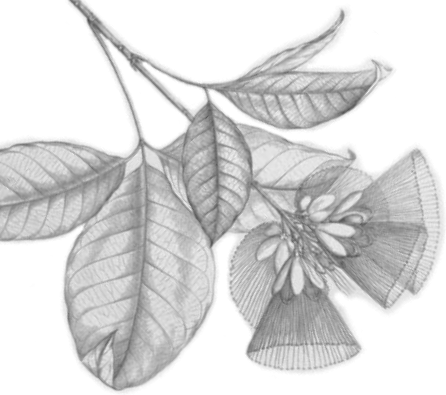


Improvement of Methods and Techniques that Subsidize the Analysis of Land Use Changes Over Time[1]

Natasha Sophie Pereira [2]
Nilton Correia da Silva [3]
Osmar Abílio de Carvalho Júnior [4]

ABSTRACT

Both natural evolution and human activities result in changes in natural resources and in the terrestrial environment. Observation of the temporal and spatial level of the Earth allows us to understand the interrelationship of the phenomena that cause these changes. The studies carried out in this sense vary according to the need of the researchers and the object to be studied. There is a need to work with a large amount of time in a temporal series of multispectral images in order to obtain more detailed data of the space changes over time. This work aims to contribute to the area of geoprocessing regarding the improvement of methods and techniques that subsidize the analysis of land use changes over time. For this purpose, an approach was proposed in order to allow the analysis of an image with k-bands, in n-times simultaneously, so the time dimension was added to the model of a common three-dimensional digital image (lines, columns and bands), making it four-dimensional (lines, columns, bands and times). New multispectral and multitemporal data storage formats (TBSQ, TBIL, TBIP) were produced, to test them was developed a two-stage prototype. The first version of the prototype consumed a considerable time to execute the tests, leading to the need for improving the prototype. In the second version of the prototype, a multi-pointer technique was implemented, which brought significant performance gains to the initial version.

Keywords: Multidimensional; Temporal-Spectrum Images; Multitemporal; TBSQ, TBIL, and TBIP; Multi-Pointers.

[1] Trabalho elaborado pelos autores a partir de: Pereira NS 2014. Novos Formatos de Dados e Algoritmos para Suporte à Imagens Multi-Dimensionais. Dissertação (Mestrado em Ciências Ambientais), Centro Universitário de Anápolis, Goiás.

[2] Doutorado em andamento em Geografia pela Universidade de Brasília, UnB, Brasil.; Mestrado em Sociedade, Tecnologia e Meio Ambiente pelo Centro Universitário de Anápolis, UniEVANGÉLICA, Brasil. Docente no Centro Universitário de Anápolis, UniEVANGÉLICA, Brasil. natasha.sophie@gmail.com

[3] Doutorado em Geologia pela Universidade de Brasília, UnB, Brasil. Docente na Universidade de Brasília, UnB, Brasil. niltoncs@unb.br

[4] Doutorado em Geologia pela Universidade de Brasília, UnB, Brasil. Docente na Universidade de Brasília, UnB, Brasil. osmarjr@unb.br

Natural resources and the terrestrial environment are always undergoing changes, both by the natural evolution and for human activities over them. The complex interrelationship of the phenomena that cause these changes must be understood by observing a wide range of temporal and spatial scales. According to Shen-Em Qian (2013), the observation of these phenomena can be carried from space, with the aid of sensors coupled in artificial satellites, in this case, the observation brings a considerable amount of information that can aid in the monitoring and preservation of the environment, including the determination of trends (Qian 2013; Schowengerdt 2007).

Remote sensing satellites launched into orbit from the 1970s (Qian 2013), using electronic sensors, acquire data from the earth's surface in the form of electrical signals corresponding to the spectral energy variation of the target. After-acquired data are stored and later converted into digital images (Lillesand, Kiefer, and Chipman 2004; Qian 2013). A digital image is a two-dimensional array of pixels that contains discrete values. If digital images from remote sensors, row and column indexes (x, y) identify a point in the image (analogous to a quadrant of the total area captured by the sensor) and the corresponding value of the array element identifies the mean value of reflectance from targets present in the electronically measured quadrant. The mean value of reflectance presented by a pixel is named Digital Number (DN), also known as grayscale, being a value between 0 and 255 (Lillesand, Kiefer, and Chipman 2004; Gonzalez and Woods 2011; Montabone 2010; Petrou and Petrou 2010).

Digital image processing is one which has both the input and output as images, involving extraction of patterns or recognizing individual objects in the studied images. Digital image processing methods are applied with the purpose of improving visual information, aiming to aid both the human interpretation and the automatic perception thru machines. A remote sensing image can be represented by a three-dimensional cube that expresses the integration between spacial (x,y) and spectral (λ) dimensions, i.e., DN values in each spectral band collected. Thus, a pixel in a remote sensing digital image denotes the spectral signature of the represented target (Schowengerdt 2007; Qian 2013; Petrou and Petrou 2010; Gonzalez and Woods 2011).

Time is continuous by its nature, yet, to be computationally represented a discrete approach is necessary, in which the temporal variation can be represented by a timeline specified according to the frequency desirable (Dias, Câmara, and Davis Jr. 2005). The Earth's surface continuous observation allows the comprehension of the environmental and natural resources changes, and it is necessary to

analyze this environment not only in a single time and space but to multiply these quantities (Qian 2013; Schowengerdt 2007; Sausen 2012). In this way, literature brings various articles that aim to analyze the phenomena occurred in a space monitored by an artificial satellite in n distinct times.

Researchers are publishing articles around the use of technologies to analyze changes that occurred over time in a specific studied area. These articles present techniques to compare an initial image to a second image captured later in time (Carvalho Júnior et al. 2011; Carvalho Júnior and Silva 2007; Silva et al. 2012). Other authors seek to analyze the land use changes through images classification in distinct times and subsequent detection of the changes that occurred between them. In this case, many technics can be applied according to the need of the researchers and the object to be studied (Menke et al. 2009; Couto, Souza Filho, and Hayakawa 2011; Carvalho Júnior et al. 2005; Gomes et al. 2011; Oliveira 2013; Carvalho et al. 2002; Ioannis and Miliadis 2011; Kampel, Amaral, and Soares 2005; Lorena, Santos, and Shimabukuro 2004; Lorena et al. 2001; Scheer and Rocha 2006; Shaikh, Gotoh, and Tachiiri 2005; Carreira et al. 2012; Diedrich and Narvaes 2013; Gurmessa, Nemomissa, and Tadesse 2013; Mata et al. 2007; Moreira, Adami, and Rudorff 2004; Van de Voorde, Demarchi, and Canters 2009; Vieira and Biondi 2008; Zhou, Troy, and Grove 2008).

Studies can vary according to the researchers need and the studied object, but due to software limitation it was not possible to process all the bands of a specific image across a big amount of time thereby limiting researchers ability to study the spectral sign through time in order to obtain a more detailed information about the changes in land covering (Carvalho Júnior, Hermuche, and Guimarães 2006; Santana et al. 2010; Carvalho Júnior et al. 2008; Couto Júnior et al. 2007; Carvalho Júnior et al. 2009).

Based on that, this work aims to contribute to the field of geoprocessing regarding the improvement of methods and techniques that subsidize the analysis of land use changes over time. For this purpose, an approach was proposed where it is possible to analyze an image with k-bands, in n-times simultaneously, so that the common three-dimensional digital image (lines, columns and bands) can be analyzed considering the time, passing it then to be four-dimensional (lines, columns, bands and times). Supporting this objective, the known data storage formats which is able to process only three-dimensional images, i.e. Band SeQuential (BSQ), Band Interleaved by Line (BIL) and Band Interleaved by Pixel (BIP), were rewritten adding the time dimension.

To test these new data storage formats proposed, a software prototype was developed in two stages. In the first version of the prototype, the tests were executed in a considerable amount of time, which led to the need to improve the efficiency of the implemented algorithms. For the second version of the prototype, then multi-pointer techniques were implemented, which brought significant performance gains to the initial version of it.

There is a wide possibility of applying this study, since the simple recognition of vegetation in a specific study area, until the prediction of future changes in the environment, based on the pattern of previous change graph in the area. Multitemporal study of a specific area allows the recognition of changes patterns in land use, and how this use evolved over time.

NEW DATA FORMATS FOR REMOTE SENSING IMAGES

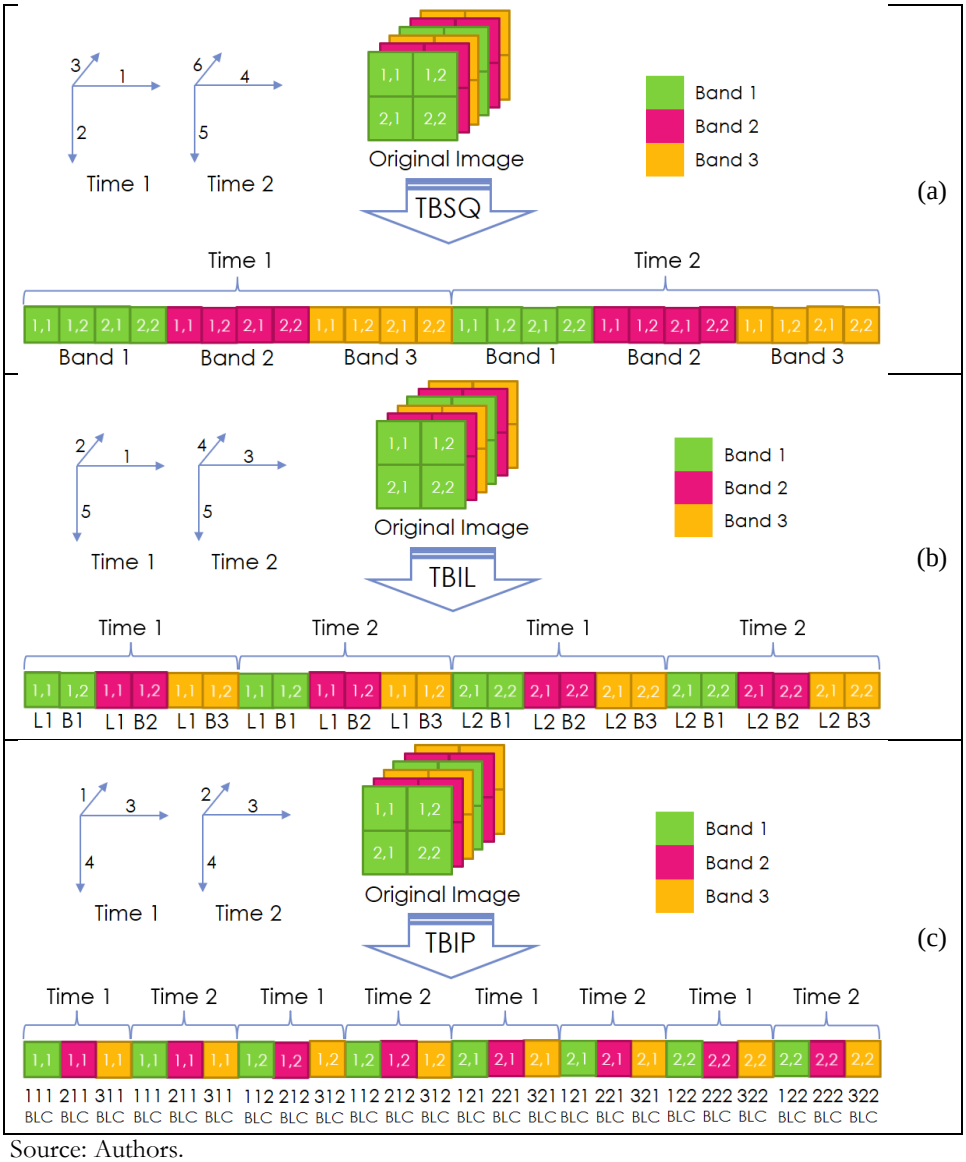
Multi-bands images are stored in disc or tape and are displayed through a combination between the space dimension (lines and columns) and bands. Such storage can be presented in three formats: BSQ, BIL, and BIP (Japan Association of Remote Sensing 1993; QIN, Chao, and Kim 2010; Schowengerdt 2007). Different formats have distinct advantages in terms of image processing operations and interactive analysis. For instance, the BIP and BSQ formats allow an excellent performance to access respectively spectral and spatial hyperspectral data, in that way the BSQ format is faster to access the full image; and the BIL format brings a balance between the spatial and spectral analysis. However, all the three storage data types are convertible to each other (QIN, Chao, and Kim 2010; Schowengerdt 2007). To manipulate remote sensing digital images adding temporal dimension (times) was developed new data storage formats, i.e. Time and Band SeQuential (TBSQ), Time and Band Interleaved by Line (TBIL), and Time and Band Interleaved by Pixel (TBIP).

