UDC 004.932.2
DOI: 10.15587/1729-4061.2024.306227

DEVELOPMENT OF A PROCEDURE FOR FRAGMENTING ASTRONOMICAL FRAMES TO ACCELERATE HIGH FREQUENCY FILTERING

Vladimir Vlasenko
PhD
Space Research and Communications Center
National Space Facilities Control and Test Center
Moskovska str., 8, Kyiv, Ukraine, 01010

Sergii Khlamov
Corresponding author
PhD, Test Automation Lead
SoftServe
Sadova str., 2D, Lviv, Ukraine, 79021
E-mail: sergii.khlamov@gmail.com

Vadym Savanevych
Doctor of Technical Sciences, Professor
Department of Systems Engineering**
Kharkiv National University of Radio Electronics
Nauky ave., 14, Kharkiv, Ukraine, 61166

Tetiana Trunova
Engineer*

Zhanna Deineko
PhD, Associate Professor*

Iryna Tabakova
PhD, Associate Professor*

*Department of Media Systems and Technologies**
**Kharkiv National University of Radio Electronics
Nauky ave., 14, Kharkiv, Ukraine, 61166

The object of this study is the process of filtering astronomical frames that contain images of potential objects in the Solar System. To contour the image of each such object and recognize it in contrast with the background of the frame, it is necessary to filter the image. Most often, a variety of high-pass filters are used to determine the high-frequency component of the image, which can be removed as a coarse-grained component. Any image filtering is aimed at increasing the signal-to-noise ratio and reducing the dynamic range of the background image. However, the filtering process is quite resource- and time-consuming. This is especially true for systems for parallel processing of series of astronomical frames in real time (online). Therefore, to solve the problem of lack of frame fragmentation, which leads to high consumption of RAM, a procedure for fragmenting astronomical frames has been proposed.

Owing to the introduction of a formal connection between the values of frame pixels and fragments, as well as determining their number, it was possible to reduce RAM utilization. Testing was carried out using the following high-pass filters – ideal filter, Butterworth filter, and Gaussian filter. Using the devised procedure for fragmenting astronomical frames has made it possible to reduce the utilization of RAM during filtering. As a result, with parallel processing, this has also made it possible to speed up the high-frequency filtering procedure itself.

The procedure devised for fragmenting astronomical frames was tested in practice within the framework of the CoLiTec project. It was implemented in the On-Line Data Analysis System (OLDAS) of the Lemur software.

The study showed that when using the devised procedure, RAM utilization was reduced by 7–10 times. And the speed of filtration itself increased by 2–3 times. Accordingly, the processing time for each astronomical frame was reduced by 2–3 times

Keywords: frame fragmentation, multiprocessing, high-pass filtering, ideal filter, Butterworth filter, Gaussian filter

Received date 09.04.2024
Accepted date 14.06.2024
Published date 30.06.2024

***How to Cite:** Vlasenko, V., Khlamov, S., Savanevych, V., Trunova, T., Deineko, Zh., Tabakova, I. (2024). Development of a procedure for fragmenting astronomical frames to accelerate high frequency filtering. Eastern-European Journal of Enterprise Technologies, 3 (9 (129)), 70–77. https://doi.org/10.15587/1729-4061.2024.306227*

1. Introduction

The field of observational astronomy [1] continues to actively evolve due to the constant improvement of various methods of astrometry [2] and photometry [3]. However, the main source of astronomical data (series of frames) is ground-based and space telescopes. In observational astronomy, astronomical frames are not only accumulated but also processed by various image processing methods. Owing to improvements and breakthroughs in digital technology and telescope construction, astronomical data has become big data [4]. Large asteroid surveys consist of several telescopes equipped with very large charge-coupled device (CCD) cameras [5]. Automated processing of such frames requires consistency in the sequence of operations on the processing pipeline, as well as high power of multiprocessor systems. During such processing, huge astronomical catalogs and archival big data are also involved [6]. That has made it possible to accumulate, obtain knowledge [7], and analyze acquired publicly available data [8] and measurements. Most often, astronomical data are classified into specific classes/types of Solar System Objects (SSOs) [9] (for example, asteroids [10] or comets), as well as artificial Earth satellites. Such measurements were obtained over a long period of observation and their analysis may even include calculation of the period and shape of rotations of such SSOs [11].

Any process of preliminary preparation of an astronomical frame using brightness leveling of the background or using calibration frames is very resource- and time-consuming. This is a rather labor-intensive and inconvenient operation for the observer, which requires a properly configured processing pipeline. In addition, during parallel processing of a series of frames under online mode, requirements are set both for the RAM used and for the actual execution time of the operation at each stage of the pipeline.

Therefore, it is a relevant task to devise a procedure for fragmenting astronomical frames. This technique will reduce RAM utilization during the high-pass filtering procedure. This, in turn, will speed up the process of preliminary frame preparation using brightness leveling of the background or using calibration frames. As a consequence, the brightness and positional coordinates of the objects under study will be more quickly assessed [12]. And this will already increase the conditional probability of correct detection (CPCD) of real objects [13] under the mode of their tracking.

