Multidimensional GIS for satellite imagery analysis

Krzysztof DRYPCZEWSKI[(1)], Łukasz MARKIEWICZ[(2)], Andrzej STEPNOWSKI[(3)]

Faculty of Electronics, Telecommunications and Informatics
Gdansk University of Technology
Gabriela Narutowicza 11/12, 80-233 Gdańsk
[(1)]kszyk@eti.pg.gda.pl, [(2)]lukasz.markiewicz@eti.pg.gda.pl, [(3)]astep@eti.pg.gda.pl

Multidimensional Geographical Information System allows storing, querying and processing of multidimensional query data. It is able to process satellite imagery and provide tools for its analysis. In the article authors present the developed system that analyzes a time series of SENTINEL - 1 mission satellite imagery acquired over the coast of Poland. The algorithm used finds and detects changes in the shape of the coastline over a long period of time.

The system uses a Raster Data Manager array database management system to simplify the process of data querying, trimming, storing and analysing. Authors present how the recent trends in GIS development, like RASDAMAN, can be applied to satellite imagery processing.

Keywords: Multidimensional GIS, Raster Data Manager, RASDAMAN, SENTINEL-1, SAR, Coastal Monitoring, Coast Change Detection

1. Introduction

Geographic Information System (GIS) is a computer system designed to process, manage, analyze and disseminate spatial data. Recent trends in GIS focus not only on storing information about spatial features, but also providing information about changes in features characteristics, typically over time. A GIS system able to query data over spatio-temporal domain is called a multidimensional GIS. Capabilities of such systems can make it a useful tool in remote sensing and Earth Observation data analytics[1][2].

In the article, authors present multidimensional GIS for satellite imagery analysis. As a test case presenting its capabilities, a time series of imagery from Sentinel-1 mission is used. Sentinels SAR (Synthetic Aperture Radar) data is commonly used in maritime and land

applications, for example: oil spills, floods, bottom topography, surface mapping, land deformation, geology and coastal zone monitoring, including ship detection, changes of seashore, ports and marine infrastructure [3][4][5].

In the aforementioned system, satellite imagery time series are stored in a multidimensional database to simplify data access and processing. Change detection output and analysis results are presented using various man-made and natural coastal areas in Northern Poland.

2. Multidimensional GIS

Multidimensional data carries (up to) 4-levels of information about geospatial features (variables values), typically in relation to a spatio-temporal domain. Any phenomenon becomes a spatio-temporal array of specific cell type or dimensionality when it is prepared for usage in a GIS system. Typical example of this kind of data is an imagery time series, in general referred to as Multidimensional Discrete Data (MDD). Processing and analysis of MDD requires spatial and temporal operators that can handle dimensionality. These tasks can be simplified using the OLAP (OnLine Analytical Processing) model. It is a dedicated solution, which consists of an array with any amount of dimensions - each of them represented by a different data parameter. MDD is modelled as a cube divided into smaller fragments. This structure must be stored in a dedicated database, which additionally enables the hierarchical arrangement of data [6][7][8][9].
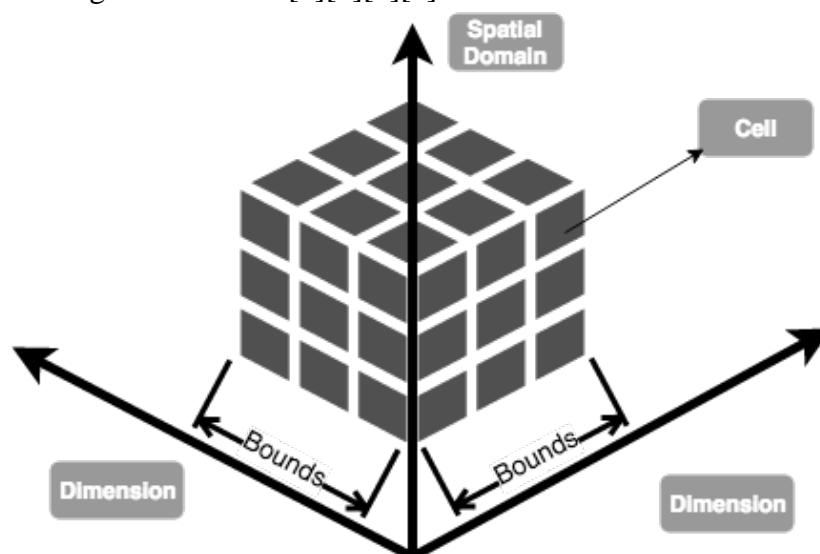

Fig. 1. Data cube.

