in order to map the areas of the image in color. For pseudo-coloring, we will use the intensity slicing procedure with seven ramps for coloring. Thus, the image will be browsed pixel by pixel, and for each pixel a gray value will be drawn; according to this value, it will fit in a certain color as shown in Table 1.2 [19].

After this processing, the final image, which will be transmitted to the resulting processing module, will be analyzed. In order to better highlight the relationships between all these functionalities, a class diagram was added for the improvement and highlighting of the features belonging to the preprocessing module presented (see Figure 1.2)[ 19].

The resulted processing module is the part where the processed image, ready to be analyzed, is shown to the user. The actual displaying is done in a JFrame object type that will contain a panel with the final image. In addition to this panel containing the final image, there will be a panel with a legend in which the meaning of each color, and therefore each region, plus a set of rules or combinations of results will be presented (because the classification of the images based on the information extracted from one channel is quite inconclusive) to help with this analysis process. In order to get as much information as possible, needed for classification, combinations of bands

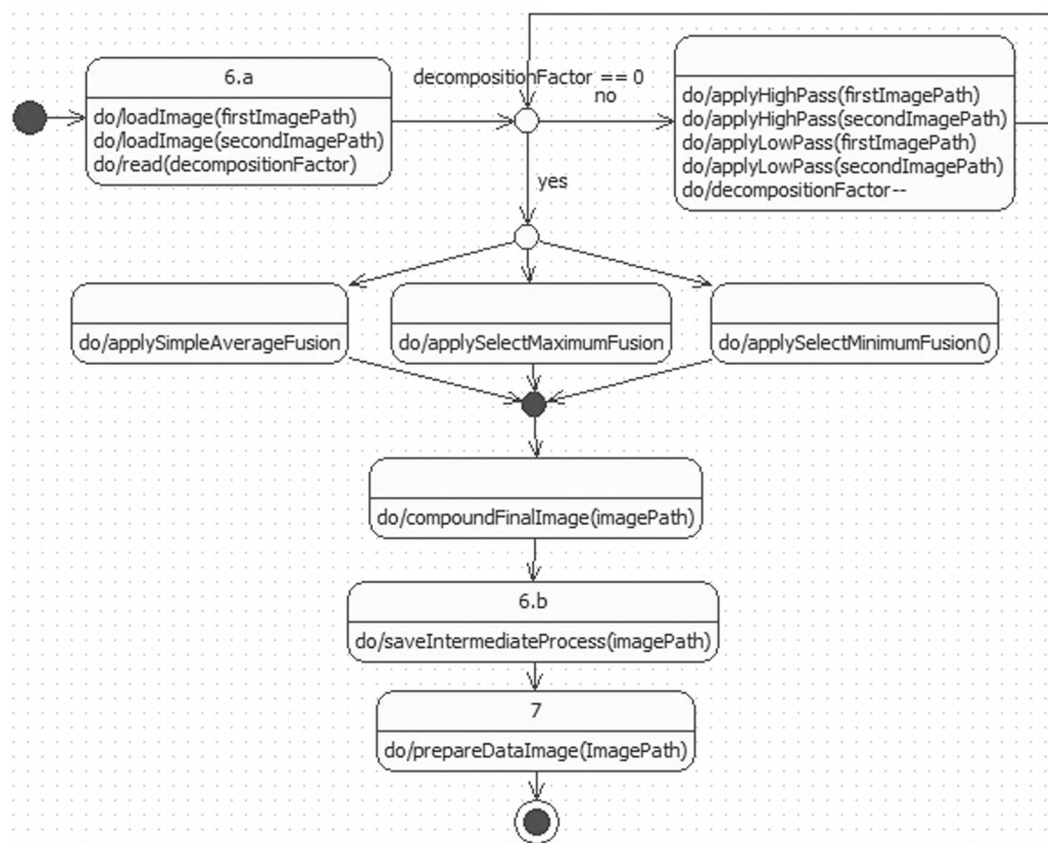


FIGURE 1.1 State diagram for discrete wavelet transform.