Figure 1 exemplifies how an image with a spatial dimension of 2 lines by 2 columns, spectral dimension of 3 bands, and time dimension of 2 times, is stored in disc in each of the proposed three new digital storage formats: TBSQ (a), TBIL (b), and TBIP (c). The flow shown in arrows on the left side of each data type enables to realize the storage sequence of pixels.

Data stored in TBSQ format, Sub-Figure 1(a), has all the pixels of a line in a band, stored in sequence, then all pixels in the next line and the process repeats until the end of the band involved, starting then the next band until all the time be stored, repeating the process for the next time until the entire image is stored. The data stored in TBIL format, Sub-Figure 1(b), the pixels are arranged, all the pixels of one line in one band are stored in sequence, accordingly pixels are in the same line in the next

band, and the process is repeated until the last band of the image, then all pixels in the same line of the first band in the next time will be stored, and the process repeats until the same line in all the times is stored, starting a new cycle for the next line until the entire image is stored. In the TBIP storage format, Sub-Figure 1(c), the first pixel of the first line of the first band is stored, then the first pixel of the first line of the next band, the process is repeated until the last band, and then the first pixel of the first line of the first band of the next time is stored and the process repeats until the first pixel in all bands of all times are stored, passing to the second pixel of the same line, repeating the process for all pixels of that line, restarting on the first pixel of the next line, the process is repeated until the end of image.

Figure 1. The new TBSQ, TBIL e TBIP Data Storage Formats.



Source: Authors.

As seen in Figure 1, the position of one pixel located in such four-dimensional space (line, column, band, and time) varies according to its storage format. The following function $f: Z^4 \rightarrow Z$ maps the memory address of a random pixel in a TBSQ image stored in disc:

$$f(l,c,b,t) = \big((t \times qtb \times qtl \times qtc) + (b \times qtl \times qtc) + (l \times qtc) + c\big) \times tp \quad (1)$$

where: l = the pixel line, c = the pixel column, b = the pixel band, t = the pixel time, qtb = the amount of bands in the image, qtl = the amount of lines in the image, qtc = the amount of columns in the image and tp = the pixel size in bits.

The function $g: Z^4 \rightarrow Z$ maps the memory address of a random pixel in a TBIL image stored in disc:

$$g(l,c,b,t) = \big((l \times qtb \times qtc \times qtt) + (t \times qtb \times qtc) + (b \times qtc) + c\big) \times tp \quad (2)$$

where: l = the pixel line, c = the pixel column, b = the pixel band, t = the pixel time, qtb = the amount of bands in the image, qtc = the amount of columns in the image, qtt = the amount of times in the image, and tp = the pixel size in bits.

The function $h: Z^4 \rightarrow Z$ maps the memory address of a random pixel in a TBIP image stored in disc:

$$h(l,c,b,t) = \big((l \times qtb \times qtc \times qtt) + (c \times qtb \times qtt) + (t \times qtb) + b\big) \times tp \quad (3)$$

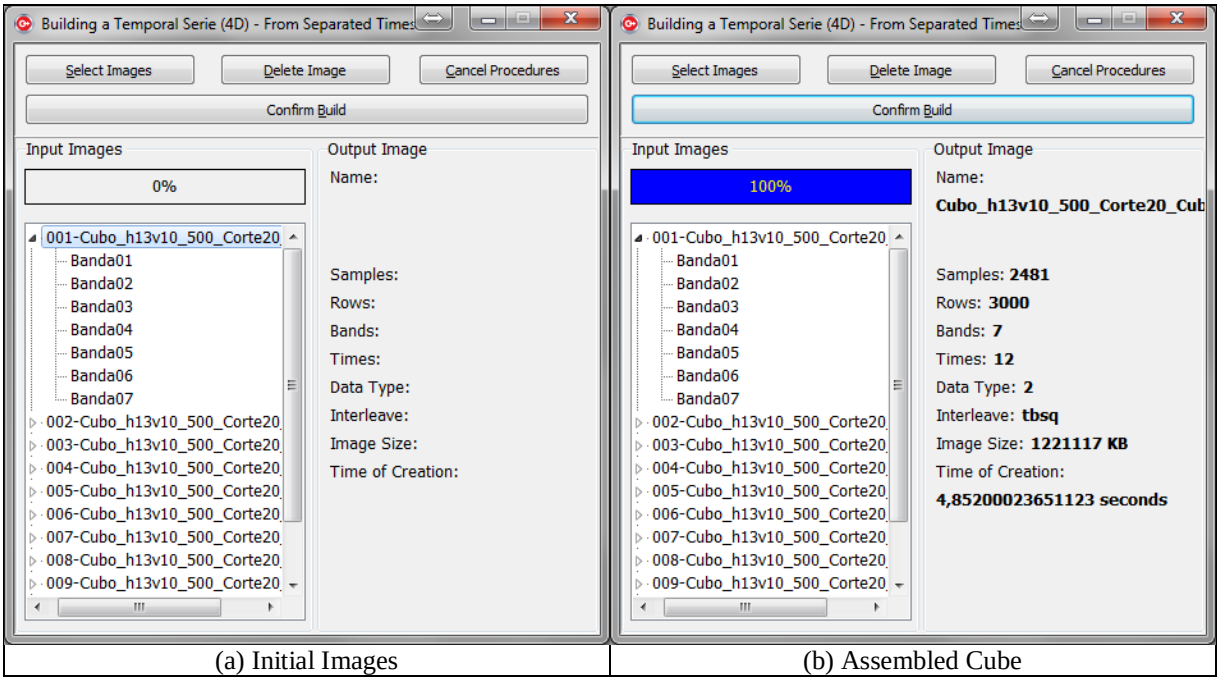
where: l = the pixel line, c = the pixel column, b = the pixel band, t = the pixel time, qtb = the amount of bands in the image, qtc = the amount of columns in the image, qtt = the amount of times in the image, and tp = the pixel size in bits.

ASSEMBLING THE FOUR-DIMENSIONAL CUBE

Figure 2 presents the interfaces for assembling the temporal cube in the prototype. The Sub-Figure 2(a) brings the initial assembly phase when solely the original images were selected through the "Select Images" button. On the other hand, the Sub-Figure 2(b) displays the following stage, after the temporal series assembly through the "Confirm Build" button, when the information attaining the temporal cube are displayed on the interface, and the temporal cube, and its header, are already assembled and filed in the disc.

The prototype in focus assembles temporal series in the TBSQ format from "multi-times" or "multi-bands" images, i.e., the original images may be stored so that 1 time encompasses k-bands ("multi-times") or that 1 band encompasses n-times ("multi-bands").

Figure 2. Prototype: Four-Dimensional Cube Assembly.

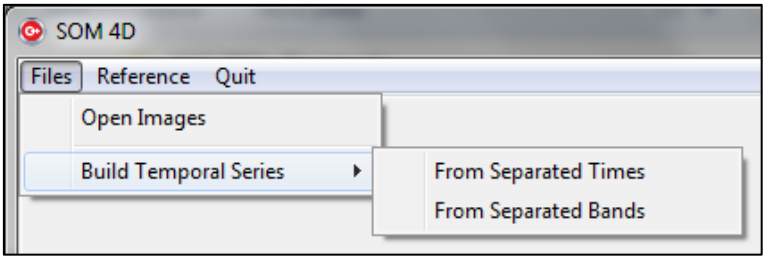


(a) Initial Images | (b) Assembled Cube

Source: Authors.

The interfaces for the selection of origin images for the four-dimensional cube assembly (Figure 2) are similar in the assembly process of both multi-bands and multi-times images. It is possible to select the entry image type (Figure 3), from which the final temporal series, in the TBSQ format, is assembled.

Figure 3. Prototype: Selecting the Entry Images for Assembly the 4D Cube.



Source: Authors.

MULTI-TIMES IMAGES (FROM SEPARATED TIMES):

Sub-Figure 4(a) displays a diagram that presents the assembly method of a TBSQ temporal series from a multi-times image, in which the initial images are the times of a remote sensing image and each time contains k-bands.

The process is carried out so that one time is entirely copied (lines, columns, and bands) to the final file (4D cube), and the process repeats itself for the remaining times of the image, on the order informed by the user upon selecting them. In Sub-Figure 4(a), each color represents a distinct time that contains k-bands.

MULTI-BANDS IMAGES (FROM SEPARATED BANDS):

Sub-Figure 4(b) displays a diagram that represents the assembly method of a TBSQ temporal series from a multi-band image, in which the initial images are the bands of a remote sensing image and each band contains n-times.

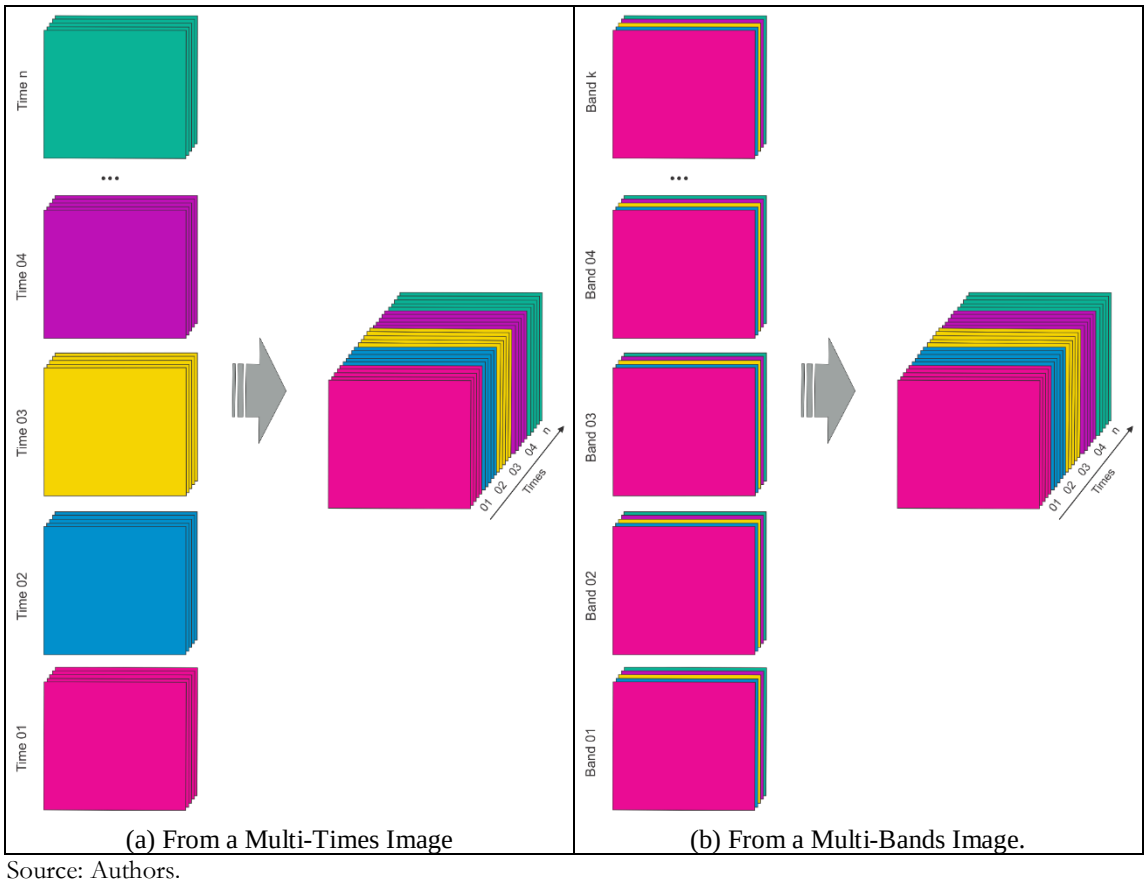
The process is carried out so that one time (lines and columns) of each band is copied to the final file (4D cube), interspersing same times in distinct bands. This process repeats itself for the number of times the image has. In Sub-Figure 4(b), each color represents a distinct time in one of the k-bands that will constitute the final image. Similar colors indicate similar times in distinct bands.