Fig. 1. presents the idea of a 3-D data cube. Data cube stores a number of cells. Cell type can be a simple scalar value, or a more complex structure (for example scalar value for grey scale image or 3/4 element tuple for RGB/RGBA channels in true color imagery).

The data cube can have an arbitrary number of dimensions though one of them must have spatial reference and is called a spatial domain. Maximum and minimum values for dimension parameters can be specified for each of the dimensions as two values tuple: lower and upper bounds [6].

3. Raster Data Manager

3.1. Overview

Raster Data Manager (*RASDAMAN)* is an array database management system (ADBMS) and GIS application server. It was approved by an Open Geospatial Consortium (OGC) as a Web Coverage Service (WCS) 2.0.1 reference implementation. Rasdaman allows a storage and manipulation of MDD in an integrated server application[7][8]. It provides data definition (*rasdl*) and selection (*rasql*) languages. Due to its client-server architecture, Raster Data Manager allows queries to be processed entirely on the server side (Fig. 2.), which enables the creation of multidimensional web-GIS systems[10][11][12][13].
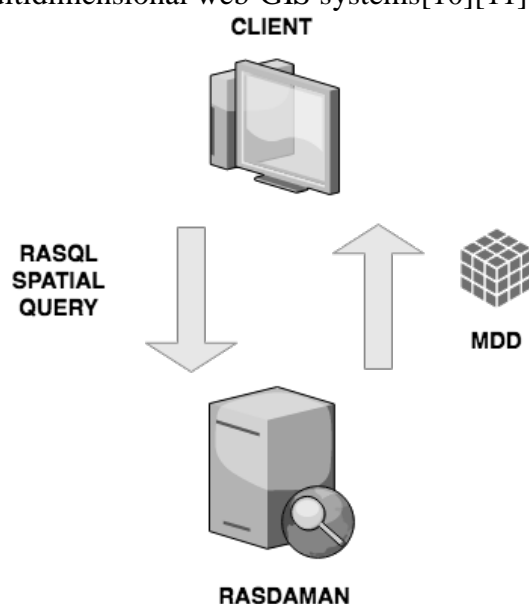

Fig. 2. RASDAMAN: client - server data flow

Rasdaman provides a database service for the MDD structures in a domain independent manner. It extends the standard relational database with the ability to store multidimensional raster data and is available in both open-source and commercially supported versions. MDD is processed by Rasdaman and stored in the underlying database, called base database. Open-source Rasdaman can be easily embedded in PostgreSQL (commercial version also in MySQL, Oracle, IBM Informix and DB2 databases). Alphanumerical information that is not in an array form is stored outside Rasdaman, but reference between alphanumeric and MDD data can be established [11].

3.2. RASDAMAN data definition - RASDL

To create a data cube in Rasdaman one must first define the type of information that is inserted into the database. Rasdaman offers a special language for this purpose: Rasdaman Definition Language, Rasdl [11]. Rasdl has syntax similar to C++ and allows all primitive C++ built-in types (char, float, int etc.) and structs (tuples of variables of primitive types). User-made type definitions are combined with header files generated by Rasdl preprocessor and then loaded into Rasdaman.