**TABLE 1.2**  
**Pseudo-Coloring – Calculation of the Color Ramps**

Grey Intervals	Color Transition	Calculation of the Color Intensity
0–72	Red ->yellow	Color(255, (grayintensity*3), 0)
73–108	Yellow->green	Color(255 – ((grayintensity – 72)*7, 255, 0)
109–144	Green->bleu ciel	Color(0, 255 – ((grayintensity – 108)*7))
145–180	Bleu ciel->blue	Color(0, 255 – ((grayintensity – 144)*7),255)
181–216	Blue->pink	Color(((grayintensity – 108)*7),0,255)
217 – 255	Pink->red	Color(255,0, 255 – ((grayintensity – 216)*7))

are used or various indexes are calculated. Depending on the module in which it is assisted or not, in case of the classification made by a human specialist, the supervised and unsupervised classifications are distinguished.

The supervised classification is achieved when all existing knowledge concerning the studied area is taken into account, and the classification is assisted by a human expert in the field. Unsupervised classification is based only on the results obtained from pixel-based calculations without any supervision by the expert. This way of classification is useful when studying hard-to-reach areas or areas not yet explored. Practically, the classification consists in determining the attributes that can be mapped on the areas that were identified after processing an image.

The proposed modalities of classification for these types of images are:

- The Anderson classification, which is a hierarchical one, where the information is clearly structured into classes and subclasses. This classification system is mainly used to identify different entities at national level.
- The decision trees-based classification. This algorithm uses certain rules learned to classify the pixels of an image [20].

#### 1.4.2 FUNCTIONALITIES AND SCENARIO/PHOTOPLAY

The user has total access, benefits from all the functionality offered by the application, and can control a large part of them with the help of the user interface available.

The functionalities and settings to which the user has access are the following:

- Selecting the number of images, when the entire execution thread can be made on a single input image, or on two such images. If the transition through the image enhancement module is considered, then two images are required, but the application does not constrain the user to do so. Thus, the user can choose the number of images that he or she wishes to load, provided that the image fusion process is used or not. Another flexibility provided in the scenario where two images are loaded is the order of their loading, which does not matter.