In Figure 4, in both sub-figures, the final assembled cube possesses 4 dimensions, being 2 spatial dimensions (lines and columns), one spectral dimension (bands), and one temporal dimension (times). In either case, the four-dimensional cube, assembled as a result of the initial image processing, is in the TBSQ format. Being so, initial images with similar spatial, spectral and temporal dimensions will result in the same final 4D image, despite the fact that the formation process of the 4D cube is different.

TEMPORAL SPECTRUM

Each and every object on the Earth's surface reflects, absorbs and transmits electromagnetic radiation proportionally in each wavelength, according to its chemical and biophysical properties. Considering an artificial satellite, that has many bands, each one responsible for capturing a specific length of the spectrum, it is possible to obtain the spectral signature of a target through observation of its reflectance in that channel (Florenzano 2011).

Figure 4. Scheme for Assembling the four-Dimensional Cube



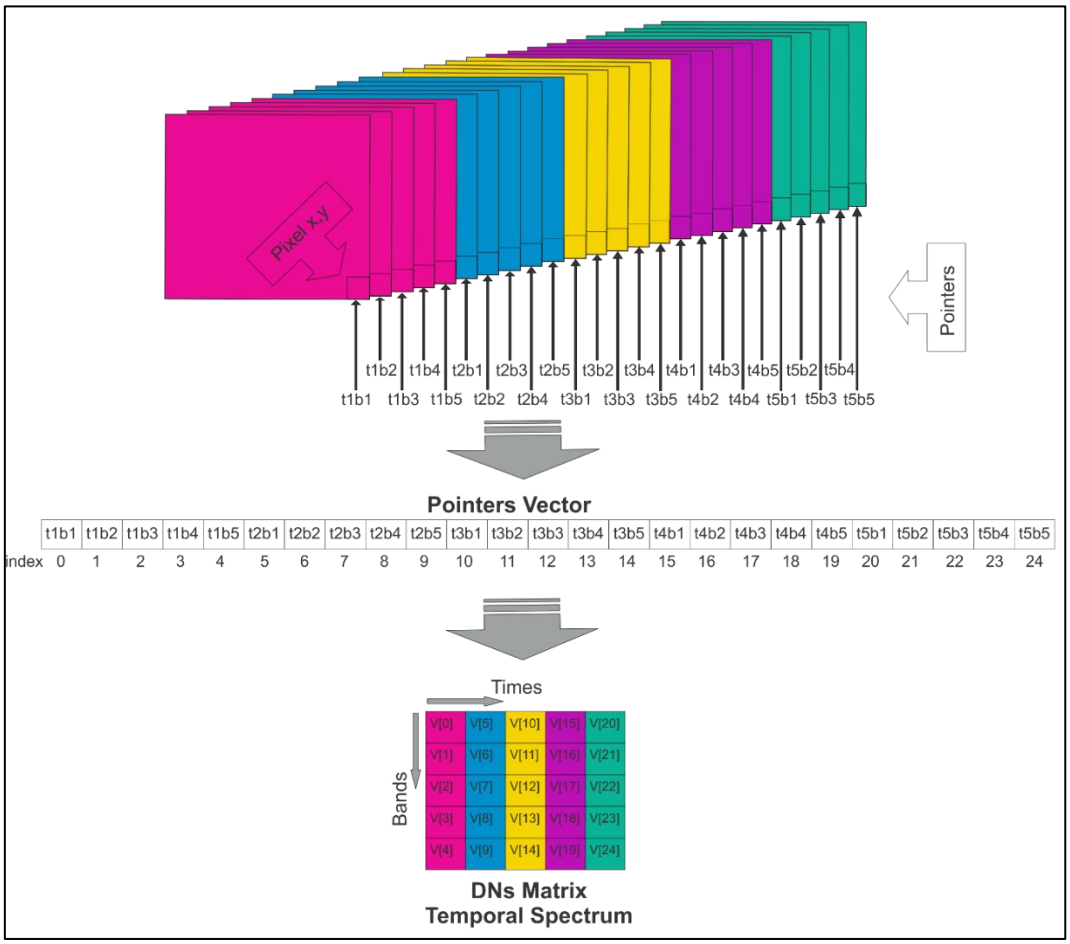
(a) From a Multi-Times Image (b) From a Multi-Bands Image.

Source: Authors.

Each pixel of an image possesses a particular spectral signature, being it possible to observe pixels with similar or extremely distinct signatures. Considering each time in the image possesses the same set of bands, the spectral signature of the same pixel may vary over time, what enacts the need for further studies of such variation. Hence, this treatise proposes a method of capitation of a pixel's spectrum along time, forming a temporal spectrum matrix of such pixel. Figure 5 displays a diagram containing the operation of this method for a random pixel.

The method presented in Figure 5 utilizes a multi-pointers computational technique. From the spatial positions (lines and columns) of a selected pixel, a pointer is positioned in each band of each time of the image on the same spatial position, and the address of this pointer is stored in a pointer vector. The pointers are then utilized to collect the values of Digital Number in their position, hence filling the DN values matrix. The filling of this DNs matrix is always carried out in the indexes according to the band and time where the pointer is placed.

Figure 5. Collecting the Temporal Spectrum Utilizing Multi-Pointers Technology.



Source: Authors.

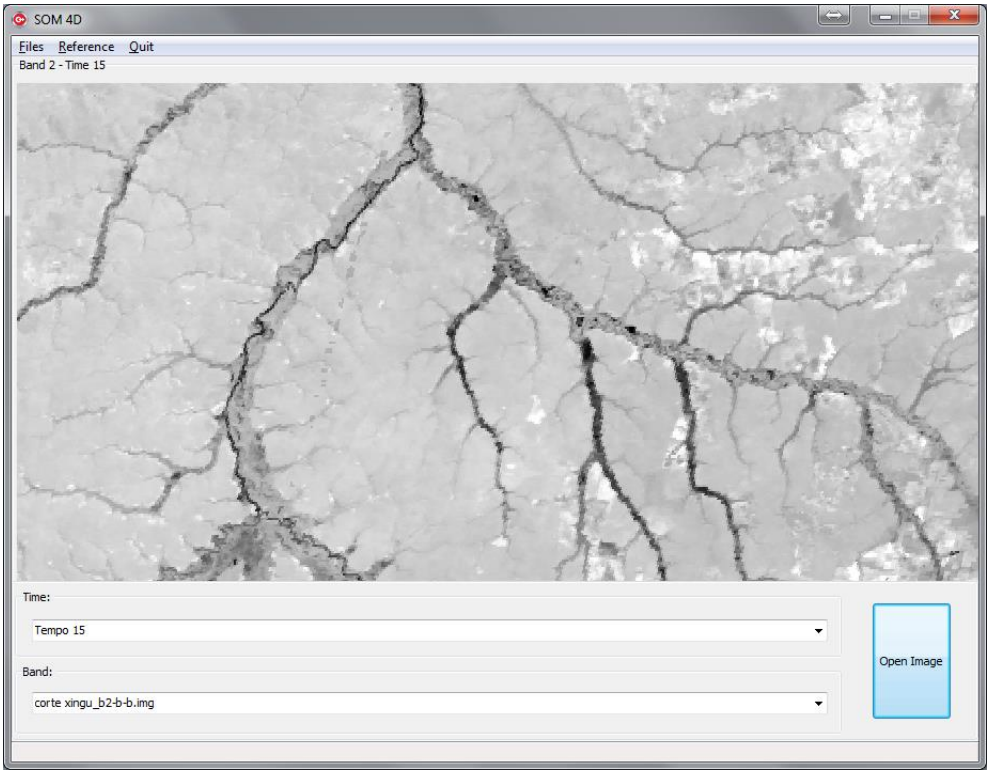
This method enables the consecutive collection of DN values in all bands and times of the image, regardless the need of dislocating the collector through all the spatial dimension (lines and columns) of each band in every new DN collection, rendering the process more efficient. The multi-pointers method was utilized in two stages of the prototype, in the temporal cube upload phase, for its visualization and study, and in the phase where the temporal spectrum characteristics of a single pixel are collected for its manipulation.

During the temporal cube upload phase, it is necessary to verify the values with the higher and lower DNs, that will be used in the assembly of the temporal spectrum graph. Such verification is arranged by utilizing the collection method of the temporal spectrum data of 20% of the pixels in the image, being individually collected and compared in pairs. It was needed to define this limit of 20% to collect the pixels due to the computational constraint. If the algorithm to find the minimum and maximum DN values in the image were to evaluate all DNs, the final reply would take a long time. This

sample is sufficient to infer the minimum and maximum DN values because the values repeat through the pixels along the whole image. At the end of all comparisons, the values of higher and lower DNs are defined.

Figure 6 displays the prototype interface, presenting the image from one chosen band and time of an already loaded temporal cube. The prototype allows the interaction between the user and the loaded face of the cube so that in every selected pixel a Temporal Spectrum graph (Figure 7) is instantaneously created, providing a visualization of the changes in that pixel over time in all image bands.

Figure 6. Loaded 4D Cube.