In general, type definition process has three steps:
1) cell type definition - allows basic C++ types and structs

2) MDD definition as an array of cells of base type (one dimension must be a spatial domain)
3) collection type definition as a set of MDD arrays of the same type

General syntax for cell type definition is as follows:
struct structName { type1 atribute_name1, type2 atribute_name2 ... }

General syntax for MDD definitions (marray keyword is used to define MDD):
typdef marray <lower_bound1,upper_bound1 : ... : lower_boundN,upper_boundN>

Define cell type for colour imagery as a struct of three 8-bit char values for RGB channels:
create type RGBPixel (red char, green char, blue char)
Define 3 dimensional data cube of RGBPixels of size 10 x 20 x 30:
create type RGBCube as RGBPixel mdarray [10, 20, 30]

Define type for grayscale imagery as a two-dimensional array of 8-bit (char) types of size x and y:
create type GreyImage as char mdarray [x, y]

Define a set of grayscale imagery:
create type GraySet as set (GrayImage)

3.3. RASDAMAN query language - RASQL

Rasdaman query language [11][12] (RasQL, rasql) enables the user to communicate with databases by usage of specific queries. It allows for MDD data querying, retrieval and manipulation. In general terms, query is send to the rasdaman server for evaluation and if query is valid, results are sent back as string or file (CSV, TIFF etc.). RasQL syntax is similar to structured query language (SQL) and extends standards defined by the International Organization for Standardization (9075, 1992). It provides array algebra that includes MDD operators. As in the SQL, RasQL returns set of arrays, that can be optionally encoded into an image (PNG, TIFF, JPEG). Spatial indexing, compression and tiling are also provided.

RasQL has a syntax based on SQL and is using well-known keywords as *SELECT, WHERE* and *FROM*. To query parts of arrays, an "[]" operator is provided. It is worth to mention, that MDD operations can be used both in SELECT and WHERE clauses. General SELECT query looks as follows:

SELECT result_specification WHERE columns_names FROM conditions

For example - query a two-dimensional "data" collection, then select pixels in range of [100:150,40:80] that have a cell value bigger than 150:
SELECT data[100 : 150, 40 : 80] FROM data WHERE some_cells (data> 150)

To invoke RasQ, "rasql" command line tool can be used. "-*q*" parameter is used to specify query string, and "-*out*" to specify output format. Below, several example queries are presented.

Get available type definitions using RAS_SET_TYPES system variable:
rasql -q "SELECT m FROM RAS_SET_TYPES as m" --out string

Get available collections using RAS_COLLECTIONNAMES system variable:
rasql -q 'select c from RAS_COLLECTIONNAMES as c' --out string

Encode "data" collection using tiff format:
rasql -q 'select encode(data, "tiff") from data' --out file

Trim "data" collection (select pixels in range):
rasql -q 'SELECT encode(m[210:600,0:300], "png") FROM data AS m' --out file
rasql -q 'SELECT encode(m[50:100,0:], "tiff") FROM data AS m' --out file*
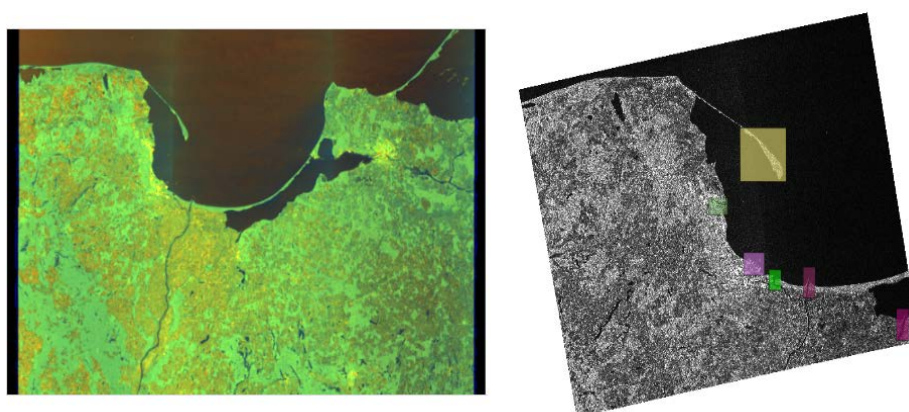
4. Coastal change detection

4.1. Sentinel-1 SAR data

ESA satellite missions called Sentinels are being developed as a very important element of the Copernicus programme. These missions use radar, optical and other sensors for multi-purpose global remote monitoring [14]. The first satellite from planned Sentinel missions was launched in April 2014. Currently 2 satellites are operational, Sentinel-1A and Sentinel-1B (launched 2 years later). Each is equipped with C-band synthetic aperture radar 12m in length. Sensor sends probing signals in the microwave band with different polarizations during transmitting and receiving [15]. VH (vertical-horizontal) mode is used by the authors for the analysis described in chapter 4.3.