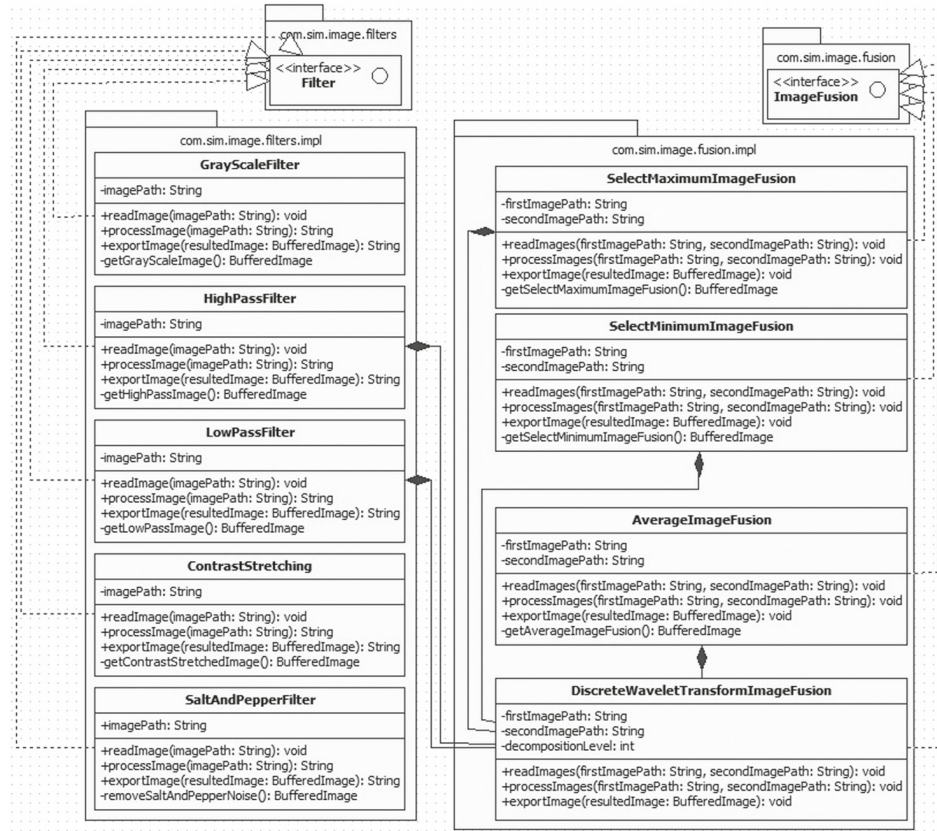


FIGURE 1.2 Class diagram for improvement and highlight of the features.



- Selecting the number of times it was processed, when the user has two buttons that allow him to add or remove a step from the execution thread. The number of these steps is not a default one but a choice of the user.
- Selecting the order of processing, when the same can be said also about the order in which the processes are to be executed. When the user adds a process, he or she basically adds a step containing a list of processes that he or she can choose from. That list of processes represents all the processing methods that the application has at its disposal. In this way, the order and the number of processes are not imposed but determined by the one using the application.
- Selecting the degree of decomposition, if the image fusion method called DWT is chosen as a step in the execution thread, the user will be allowed to choose the degree of decomposition. This degree of decomposition represents the number of times the high-pass and low-pass filters are applied to the image. This number should be between 1 and 5 inclusive. Choosing a higher degree of decomposition is not allowed, as it would result in cumbersome processing that would decrease from the performance of the application.
- Selecting the fusion method, when choosing the method of merging decomposed images will be allowed (in the case of the DWT merger). One of the three methods (minimum selection, maximum selection, or arithmetic mean calculation) can be selected by the user as a method of fusing between decomposed images. Two cases will be analyzed.
- For single image processing (considered as a scenario: image upload, application of the *grayscale* filter, also of the *contrast stretching* one, and pseudo-coloring), the order of the steps will be as shown in Figure 1.3.
- Creating the project directory, when the application is launched, named after the current date and time.
- Loading the image, the user using one of the two reserved spaces to load the desired image; this will need to have the .tif extension.
- Saving the intermediate image that is saved in the project directory.
- Applying the *grayscale* filter, which applies to the image saved in previous step.
- Saving the intermediate image; this is saved in the project directory.
- Applying the *contrast stretching* filter; this applies to the image saved in previous step.
- Saving an intermediate image that is saved in the project directory.
- Pseudo-coloring, which is applied to the image saved in previous the step.
- Saving the intermediate image; this is saved in the project directory.
- Showing the final image in an application frame.

For processing on two images (considering as a scenario: loading images, applying DWT fusion to the degree of decomposition 2 and arithmetic mean fusion, pseudo-coloring application), in which case the order of the steps will be:

- Creating the project folder, named after the current date and time, when the application starts.
- The user will use the two reserved spaces to upload the desired images, which will need to be in the TIFF format, meaning the .tif extension.