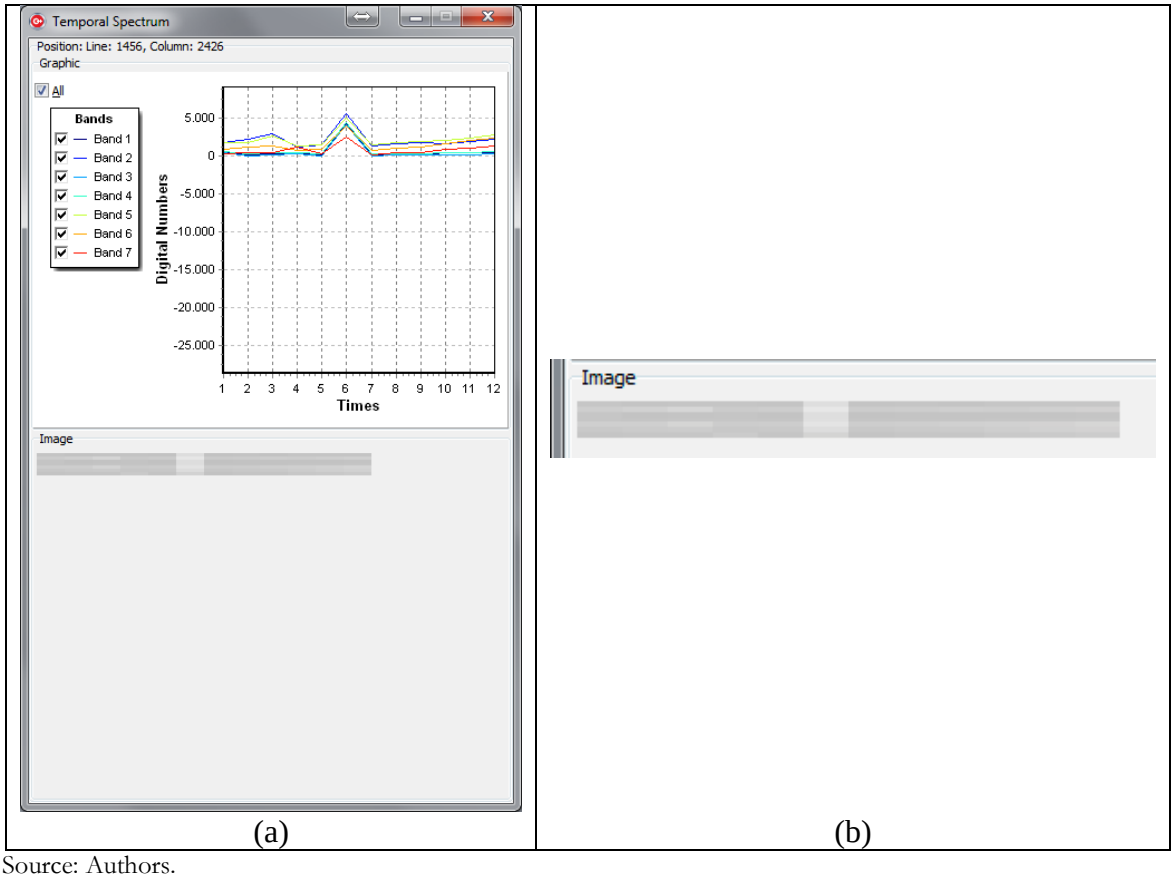


Source: Authors.

In the phase which presents the temporal spectrum graph of a selected pixel, the method is utilized to collect the values of DN relative to this pixel's spectrum over time. Figure 7 presents the prototype interface that provides the visualization of the spectrum over time from a selected pixel. The preview is arranged in separate bands, allowing for the visualization of changes in the values presumed by the pixel throughout time in each of the spectral bands independently.

In Sub-Figure 7(a), the ordinate axis of the temporal spectrum graph is composed by a range of values among the lower and higher DN values, collected in the 4D cube upload phase. The abscissa axis of the same graph brings the temporal dimension of the image, in the example displayed in Figure 7, an image with 12 times. The spectral dimension is presented by colors, in which each color represents one band in the image, and they can be distinguished in the subtitle presented beside the graph, where one can select which bands should be presented in the graph, facilitating the analysis. The visualization form of the temporal spectrum brings, in addition, a figure in a grayscale, detailed in Sub-Figure 7(b), where one can visually verify the shifts in DN values in each band, so that same tonalities represent the same DN value.

Figure 7. The Temporal Spectrum Graph.



Source: Authors.

RESULTS AND DISCUSSION

In order to accomplish testing, a computer with the following configuration was utilized: Operational system Windows® 7 Ultimate 64 Bits, Intel(R) Core(TM) i7-3770 CPU @ 3.40GHz 3.90GHz processor, 8GB RAM memory. The prototype performance test was done on a Western

Digital Corporation® HD (Hard Disk) model WD10EARS-00Y5B1 (930261 cylinders, 16 heads, 63 sectors per tracks, 512 bytes per sector), SATA-II with 1TB in storage capacity, and also on a Kingston® SSD (Solid-State Drive) model SV300s37A480G (1938021 cylinders, 16 heads, 63 sectors per tracks, 512 bytes per sector), SATA-II with 470GB in storage capacity. Auxiliary external HDs stored the resulting products, as well as the original images and cuts.

The original image was provided by the Laboratory of Spatial Information Systems (LSIE - Laboratório de Sistemas de Informações Espaciais) from the Department of Geography (GEA - Departamento de Geografia) at the University of Brasília (UnB - Universidade de Brasília). Initially stored in the multi-band format, and 242,58GB total disk size, it has a spatial dimension of 5000 lines by 6296 columns, spectral dimension of 7 bands, a temporal dimension of 591 times, and pixels with 16 bits in length. The ENVI (Environment for Visualizing Images) software (Harris Geospatial Solutions 2018) was used for the temporal and spatial cut of the image and the separation of the temporal and spectral dimensions, in order to obtain suitable materials for assembling images in a multi-times format.

To demonstrate the efficiency and effectiveness of the proposed methods, tests were made with images in two formats, multi-times, and multi-bands. The spectral dimension of the images was set at 7 bands for all tests. As the image's spatial dimension (lines×columns) the 3000×2481 and 5000×6296 sizes were defined. The temporal dimension was set in 8 different times. The pixel size was the same (16 bits) for all samples.

In the original images were made temporal cuts in order to determine the 8 different times ranging from 2 (two) to 400 (four hundred). Table 1(a) presents a description of the cuts made in the image for testing the performance of the prototype images in a multi-bands format. Table 1(b) shows the image cuts prepared for testing in multi-times format, in which, due to hardware constrictions, it was not possible to work with a temporal dimension over 350 times.

ASSEMBLING THE FOUR-DIMENSIONAL TEMPORAL CUBES

The four-dimensional temporal cubes (4D cube) were assembled from the samples specified in Table 1. A benchmarking was performed in order to compare the performance in creating the 4D cube, both in tests made on an HD on the SSD.

Figure 8 and Figure 9 show the performance graphs of the prototype during the 4D cube assembly from multi-bands images (Figure 8) and from multi-times images (Figure 9), furthermore, it presents the tested performance on HD (Sub-Figure 8(a) and Sub-Figure 9(a)) and on SSD (Sub-Figure

8(b) and Sub-Figure 9(b)). The images used for testing had a spatial dimension of 3000×2481 and 5000×6296.

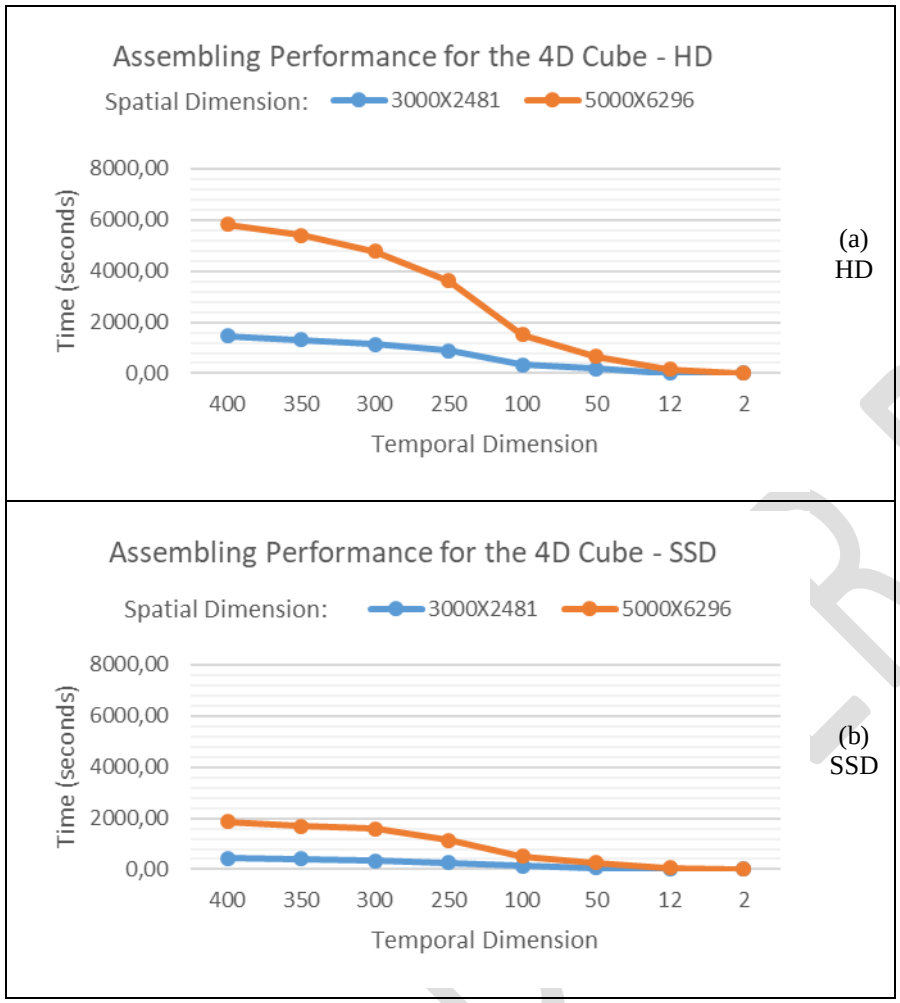
Table 1. Image Samples for Testing.

Multi-Bands Format (a)						
Sample	**Spatial Dimension**		**Spectral Dimension**	**Temporal Dimension**	**Pixel Size (bits)**	**Sample Size (GB)**
	Lines	**Columns**				
Sample 01a	3000	2481	7	400	16	38.82
Sample 02a	3000	2481	7	350	16	33.97
Sample 03a	3000	2481	7	300	16	29.11
Sample 04a	3000	2481	7	250	16	24.26
Sample 05a	3000	2481	7	100	16	9.70
Sample 06a	3000	2481	7	50	16	4.85
Sample 07a	3000	2481	7	12	16	1.16
Sample 08a	3000	2481	7	2	16	0.19
Sample 09a	5000	6296	7	400	16	164.18
Sample 10a	5000	6296	7	350	16	143.66
Sample 11a	5000	6296	7	300	16	123.14
Sample 12a	5000	6296	7	250	16	102.61
Sample 13a	5000	6296	7	100	16	41.05
Sample 14a	5000	6296	7	50	16	20.52
Sample 15a	5000	6296	7	12	16	4.93
Sample 16a	5000	6296	7	2	16	0.82
Multi-Times Format (b)						
Sample	**Spatial Dimension**		**Spectral Dimension**	**Temporal Dimension**	**Pixel Size (bits)**	**Sample Size (GB)**
	Lines	Columns				
Sample 01b	3000	2481	7	350	16	33.97
Sample 02b	3000	2481	7	300	16	29.11
Sample 03b	3000	2481	7	250	16	24.26
Sample 04b	3000	2481	7	100	16	9.70
Sample 05b	3000	2481	7	50	16	4.85
Sample 06b	3000	2481	7	12	16	1.16
Sample 07b	3000	2481	7	2	16	0.19
Sample 08b	5000	6296	7	350	16	143.66
Sample 09b	5000	6296	7	300	16	123.14
Sample 10b	5000	6296	7	250	16	102.61
Sample 11b	5000	6296	7	100	16	41.05
Sample 12b	5000	6296	7	50	16	20.52
Sample 13b	5000	6296	7	12	16	4.93
Sample 14b	5000	6296	7	2	16	0.82

Source: Authors.