SAR data is weather and day/night independent. Various products, raw and pre-processed are available on the Copernicus Open Access Hub [16]. Pre-processed Ground Range Detected product (Level-1 GRD) is used. It is projected on an Earth model, and is suitable for image processing analysis, including change detection algorithm, described in the next chapter. Data is acquired in Interferometric Wide Swath mode (IW) and the ground resolution is 21m.

On Fig. 3. original data (on the left) and pre-processed subsets (on the right) used by the authors are shown in native image format.
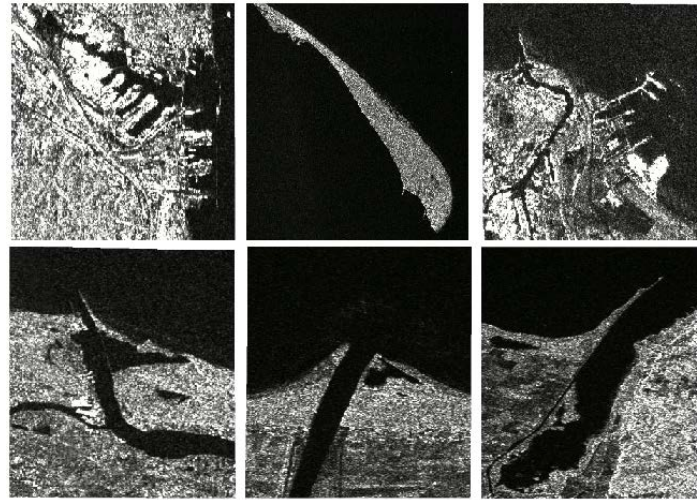
Fig. 3. SAR data and subsets used by the authors - Port of Gdynia, Hel Peninsula, Port of Gdansk, Vistula Branch near Gdansk Stogi, Vistula mouth, Bay of Elblag

4.2. Change detection using background subtraction

Fig. 4. presents diagram of the whole developed system. In the upper part of the figure the procedure of change detection is shown.
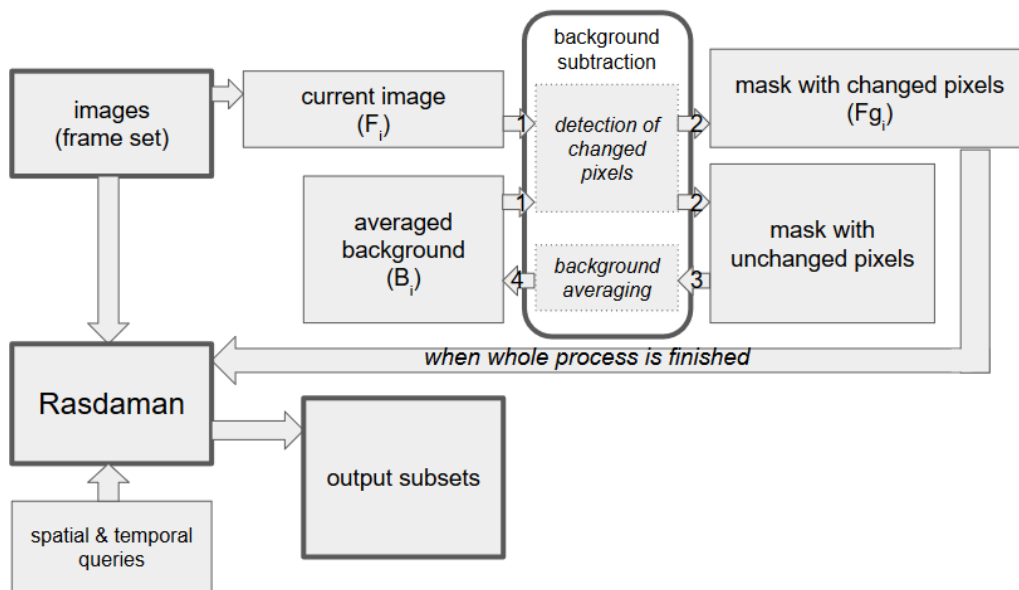

Fig. 4. Diagram of the system, including background subtraction algorithm (upper part of the image)

Background subtraction is commonly used for detection of moving objects in the frame-by-frame analysis of video streams [17][18][19]. Authors use this method for change detection in the satellite image sets.