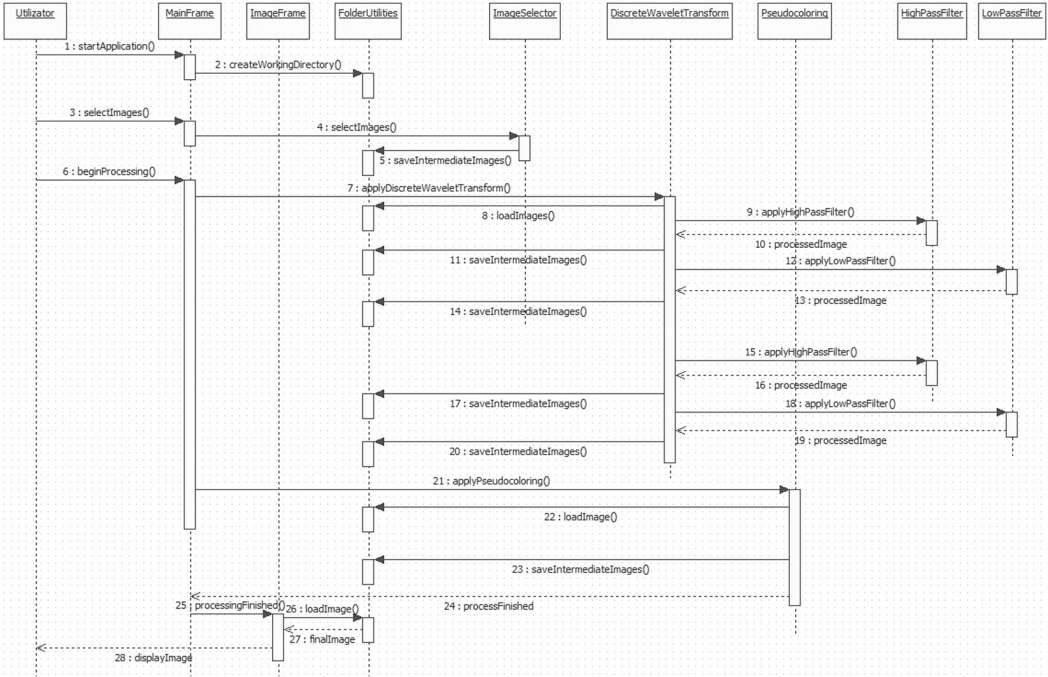


FIGURE 1.3 Sequence diagram – scenario.

- Saving the loaded images in the project directory.
- Applying to both images that were saved at the previous step, the low-pass and high-pass filters, for the decomposition degree 1.
- Saving the images resulting from the application of the filters in the project directory.
- Applying to the images saved in the previous step, the low-pass and high-pass filters, for the decomposition degree 2.
- Saving the images resulting from the application of the filters in the project directory.
- The fusion of images by calculating the arithmetic mean of two-by-two images, from those saved in the previous step, is performed.
- Saving the images resulting from the merger in the project directory.
- Composing the saved images for shifting from the decomposition degree 2 to decomposition degree 1.
- Saving the images resulting from the composition in the project directory.
- Composing the saved images for the transition from the decomposition degree 1 to the decomposition degree 0 is achieved (thus, a single image is obtained, i.e., the final image).
- Saving the image resulting from the composition (intermediate image saving) in the project directory.
- pseudo-coloring is applied on the image saved in previous step;
- Saving the image resulting from the pseudo-coloring in the project directory.
- Displaying the final image into an application frame.

### 1.4.3 TESTING, INSTALLING, AND USING THE APPLICATION

The testing of the application will consider the analysis of the application's features from different points of view. To highlight these features, a series of tests will be performed that address certain aspects such as performance, precision, or external factors.

The first test will be the performance one, which highlights the time the application manages to process an input image to a final result ready for analysis. In the present case, this would be hard to show, and even if it were done, it would not indicate any precise timing.

Determining accurate values to indicate the performance of the application depends largely on two factors: input images and processes in the execution thread.

The images on the bands may vary in size, variation that is not a small one, or one that can be neglected; there may be cases with differences of the fourth degree in the sizes of two images.

The flexibility of the program allows the user to choose the execution thread as desired. Thus, there may be processes in which only a simple fusion and a pseudo-coloring are used, resulting in relatively little processing time, or there may be processes that include 2–3 filters in the preprocessing module, a fusion of images using DWT with a decomposition factor equal to 5, and a pseudo-coloring, in which case the execution time would be totally different from the first situation. Considering this situation, performance tests will be applied on each algorithm in part on two different