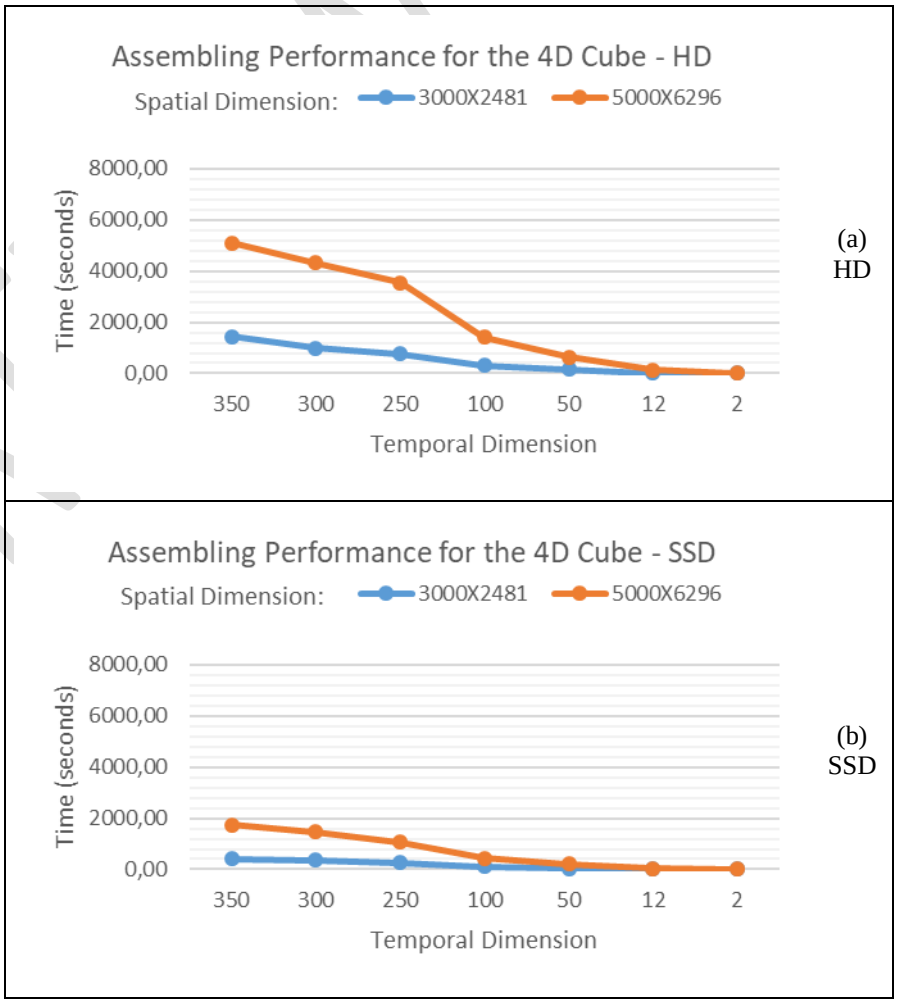
A simple observation of the graphs in Figure 8 and Figure 9 enables the perception that the performance of the prototype is superior when running on an SSD, due to the optimization of the hardware features. One can also see that the processing time spent by the prototype is directly proportional to the image size. When cross-analyzing the graphs with Table 1, it is noteworthy that the higher the spatial and temporal dimensions of the image, the larger the size of the stored image, and, consequently, the longer the time for assembling the 4D cube.

Figure 8. Assembling Performance for the Multi-Bands 4D Cube



Source: Authors.

Figure 9. Assembling Performance for the Multi-Times 4D Cube



Source: Authors.

The samples 01a and 09a have 400 times, only being able of testing originally for multi-bands images, which have seven original files (bands), and each one is composed of 400 layers (times). This limitation in the tests took place due to the hardware, with the settings established for testing, which did not support loading 400 distinct files (times), as it is the case of the configuration of multi-times images, in which each time is composed of 7 layers (bands). Comparing the performance of the prototype in assembling 4D cubes in multi-bands (Figure 8) and multi-times (Figure 9) configurations, one notices the similarity in performance on images of the same size tested on similar hardware configurations. It is important to note that the product (a 4D cube) generated in both cases is the same (Figure 4), that is, a TBSQ cube.

Figure 8 brings the performance of the software during the assembly of four-dimensional cubes from multi-bands images. It was possible to test images up to 400 times. In this case, the software spent 1472.48 seconds to carry out the assembly of a cube of 3000×2481 spatial dimension on an HD, and 425.40 seconds to assemble the same cube on an SSD. On the other hand, the assembly of a cube with 400 times with a spatial dimension of 5000×6296 took 5829.24 seconds on an HD, and 1877.43 seconds on an SSD. The temporal dimension presented in samples 01a and 09a was the largest amount of times tested in the prototype due to hardware limitations, however, utilizing the prototype for images with higher temporal dimension is possible with adequate hardware.

Both Figure 8 and Figure 9 bring tests in samples with a time dimension of 350 times or less. An image with this temporal dimension and a spatial dimension of 3000×2481 was processed by the prototype on an HD in 1313.06 seconds in the case of original multi-bands images, and in 1441.18 seconds in the case of original multi-times images. The difference of approximately 128 seconds is due to the cube assembling technique used in each original type. If the assembly occurs on an SSD, the assembly time of the same cube with original multi-bands images is 410.60 seconds, and for multi-times images, 415.51 seconds, meaning a difference of approximately 5 seconds. When assembling images with a spatial dimension of 5000×6296, under the same conditions, the assembly time is 5829.24 and 1699.76 seconds for multi-bands images, and 5109.95 and 1739.10 seconds to multi-times images, respectively on an HD and an SSD.

The difference in performance in assembling four-dimensional cubes from multi-bands and multi-times images with the same spatial and temporal dimensions in similar circumstances is minimal, not hampering the overall performance of the prototype.

For images with low temporal dimensions, the difference in performance is even more negligible. For images with 2 times, the spatial dimension of 3000×2481, assembled on an HD, the processing time was 0.51 seconds (multi-bands) and 0.50 seconds (multi-times). For a temporal cube assembled on an SSD, the processing time was 0.50 seconds (multi-bands) and 0.47 seconds (multi-times). For a spatial dimension of 5000×6296, the performance of the prototype was 2.21 seconds (multi-bands) and 2.00 seconds (multi-times) for an HD, and 2.17 seconds (multi-bands) and 1.97 seconds (multi-times), when using an SSD.

In samples with a temporal dimension of 12, the software performance was 4.46 (multi-bands) and 3.68 (multi-times) seconds using an HD, and 3.23 (multi-bands) and 2.81 (multi- times) seconds, using an SSD, for images with a spatial dimension of 3000×2481. And for images with a spatial dimension of 5000×6296, the software had a processing time of 154.85 (multi-bands) and 147.90 (multi-times) seconds, using an HD, and 49.50 seconds (multi-bands) and 48.27 seconds (multi-times), using an SSD.

The performance in assembling the temporal cubes in samples 07a, 06b, 08a and 07b, respectively with 2 and 12 as temporal dimensions, shows that images of low spatial and temporal dimensions have a similar processing time in using both, HD and SSD. This happens due to the size of the memory buffer (8192 bytes) set to the assembly phase of the four-dimensional cube, which allows these images to be completely processed by the computer's memory, requiring no swapping[5], thus saving processing time.

Table 2 shows the data on the time processing reduction when using an SSD to assembly the 4D cube for all samples. In it is possible to realize the amount of time reduction in performance from the utilizing the prototype on an SSD, compared to its use on an HD. In the tested images with a higher temporal dimension, i.e. samples 01a and 09a, the time reduction reaches 71.11% and 67.69%, respectively. For images with 350 times, such as samples 02a, 01b, 10a and 08b, the time reduction in the first two is similar to the previous cases, 68.73% and 71.17% respectively, and also in the last two cases, 68.47% and 65.97% respectively.

For medium images, with 250 times, the time reduction reaches 70.82% (sample 04a), 66.79% (sample 03b), 68.50% (sample 12a) and 69.76% (sample 10b). One can see, when analyzing Table 2, that samples with more than 50 times and a spatial dimension of 3000×2481 (samples from 01a to 06a and samples from 01b to 05b) have an average performance time reducing of 68.77%. In the case of

[5] Data exchange between a secondary memory and main memory.

images with a spatial dimension of 5000×6296, the time reduction is higher than 60% for images with more than 12 times (samples from 09a to 15a and samples from 08b to 15b), being the average time reduction in this case 67.23%. Hence, in both cases, the time reduction is always similar in samples with similar characteristics, although in distinct kinds of original images (multi-bands and multi-times).

Table 2. Performance Time Reduction in Assembling the 4D Cube.

Initial Images in the multi-Bands format			
Sample	**Time to Assembly the 4D Cube (seconds)**		**Gain (%)**
	on HD	**on SSD**	
Sample 01a	1472.48	425.40	71.11
Sample 02a	1313.06	410.60	68.73
Sample 03a	1141.38	336.67	70.50
Sample 04a	900.14	262.71	70.82
Sample 05a	334.25	126.27	62.22
Sample 06a	182.83	50.22	72.53
Sample 07a	4.46	3.23	27.62
Sample 08a	0.51	0.50	2.92
Sample 09a	5829.24	1877.43	67.79
Sample 10a	5391.16	1699.76	68.47
Sample 11a	4780.31	1589.85	66.74
Sample 12a	3630.28	1143.68	68.50
Sample 13a	1511.12	512.01	66.12
Sample 14a	672.60	250.76	62.72
Sample 15a	154.85	49.50	68.03
Sample 16a	2.21	2.17	2.08
Initial Images in the multi-Times format			
Sample	**Time to Assembly the 4D Cube (seconds)**		**Gain (%)**
	on HD	**on SSD**	
Sample 01b	1441.18	415.51	71.17
Sample 02b	982.82	364.23	62.94
Sample 03b	756.85	251.35	66.79
Sample 04b	309.58	99.98	67.70
Sample 05b	156.08	42.90	72.51
Sample 06b	3.68	2.81	23.74
Sample 07b	0.50	0.47	5.81
Sample 08b	5109.95	1739.10	65.97
Sample 09b	4333.19	1471.79	66.03
Sample 10b	3545.37	1072.13	69.76
Sample 11b	1411.09	434.54	69.21
Sample 12b	648.03	213.94	66.99
Sample 13b	147.90	48.27	67.37
Sample 14b	2.00	1.97	1.35

Source: Authors.

Images with only two times have a low time reduction (2.92% - sample 08a; 5.81% - sample 07b; 2.08% - sample 16a; 1.35% - sample 14b). This fact is justified due to the small size of the image, the hardware can work with much of the latter in its main memory, reducing the need to constantly search for parts of the image in the secondary memory (swapping). The same goes for samples with a

spatial dimension of 3000×2481 and temporal dimension of 12 times (27.62% - sample 07a; 23.74% - sample 06b)

In general, for images with a spatial dimension of 3000×2481, the average of time reduction is 54.38%, while the time reduction in images with a spatial dimension of 5000×6296 is, on average, 58.45%. This shows that for images with a larger size, the time reduction is also higher. This feature can also be seen when analyzing the individual time reductions presented in Table 2.

UPLOADING THE CUBE FOR VISUALIZATION

After assembling the four-dimensional cubes, it is required to load it into the prototype for the purpose of viewing a face image and/or the temporal spectrum of a pixel. Upon loading the temporal cube in the prototype, it performs a scan around the cube seeking the smallest and largest pixel values present in the whole 4D cube.