Background subtraction method is used for detecting changes on the Northern Poland shore areas. Satellite images from a 7-month time period from each subset are used as a frame set (input) for the analysis. The first image is used both as current image (F_i) and first

averaged background (B_i). Using formula for detection of changed pixels (Fig. 5.) pixels are divided into two groups, changed pixels (foreground) and unchanged pixels (background).

Pixel is marked as changed when the value difference between current image pixel (F_i) and background pixel is larger than 85 percent of the max pixel value. After each image processing current averaged background is calculated using background formula in Fig. 5. Each analysed area pixel value is determined by adding 5 percent of the corresponding foreground pixel value and 95 percent of previously calculated background. An example of detected changed pixels is shown on Fig. 6. The larger the image set is, the background becomes better averaged, and change detection pixel set includes less false detections.

background averaging:

$$B_i(x,y) = \alpha \times F_{i-1}(x,y) + (1 - \alpha) \times B_{i-1}(x,y)$$

detection of changed pixels:

$$Fg_i(x,y) = \begin{cases} 1, & |F_i(x,y) - B_i(x,y)| > A \\ 0, & |F_i(x,y) - B_i(x,y)| < A \end{cases}$$

where:

$B_i(x,y)$ - *current averaged background pixel with position (x,y),*

B_{i-1} - *averaged background in previous step,*

Fg_i - *detected changes in current frame,*

F_i - *currently analyzed frame,*

F_{i-1} - *previously analyzed frame,*

α - *averaging coefficient (5%),*

A - *threshold (depends on raster data, 85% of max value).*
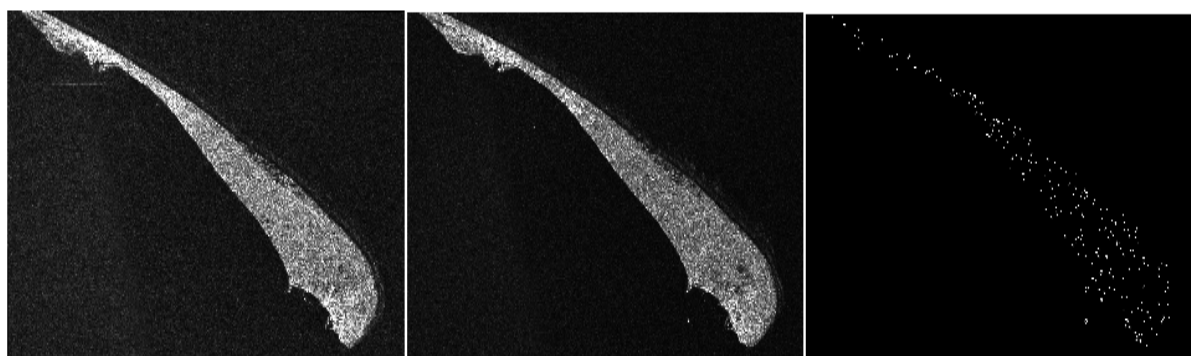
Fig. 5. Background subtraction formulas

Fig. 6. Changed pixels example (Hel Peninsula), 10.02.2017-06.03.2017

Sentinel-1 Level-1 Ground Range Detected data obtained from ESA is ingested into the system in the form of greyscale georeferenced GeoTiff files. Each pixel in the imagery represents a value of surface reflectance. To simplify processing in the proposed algorithm data is formed into the data-cube using Rasdaman. The data-cube is a three-dimensional array of 8-bit (char) types of size x, y and z:

create type SentinelCube as char mdarray [x, y, z]