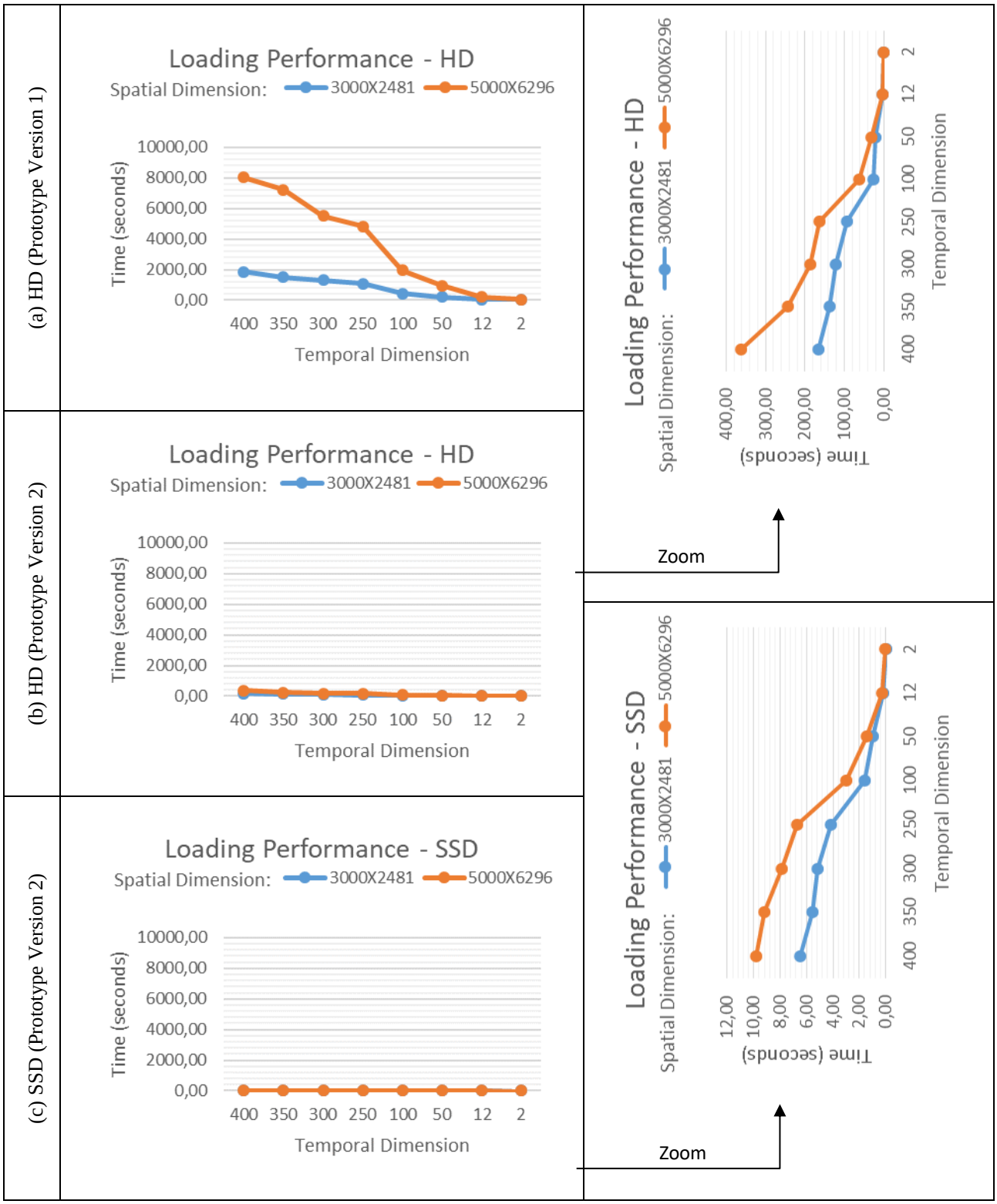
To verify the performance of the prototype at this stage, the four-dimensional cubes assembled for each sample were loaded. Such loading was made on an initial version of the prototype using an HD (Sub-Figure 10(a)), in which the multi-pointer technology was not used yet. Also, the loading tests were performed in the second version of the prototype, in which the multi-pointers technique was already implemented, on an HD (Sub-Figure 10(b)) and an SSD (Sub-Figure 10(c)). For a better visualization, the graphs in Figure 10 are initially presented using as reference for the y axis a scale of time values (in seconds) taken by the prototype during the loading of the cube in the first version of the software. In a second phase, the graphs of tests in the new version of the prototype are presented over again, in their own time scales (in seconds) of the loading processing of each 4D cube.

Figure 10 presents the performance graphics regarding the loading of the four-dimensional cube, previously assembled in the prototype. The graphs shown are related to the cubes generated from multi-bands images, as from such images it was possible to generate cubes with a temporal dimension range from 2 to 400, while, from multi-times images, the maximum temporal dimension feasible for processing reached 350 times.

Loading sample 01a in the old version of the prototype, utilizing an HD, required 1851.92 seconds. Loading the same sample in the second version of the software (with the multi-pointers technique), and an HD, required a processing time of 164.60 seconds. One can see that the time reduction was circumstantial (approximately 1687 seconds). The sample 09a loading time, in the old version of the prototype, was 8051.32 seconds, while 362.80 seconds for loading in the latest version of

the prototype, that is, over 7688 seconds of saved time. By doing the same tests using an SSD, the time reduction is even higher, 6.52 and 9.83 seconds respectively for samples 01a and 09a.

Figure 10. 4D Cube Loading Performance.



Source: Authors.

Cubes of intermediate sizes, such as those assembled from samples 04a and 12a, were loaded into 1097.89, and 4841.63 seconds, respectively, in the first version of the prototype. In the current version, using HD, loading took place at 93.27 (sample 04a) and 164.32 (sample 12a) seconds. By using an SSD, the time was 4.12 seconds for sample 04a and 6.72 seconds for sample 12a.

Temporal cubes assembled from smaller images, of 12 times (samples 07a and 15a), require, in the first version of the prototype, approximately 47.17 (07a) and 211.62 (15a) seconds to load. Using an HD and the version of the prototype with the multi-pointers implementation, this time falls to 3.10 (07a) and 3.42 (15a) seconds. Using multi-pointers and SSD, the processing time to load cubes is 0.17 (07a) and 0.27 (15a) seconds.

In the case of temporal dimension 2 (samples 08a and 16a), the lowest temporal dimension tested, the prototype needed, with no multi-pointers, 5.97 (08a) and 33.17 (16a) seconds to load the cubes. With multi-pointers and processing performed on an HD, 0.02 (08a) and 0.50 (16a) seconds were the loading times for four-dimensional cubes. With the SSD technology, the prototype required 0.0 and 0.01 seconds to load, respectively, cubes 08a and 16a.

There was a significant improvement upon implementing multi-pointers in the loading phase of the image, what can be determined by analyzing the graphs shown in Figure 10. Such improvement is enhanced by joining the technique of multi-pointers with the native features of the SSD technology.

The loading performance, utilizing the multi-pointers version of the prototype for temporal cubes from samples 08a and 16a, shows that images of low spatial and temporal dimensions have a similar processing time in the use both HD and SSD. As in the assembly of four-dimensional cubes, this fact occurs due to the size of the memory buffer set for the loading phase of the 4D cube.

Table 3 surveys the time reduction when loading an already assembled 4D cube. First, on improving the prototype by implementing multi-pointers, i.e., from versions 1 to 2 of the prototype, both tested in HD. Following, the table displays the time reduction solely related to the second version of the software when using SSD. All demonstrations are performed for all samples.

Table 3 presents the time reduction by using, on an HD, the prototype in its second version compared to version 1; and, in the second version, on an SSD compared to its use on an HD. In the tested images with higher temporal dimension, i.e. samples 01a and 09a, the time reduction reaches 91.11% and 95.49%, respectively, if compared between versions 01 and 02 of the prototype; and 96.04% and 97.29%, respectively, when compared between the HD and SSD on version 2. The final

time reduction, comparing the first version of the prototype running on an HD and version 2 of the prototype, running on an SSD, reaches 99.65% (sample 01a) and 99.88% (sample 09a).

Table 3. Performance Time Reduction in Loading the 4D Cube.

Sample	Loading Time for 4D Cube (seconds)		Gain (%)	Loading Time for 4D Cube (seconds)		Gain (%)	Final Gain (%)
	Version 1 on HD	Version 2 on HD		Version 2 on HD	Version 2 on SSD		
Sample 01a	1851.92	164.60	91.11	164.60	6.52	96.04	99.65
Sample 02a	1497.66	137.16	90.84	137.16	5.57	95.94	99.63
Sample 03a	1295.40	122.31	90.56	122.31	5.20	95.75	99.60
Sample 04a	1097.89	93.27	91.50	93.27	4.12	95.59	99.62
Sample 05a	439.60	26.00	94.08	26.00	1.62	93.76	99.63
Sample 06a	207.33	20.58	90.08	20.58	0.97	95.30	99.53
Sample 07a	47.17	3.10	93.42	3.10	0.17	94.46	99.64
Sample 08a	5.97	0.02	99.75	0.02	0.00	0.00	100.00
Sample 09a	8051.32	362.80	95.49	362.80	9.83	97.29	99.88
Sample 10a	7232.05	242.94	96.64	242.94	9.20	96.21	99.87
Sample 11a	5528.04	186.16	96.63	186.16	7.91	95.75	99.86
Sample 12a	4841.63	164.32	96.61	164.32	6.72	95.91	99.86
Sample 13a	1956.18	63.06	96.78	63.06	3.01	95.22	99.85
Sample 14a	965.51	32.18	96.67	32.18	1.42	95.59	99.85
Sample 15a	211.62	3.42	98.39	3.42	0.27	92.21	99.87
Sample 16a	33.17	0.50	98.50	0.50	0.01	96.99	99.95

Source: Authors.

In medium images, containing 250 times, the time reduction reaches 91.50% (sample 04a) and 96.61% (sample 12a), in the comparison between versions 1 and 2 of the prototype. If compared in the version 2 on an HD and on an SSD the time reduction is 95.59% (sample 04a) and 95.91% (sample 12a), gaining these samples a final improvement of 99.62% (sample 04a) and 99.86% (sample 12a).

One can tell, by analyzing Table 3, that in all sections the average time reduction in performance is 94.82% if versions 1 and 2 of the prototype are compared, both tested on an HD. By verifying the second version, the average time reduction in loading a 4D cube, on HD and SSD, is 89.50%. The reduction of performance gain occurs since, in version 2 of the software, sample 08a was loaded in very similar times in both technologies, with no time reduction in this case. The overall average of time reduction in loading a 4D cube is 99.77%.

Table 3 shows that, in fact, the improvement achieved by implementing multi-pointers is real, reaching more than 99% increase in performance in loading the four-dimensional cubes.

THE VISUALIZATION OF THE TEMPORAL SPECTRUM OF A PIXEL

With a four-dimensional cube loaded in the prototype, it is possible to realize the visualization of the temporal spectrum from a selected pixel. This visualization is performed as a graph, in which the

y axis limiting values are the lowest and the highest values of DN collected from the 4D cube. The x axis has the temporal dimension of the cube as marks, and the spectral dimension is represented individually (in each band) by the lines shown in the graph.

To verify the performance of the prototype at this stage, four-dimensional cubes assembled for each sample were loaded, this loading was made on an initial version of the prototype using an HD (Sub-Figure 11(a)), in which the multi-pointer technology was not used. In addition, the loading tests were performed in the second version of the prototype, in which the multi-pointers technique was already implemented, in HD (Sub-Figure 11(b)) and SSD (Sub-Figure 11(c)). Five random pixels were collected, from the beginning to the end of the cube, and the values presented are the arithmetic mean of the collected values.