To trim imagery of only the selected coastal zone, the following rasql query is used:

rasql -q 'SELECT encode(s[lat_min : lat_max, lon_min,: lon_max, *:*], "tiff") FROM SentinelCube AS s' --out file

For the purpose of background averaging, system requests values of pixels at a position (x,y) in the entire time-series:

rasql -q 'SELECT encode(s[x, y, *:*], "tiff") FROM SentinelCube AS s' --out csv
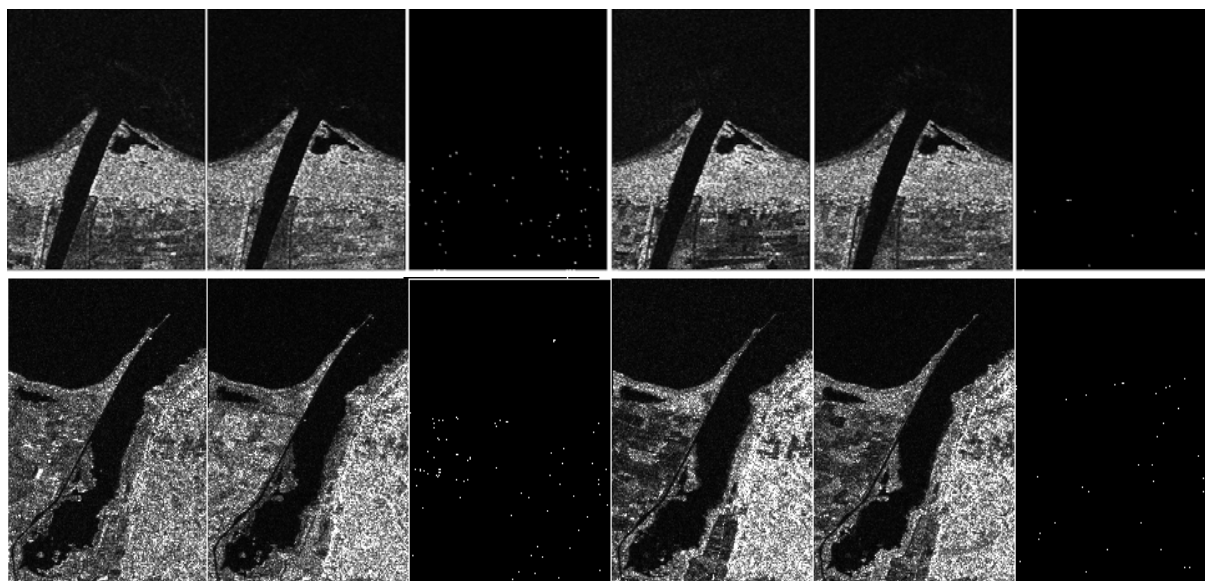
4.3. Analysis results
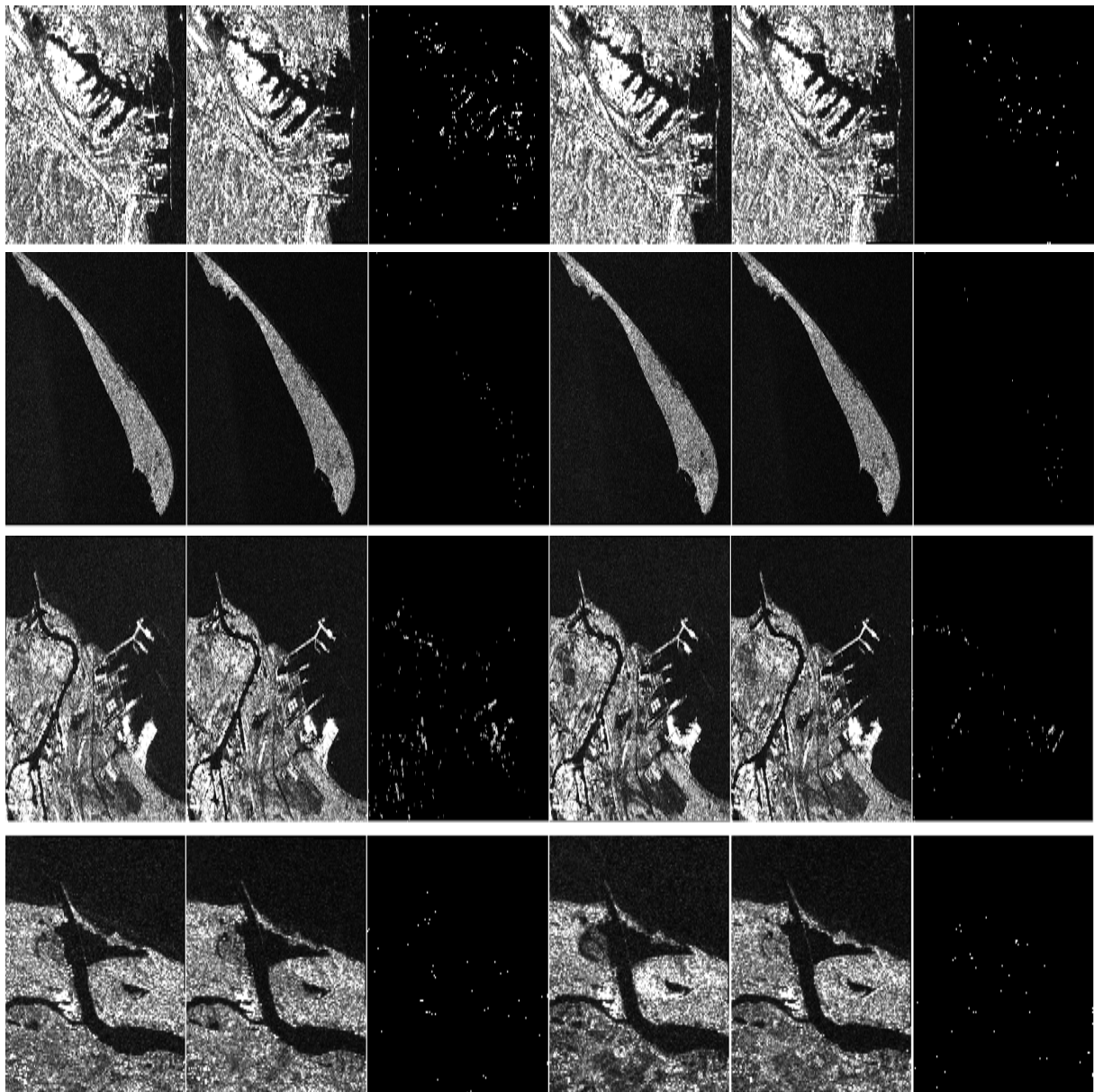
Fig. 7. Change detection (13.10.2016-6.11.2016) and last detection (11.04.2017-05.05.2017) from the algorithm sequence in: Port of Gdynia, Hel Peninsula, Port of Gdansk, Vistula Branch near Gdansk Stogi, Vistula mouth, Bay of Elblag

Algorithm threshold and coefficient (Fig. 5.) had to be calibrated to SAR images. Two background subtraction methods were used, with and without background averaging. The second type of algorithm implementation improved the detection significantly. Changed pixels mask, after a few images, began to include pixels with the greatest change in value. Change detection after the first two images (or during the whole analysis without background averaging) presented many pixels with small change of value between the two image acquisition dates. It can be observed on Fig. 7. when comparing detected changed pixels between the first and second image pair on the same area. Detections are concentrated on the seashore, riverbanks and marine infrastructure; and also include land areas with considerable difference of pixel value, passing ships were also detected.

5. Conclusions

Analysis of satellite data applied in Earth Observation provides a wide range of problems. This is especially true in the context of large-scale satellite coverages over a long period of time, which require vast amounts of storage space and computing power. Therefore, it is essential for computer science to create specialized solutions for satellite imagery analytics.

Recent research in the area of Multidimensional Geographical Information Systems allowed for the development of tools that simplify processing of MDD. Array DBMS, like Rasdaman, provide array syntax along with spatial and temporal operators. Satellite imagery can be manipulated and stored in multidimensional GIS, and it can provide analytics tools for complex algorithms and fusion with data from other sources.

In the article, authors presented application of Multidimensional GIS in coastal zone change detection using Sentinel-1 imagery. As shown at the end of the paragraph 4.2, usage of Rasdaman to form data into data-cube allows for easy access to the entire time-series. Array algebra offered by query language allows for advanced and efficient data manipulation on large scale imagery that is easy to use. Moreover, from the analytic point of view, there is no need to operate on (open, read, close) and track multiple files in time series during

processing. One is only required to specify rasql queries. Overall, proposed solution allowed for faster data selection, simpler data trimming and in general, a simplified algorithm and its application

References

[1]   K. Drypczewski, A. Stepnowski, SOLAP GIS in maritime research, Hydroacoustics pp. 93-100, vol. 19, 2016, Gdansk.

[2]   K. Drypczewski, B. Wiśniewski, M. Kulawiak, K. Bruniecki, GIS for processing multidimensional marine data in SAAS model, Hydroacoustics pp. 47-58, vol. 16, 2013, Gdansk.

[3]   P. Snoeij, E.Attema, M.Davidson, The Sentinel-1 radar mission: status and performance, Radar Conference - Surveillance for a Safer World, 2009.

[4]   A. Rucci, A. Ferretti, A. Monti Guarnieri, F. Rocca, Sentinel 1 SAR interferometry applications: The outlook for sub millimeter measurements, Remote Sensing of Environment Volume 120, p. 156-163, 2012.

[5]   S-1 Polarimetry, https://sentinel.esa.int/web/sentinel/user-guides/sentinel-1-sar/product-overview/polarimetry.

[6]   P. Baumann, A. Dehmel, M. Höfner, N. Widmann, Exploring Sensor and Simulation Raw Data with RasDaMan. Extending Database Technology (EDBT) 2000, March 27-31, 2000, Konstanz, Germany.

[7]   L. Libkin, R. Machlin, L. Wong, A query language for multidimensional arrays: Design, implementation, and optimization techniques, Proceedings of the ACM SIGMOD'96, 1996, Montreal, Canada, pp. 228 - 239.

[8]   A. Marathe, K. Salem: Query Processing Techniques for Arrays, Proceedings of the ACM SIGMOD'99, 1999, Philadelphia, USA, pp. 323-334.

[9]   S. Sarawagi, M. Stonebraker: Efficient Organization of Large Multidimensional Arrays, Proceedings of the ICDE'94, 1994 Houston, USA, pp. 328-336.

[10] P. Baumann, A. Dehmel, P. Furtado, R. Ritsch, N. Widmann, Spatio-Temporal Retrieval with RasDaMan. Very Large Data Bases (VLDB) 1999, September 7-10, 1999, Edinburgh, Scotland, UK, Morgan Kaufmann 1999, pp. 746-749.

[11] Rasdaman website, http://www.rasdaman.org/

[12] P. Baumann, Query Language Guide, http:// rasdaman.eecs.jacobs-university.de/trac/rasdaman/browser/manuals_and_examples/manuals/pdf/ql-guide.pdf.

[13] International Organization for Standardization (ISO): Database Language SQL. ISO 9075, 1992(E), 1992.

[14] Copernicus   overview, http://www.esa.int/Our_Activities/Observing_the_Earth/Copernicus/Overview4

[15] S-1   acquisition,   https://sentinel.esa.int/web/sentinel/user-guides/sentinel-1-sar/acquisition-modes.

[16] Copernicus Open Access Hub, https://scihub.copernicus.eu/.

[17] M. Piccardi, Background subtraction techniques: a review, IEEE International Conference on Systems, Man and Cybernetics, 2004.

[18] Dar-Shyang Lee, Effective Gaussian mixture learning for video background subtraction, IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 27, Issue 5, 2005.

[19] O.Barnich, M. Van Droogenbroeck, ViBe: A Universal Background Subtraction Algorithm for Video Sequences, IEEE Transactions on Image Processing, Volume 20, Issue 6, 2011.