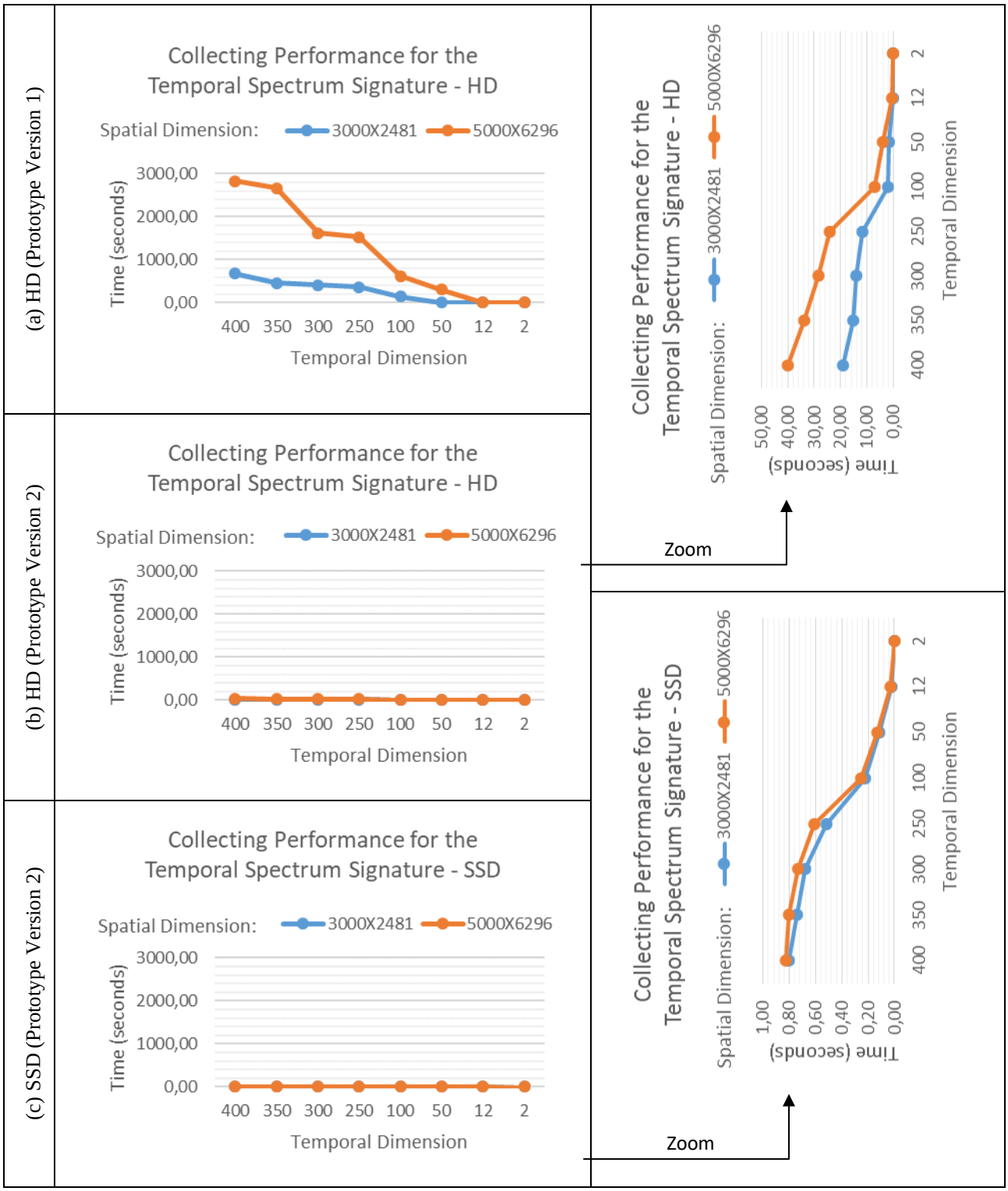
For a better visualization, the graphs in Figure 11 are initially presented using as values on the y axis a time scale (in seconds) taken by the prototype during the collection for the cube, using the first version of the software. In a second moment, the graphs of the tests on the new version of the prototype are presented, in their own processing time scales for the collection of the temporal spectrum (in seconds) of the pixels for each 4D cube assembled.

As well as in the analysis of the loading time of a four-dimensional cube in the prototype, one can perceive a performance improvement obtained with the implementation of multi-pointers in the collection and visualization phases of a temporal spectrum. The technique brought the prototype considerable time reduction, which is even better exploited when used in conjunction with the SSD technology.

To collect the temporal spectrum in a cube of temporal dimension 400 (samples 01a and 09a), the prototype spends 674.25 (01a) and 2827.24 (09a) seconds when its first version is used on HD. The performance difference of the temporal spectrum collection is huge compared to the new version of the prototype, which spends approximately 18.99 (01a) and 21.21 (09a) seconds when used on HD, or just 0.81 (01a) and 0.83 (09a) seconds when used on SSD.

In order to collect data from medium-sized cubes, 250 times, i.e., samples 04a and 12a, the prototype requires 362.14 and 1525.38 seconds, respectively, when using the first version of the prototype without multi-pointers. Performing the same operation on the new version of the prototype, the time spent is 11.72 (04a) and 12.65 (12a) seconds when running on HD, or 0.52 (04a) and 0.61 (12a) seconds if using SSD.

Figure 11. Collecting Performance for the Temporal Spectrum Signature.



Source: Authors.

Analyzing the processing time of the first version of the prototype on the collection of temporal spectrum in small sized images (samples 07a and 15a), one realizes that time of collection is

dramatically reduced, being 0.38 (07a) and 1.41 (15a) seconds, somewhat an acceptable time. However, when compared with the processing time of the multi-pointers version, which is 0.32 (07a) and 0.33 (15a) seconds using HD, or 0.03 seconds to both samples using SSD, one perceives that there is still a large performance gain in implementing multi-pointers.

In order to collect the temporal spectrum of cubes generated from samples 08a and 16a, the prototype spent 0.01 (08a) and 0.25 (16a) seconds when using the first version of the prototype on HD, and 0.0 seconds when using the multi-pointers version of the software, for both HD and SSD.

Moreover, it is clear that the time reduction is proportional to the image size. The larger the image, the greater the time reduction in using the new version of the prototype (with multi-pointers), compared to the previous version. I.e., for cubes with small temporal and spatial dimensions, the time reduction exists, however not as significant as for cubes with large dimensions.

Table 4 shows the data regarding the time reduction when loading an already assembled 4D cube, at first on the prototype improvement by implementing multi-pointers, i.e., from version 1 to version 2 of the prototype, both tested on HD. Following, the time reduction related solely to version 2 of the software when using SSD instead of using HD is displayed. All data is displayed for all samples.

Table 4. Performance Time Reduction for the Temporal Spectrum Signature Collection.

Sample	Signature Collecting Time (seconds)		Gain (%)	Signature Collecting Time (seconds)		Gain (%)	Final Gain (%)
	Version 1 on HD	Version 2 on HD		Version 1 on HD	Version 2 on SSD		
Sample 01a	674.25	18.99	97.18	18.99	0.81	95.75	99.88
Sample 02a	461.21	15.34	96.67	15.34	0.74	95.16	99.84
Sample 03a	410.46	14.00	96.59	14.00	0.68	95.12	99.83
Sample 04a	362.14	11.72	96.76	11.72	0.52	95.58	99.86
Sample 05a	139.29	2.30	98.35	2.30	0.22	90.37	99.84
Sample 06a	2.25	1.67	26.04	1.67	0.12	92.89	94.74
Sample 07a	0.38	0.32	15.55	0.32	0.03	92.10	93.33
Sample 08a	0.01	0.00	0.00	0.00	0.00	0.00	0.00
Sample 09a	2827.24	21.21	99.25	21.21	0.83	96.10	99.97
Sample 10a	2666.40	18.52	99.31	18.52	0.80	95.67	99.97
Sample 11a	1624.48	14.26	99.12	14.26	0.73	94.86	99.95
Sample 12a	1525.38	12.65	99.17	12.65	0.61	95.19	99.96
Sample 13a	624.32	4.81	99.23	4.81	0.25	94.75	99.96
Sample 14a	311.70	2.41	99.23	2.41	0.13	94.68	99.96
Sample 15a	1.41	0.33	76.33	0.33	0.03	89.75	97.57
Sample 16a	0.25	0.00	0.00	0.00	0.00	0.00	0.00

Source: Authors.

Table 4 shows the extent of the time reduction in performance in the collection of the Temporal Spectrum Signature upon using, on HD, the prototype in its second version, compared to version 1; and, in the version 2, on an SSD compared to its use on an HD. In the tested images with

higher temporal dimension, i.e. samples 01a and 09a, the time reduction reaches 97.18% and 99.25%, respectively, when compared between the two versions of the prototype; and 95.75% and 99.97%, respectively, when compared, in version 2, between the HD and SSD. The final time reduction, comparing the first version of the prototype running on HD, and version 2 of the prototype on SSD, reaches 99.88% (sample 01a) and 99.97% (sample 09a).

In medium images, with 250 times, the time reduction reaches 96.76% (sample 04a) and 99.17% (sample 12a), in comparing versions 1 and 2 of the prototype. In comparing version 2 on HD and SSD it is 95.58% (sample 04a) and 95.19% (sample 12a), remaining these sections with a final time reduction of 99.86% (sample 04a) and 99.96% (sample 12a).

In images of two times, the time reduction is 0.00% (samples 08a and 16a) in all tests, granting no time reduction from version 1 to 2, or on the use of SSD in relation to HD. This occurs in part due to the absence of swapping and mainly due to the implementation of multi-pointers which rendered the temporal spectrum signature collecting process very fast if compared to the first version of the prototype.

By analyzing Table 4, one can realize that for all samples the average of time reduction is 74.92% when comparing version 1 with version 2 of the prototype, both tested on HD. Upon verifying the time reduction in the second version, by comparing the performance in collecting the Spectral signature of a pixel on HD and SSD, the average reduction is 82.37%. The overall average of time reduction in collecting the Spectral signature of a pixel is 86.54%.

FINAL CONSIDERATIONS

The need to analyze changes in land use over time is constant. Scholars and researchers seek to analyze these changes, but due to software limitations to assist in this study, the time dimension is limited. This article brought new concepts for the use of time in this analysis, contributing to a significant improvement in processes involving the analysis of land use over time.

Two key concepts for a better use of the temporal dimension in the analysis of changes in spectral behavior were presented. One is the inclusion of the temporal dimension in popular storage formats of digital data (BSQ, BIL and BIP), bringing, therefore, a new approach to the storage of digital images on disc: TBSQ, TBIL and TBIP formats, enabling the treatment of four-dimensional data (Lines, Columns, Bands and Times) into digital data files. These new digital formats allow the manipulation of temporal data in an efficient and effective manner.

Another concept introduced was the use of multi-pointers on the temporal spectrum processing of n-temporal images. This brought a significant performance gain for the prototype, in both the loading of the four-dimensional cube and in the collection phase of the temporal spectrum signature of a selected pixel on a cube face.

Performance tests were conducted in three phases of n-temporal image processing: i) the assembly phase of the four-dimensional cube; ii) the loading phase of an already assembled 4D cube; iii) and in the collection phase of the temporal spectrum signature of a selected pixel. As expected, generally speaking, the image processing time is directly proportional to its size. The tests were performed on HD and SSD technologies, thereby demonstrating the improved efficiency of the version 2 of the prototype when utilized along with the SSD technology. In small images, the difference in processing time between the use of HD and SSD is imperceptible to the user, since no swapping occurs - the main memory of the hardware was sufficient to load the entire image during the process.

Tests performed comparing the processing time of the initial version (version 1) of the prototype and the time spent by the current prototype version (version 2), in which the concepts of multi-pointers are already implemented, demonstrate the effectiveness of this method by presenting significant time reduction from the first run on HD to the new version on HD and SSD.

Thus, the experiments proved the effectiveness of the proposed concepts, in practice demonstrating the advantages that their use brings to the spectrum analysis over time.

ACKNOWLEDGMENTS

Author NSP thanks to Fundação de Amparo à Pesquisa em Goiás (FAPEG) for Financial Support during the Master's degree, when the research has been carried out.

REFERENCES

Carreira JC, Rudke AP, Brito ACC, Bezerra RR, Santos AM 2012. Técnicas de Sensoriamento Remoto Aplicadas à Análise Da Cobertura Vegetal No Município de JI-Paraná/RO. *Revista Brasileira de Ciências Da Amazônia*, 1(1):38-43.

Carvalho Júnior OA, Silva NC 2007. Detecção de Mudança Espectral uma nova metodologia para análise de séries temporais. In *Anais Simposio Brasileiro de Sensoriamento Remoto* XIII. Florianópolis, Brasil: INPE: 5635-41.

Carvalho Júnior OA, Couto Júnior AF, Silva NC, Martins ES, Carvalho APF, Gomes RAT 2009. Avaliação Dos Classificadores Espectrais de Minima DistâNcia Euclidiana e Espectral Correlation Mapper Em Séries Temporais NDVI-MODIS No Campo de Instruçao Militar de Formosa (GO). *Revista Brasileira de Cartografia*, 61(4):399-412.

Carvalho Júnior OA, Sampaio VS, Silva NC, Couto Júnior AF, Gomes RAT, Carvalho APF, Shimabukuro YE 2008. Classificação de Padrões de Savana Usando Assinatura Temporais NDVI Do Sensor MODIS No Parque Nacional Chapada Dos Veadeiros. *Revista Brasileira de Geofísica*, 26(4):505-17.

Carvalho Júnior OA, Hermuche PM, Guimarães RF 2006. Identificaçao Regional Da Floresta Estacional Decidual Na Bacia Do Rio Paranã a Partir Da Análise Multitemporal de Imagens MODIS. *Revista Brasileira de Geofísica*, 24(3):319-32.

Carvalho Júnior OA, Guimarães RF, Gillespie AR, Silva NC, Gomes RAT 2011. A New Approach to Change Vector Analysis Using Distance and Similarity Measures. *Remote Sensing*, 3(12):2473-93. DOI: https://doi.org/10.3390/rs3112473 .

Carvalho Júnior OA, Guimarães RF, Carvalho APF, Gomes RAT, Melo AF, Silva PA 2005. Processamento e análise de imagens multitemporais para o perímetro de irrigação de Gorutuba (MG). In *Anais Simposio Brasileiro de Sensoriamento Remoto* XII. Goiânia, Brasil: INPE: 473-80.

Carvalho APF, Carvalho Júnior OA, Guimarães RF, Martins ES, Rocha VMS, César CS 2002. Análise Multitemporal de Imagens NDVI Em Ações de Reforma Agrária. *Espaço & Geografia*, 5(1):139-52.

Couto Júnior AF, Carvalho Júnior OA, Martins ES, Encinas JI 2007. Identificação Das Assinaturas Temporais NDVI Das Principais Fisionomias Da Região Do Ecomuseu Do Cerrado. *Espaço & Geografia*, 10(1):173-89.

Couto EV, Souza Filho EE, Hayakawa EH 2011. Análise Das Modificações Da Cobertura Vegetal Da Planície Fluvial Do Alto Rio Paraná No Período Entre 1976 e 2007. *Acta Scientiarum. Technology*, 33(2):205-13.

Dias TL, Câmara G, Davis Júnior CA 2005. Modelos Espaço-Temporais. In MA Casanova, G Câmara, CA Davis Júnior, L Vinhas, GR Queiroz (eds.). *Bancos de Dados* Geográfico. Editora Mundogeo, Curitiba, p. 147-79. http://www.inf.puc-rio.br/~casanova/Publications/Books/2005-BDG.pdf.

Diedrich TJ, Narvaes IS 2013. Técnicas de Detecçao de Mudanças Em Áreas Urbanas de Brasília. *Revista Brasileira de Cartografia*, 65(3):493-509.

Florenzano TG 2011. *Iniciação Em Sensoriamento Remoto*. 3.ed. Oficina de Textos, São Paulo. http://books.google.com/books?id=18GkH5X81XcC&pgis=1.

Gomes DDM, Mendes LMS, Medeiros CN, Veríssimo CUV 2011. Análise Multitemporal Do Processo de Degradação Da Vegetação Da Bacia Hidrográfica Do Rio Jaibaras No Estado Do Ceará. *Geografia Ensino & Pesquisa*, 15(2):41-62.

Gonzalez RC, Woods RE 2011. *Digital Image Processing*. Vol. 21. Pearson Education.

Gurmessa D, Nemomissa S, Tadesse G 2013. Borana Rangeland Change from 1986-2003: The Application of Remote Sensing and GIS. *Proceedings of Global Geospatial Conference*, 4(8).

Harris Geospatial Solutions 2018. ENVI - Environment for Visualizing Images. Accessed September 15. http://www.harrisgeospatial.com/docs/using_envi_Home.html.

Ioannis M, Miliadis M 2011. Multi-Temporal Landsat Image Classification and Change Analysis of Land Cover/Use in the Prefecture of Thessaloniki, Greece. *Proceedings of the International Academy of Ecology and Environmental Sciences*, 1(1):15-25.

Japan Association of Remote Sensing 1993. Data Used in Remote Sensing. In *Remote Sensing Note*. http://www.jars1974.net/pdf/rsnote_e.html.

Kampel M, Amaral S, Soares MLG 2005. Imagens CCD/CBERS e TM/Landsat Para Análise Multi-Temporal de Manguezais No Nordeste Brasileiro. Um Estudo No Litoral Do Estado Do Ceará. In *XII Simpósio Brasileiro de Sensoriamento Remoto*. INPE, Goiânia, p. 979-986.

Lillesand TM, Kiefer RW, Chipman JW 2004. *Remote Sensing and Image Interpretation*. 5.ed. John Wiley & Sons, New York.

Lorena RB, Santos JR, Shimabukuro YE 2004. Técnica de Detecção de Mudanças Aplicada Em Imagens de Satélite Como Ferramenta de Monitoramento de Uso e Cobertura Da Terra Na Amazônia Ocidental (Acre). *Espaço & Geografia*, 7(1):47-76.

Lorena RB, Santos JR, Shimabukuro YE, Sant'Anna HM, Sant'Anna HSS, Menezes RS 2001. Dados Multitemporais de Sensoriamento Remoto Para a Análise Da DinâMica Do Uso e Da Cobertura Da Terra Na Região Do Peixoto (AC). In *X Simpósio Brasileiro de Sensoriamento Remoto - SBSR*. INPE, Foz do Iguaçu, p. 1653-56.

Mata CL, Carvalho Júnior OA, Carvalho APF, Gomes RAT, Martins ES, Guimarães RF 2007. Avaliação Multitemporal Da Susceptibilidade Erosiva Na Bacia Do Rio Urucuia (MG) Por Meio Da Equação Universal de Perda de Solos. *Revista Brasileira de Geomorfologia*, 8(2):57-71.

Menke AB, Carvalho Júnior OA, Gomes RAT, Martins ES, Oliveira SN 2009. Análise das Mudanças do Uso Agrícola da Terra a Partir de Dados de Sensoriamento Remoto Multitemporal no Município de Luis Eduardo Magalhães (BA - Brasil). *Sociedade & Natureza*, 21(3):315-26.

Montabone S 2010. *Beginning Digital Image Processing: Using Free Tools for Photographers*. Apress, New York.

Moreira MA, Adami M, Rudorff BFT 2004. Análise Espectral e Temporal Da Cultura Do Café Em Imagens Landsat. *Pesquisa Agropecuária Brasileira*, 39(3):223-31.

Oliveira ES 2013. Estudo Preliminar Sobre a Evolução Do Uso e Ocupação Do Solo No Município de Planaltina de Goiás. *GeoTemas*, 3(1):111-23.

Petrou M, Petrou C 2010. *Image Processing: The Fundamentals*. 2.ed. John Wiley & Sons, New Jersey.

Qian SE 2013. *Optical Satellite: Signal Processing and Enhancement*. Spie Press, Bellingham.

Qin J, Chao K, Kim MS 2010. Raman Chemical Imaging System for Food Safety and Quality Inspection. *American Society of Agricultural and Biological Engineers*, 53(6):1873-82.

Santana OA, Carvalho Júnior OA, Pimentel CMM, Gomes RAT, Oliveira SN 2010. Modelagem de Espectros Temporais NDVI-MODIS, No Periodo de 2000 a 2008 Na Bacia Do Rio Paracatu, Brasil. *Revista Brasileira de Geofísica*, 28(1): 47-60.

Sausen TM 2012. Sensoriamento Remoto e Suas Aplicações Para Recursos Naturais. http://www.inpe.br/unidades/cep/atividadescep/educasere/apostila.htm#tania.

Scheer MAPS, Rocha JV 2006. Detecção de Mudanças No Uso Da Terra No Município de Sertãozinho (SP) Por Meio de Técnicas de Geoprocessamento, 1981 - 2001. *Revista Brasileira de Cartografia*, 58(02):163-74.

Schowengerdt RA 2007. *Remote Sensing: Models and Methods for Image Processing*. 3.ed. Elsevier, California.

Shaikh AA, Gotoh K, Tachiiri K 2005. Multi-Temporal Analysis of Land Cover Changes in Nagasaki City Associated with Natural Disasters Using Satellite Remote Sensing. *Journal of Natural Disaster Science*, 27(1):9-15.

Silva NC, Carvalho Júnior OA, Rosa ANCS, Guimarães RF, Gomes RAT 2012. Change Detection Software Using Self-Organizing Feature Maps. *Revista Brasileira de Geofísica*, 30(4):505-18.

Vieira CHSD, Biondi D 2008. Análise Da DinâMica Da Cobertura Vegetal de Curitiba, PR (de 1986 a 2004), Utilizando Imagens LANDSAT TM. *Revista Árvore*, 32(3):479-87.

Voorde TV, Demarchi L, Canters F 2009. Multi-Temporal Spectral Unmixing to Characterise Urban Change in the Greater Dublin Area. *Remote Sensing for a Changing Europe*, 276-283. DOI: https://doi.org/10.3233/978-1-58603-986-8-276.

Zhou W, Troy A, Grove M 2008. Object-Based Land Cover Classification and Change Analysis in the Baltimore Metropolitan Area Using Multitemporal High Resolution Remote Sensing Data. *Sensors*, 8:1613-1636.

Melhoramento de Métodos e Técnicas que Subsidiam a Análise de Alterações de Uso de Solo no Decorrer do Tempo

RESUMO

Tanto a evolução natural quanto as atividades humanas acarretam em alterações nos recursos naturais e no meio ambiente terrestre. A observação a nível temporal e espacial da Terra nos permite entender o inter-relacionamento dos fenômenos causadores dessas mudanças. Os estudos realizados neste sentido variam de acordo com a necessidade dos pesquisadores e do objeto a ser estudado. Existe a necessidade de se trabalhar com uma grande quantidade de tempos em uma série temporal de imagens multiespectrais a fim de obter dados mais detalhados das mudanças do espaço no decorrer do tempo. O objetivo deste trabalho é contribuir com a área de geoprocessamento no que se refere ao melhoramento de métodos e técnicas que subsidiem as análises de mudanças de uso do solo ao longo do tempo. Para isso, uma abordagem foi proposta a fim de possibilitar a análise de uma imagem com k-bandas, em n-tempos simultaneamente, para isso a dimensão tempo foi agregada ao modelo de uma

imagem digital comum, tridimensional (linhas, colunas e bandas), tornando-a quadridimensional (linhas, colunas, bandas e tempos). Foram elaborados novos formatos de armazenamento de dados multiespectrais e multitemporais (TBSQ, TBIL, TBIP), para testá-los, foi desenvolvido um protótipo em duas etapas. A primeira versão do protótipo consumiu um considerado tempo na execução dos testes, levando à necessidade de aprimoramento do protótipo. Na segunda versão do protótipo foi implementada uma técnica de multi-ponteiros, o que trouxe ganhos significativos de performance.

Palavras-Chave: Multidimensional; Imagens Espectro-temporais; Multitemporal; TBSQ, TBIL e TBIP; Multi-Ponteiros.

Submission: 14/10/2018
Acceptance: 16/04/2019