2. Literature review and problem statement

In the field of processing astronomical frames, one of the key problems relates to the heterogeneity of the typical shape of the image of objects [14]. The typical shape directly depends on the conditions and quality of shooting with a CCD camera. It is an important factor that influences the subsequent process of detecting objects and assessing their position and brightness. It is assumed that methods of preliminary image preparation can remove artifacts and shifts in the positional coordinates of the frame center between frames in a series, as well as align the brightness of the background. All such methods of preliminary image preparation are aimed at improving the quality of the frames themselves, which significantly affects the accuracy of the main image processing methods. However, any parallel processing pipeline in a multiprocessor system imposes strict requirements on the use of RAM and on the speed of preprocessing. All these requirements directly depend on the throughput of the system for simultaneous online frame processing.

Computer vision methods [15] have the main disadvantage that they are not able to provide the required level of processing speed using standard libraries. Classical methods for object image recognition [16, 17] require analysis of all pixels of potential objects to determine their typical shape. This also significantly affects processing time. In the case of heterogeneity of the standard shape, objects become confused, and this, in turn, increases the processing time of both the strobe with the image of the object and the entire frame.

Methods for estimating image parameters [18, 19] are based on the analysis of only those pixels that potentially belong to the object under study. Their disadvantage is the inability to initially accurately determine specific pixels and reject those whose intensity exceeds a specified limit value. Because of this, the processing time of each frame increases.

The authors of [20] use automatic selection of a reference point as preprocessing to select calibration frames. This algorithm requires quite a lot of execution time because it involves first analyzing the image to find reference points. Next, these points are identified with astronomical catalogs to obtain coordinates on the celestial sphere. After this, the telescope is pointed at the coordinates of the desired area of the sky and takes calibration frames. This entire algorithm requires the coherence of all systems, including the automated telescope mount. At the same time, if there are artifacts in the images, these control points may be false, this leads to an increase in processing time (to search for new control points), or even the impossibility of executing the algorithm.

The median pooling algorithm [21] proposes to accumulate a large number of frames for pre-calibration to get rid of real objects, leaving only a uniform background signal. The disadvantage of this approach is the fact that the formation and use of data arrays for each such image requires huge volume of RAM.

The variety of typical shapes (stroke, extended, circular) or the intersection of an object image with parasitic illumination also affects various methods of cluster analysis of large data sets [22] and time series [23]. The disadvantage of these methods is that their application requires a very large sample of data with "pure" measurements, and the heterogeneity of the standard form will greatly spoil the overall indicator. A similar influence of the heterogeneity of the standard shape appears on the Wavelet transform methods [24, 25]. Thus, the processing time of the entire series of frames will also increase, which contradicts the established requirements for the rapid processing of series of frames online.

A matched filtering algorithm is also known [26, 27], but it uses only an analytical image model. The disadvantage of this algorithm is the use of a huge volume of RAM in the case of processing a frame with a high resolution (more than 4000×4000 pixels).

Work [28] presents an implementation of fragmentation using the PointRend neural network (point rendering). This module performs fragmentation prediction for segments of potential objects based on points at adaptively selected locations based on an iterative subdivision algorithm. However, the efficiency of PointRend provides an output resolution that would otherwise be impractical in terms of memory and computation compared to existing approaches.

The widespread success of the deep learning approach has also generated interest for its application in the field of image fragmentation and segmentation [29]. The cited work proposes the use of convolutional pixel labeling networks, recurrent networks, adversarial visual attention models, and generative methods. However, these approaches require a long time to prepare the training sample and the deepest training. And in the case of online image processing, processing time is very critical.

Based on the above, all existing image preprocessing methods perform operations on the entire source frame. With large resolutions of such frames, these operations will require a large volume of RAM. On weaker multiprocessor systems there are limits on available RAM and a processing queue. Accordingly, under the online frame receipt mode, the series itself as a whole will be processed much longer due to the introduced restrictions on the use of RAM.

Thus, a revealed problem in online high-frequency processing is the lack of frame fragmentation (splitting the frame into fragments) for subsequent parallel processing of the fragments separately. Existence of this problem leads to extreme RAM utilization. Therefore, it is necessary to devise a procedure for fragmenting an astronomical frame, which will divide the frame into fragments, taking into account the peculiarities of frame formation. It will make it possible to significantly reduce RAM utilization and speed up the process of subsequent frame processing (for example, the process of high-frequency filtering).

3. The aim and objectives of the study

The purpose of our study is to devise a procedure for fragmenting astronomical frames, ensuring a reduction in RAM utilization during image processing by high-frequency filtering. This will speed up the process of processing astronomical frames in general.

To achieve the goal, the following tasks were set:

– to introduce a formal connection between the values of frame pixels and fragments;

– to determine the number of fragments of the astronomeical frame;

– to develop an algorithm for the procedure for fragamenting astronomical frames;

– to verify the procedure for fragmenting astronomical frames.

4. The study materials and methods

The object of our study is the process of filtering astronomical frames that contain SSO images. The initial series for the study were acquired from a variety of telescopes installed at observatories in Ukraine and around the world. Namely, the ISON-NM observatory, the SANTEL-400AN telescope (New Mexico, USA); Vihorlat Observatory, VNT telescope (Humenne, Slovakia) [30]; Odesa-Mayaky observatory, OMT-800 telescope (Mayaki, Ukraine) [31]. The original astronomical frames considered for the study had a variety of resolutions, namely: 512×512, 768×512, 3056×3056, and 4008×2672 pixels.

Implementation of high-frequency filtering of large images with discrete Fourier transform (DFT) requires significant volume of RAM. We propose to test the hypothesis that dividing the original frame into fragments and sequentially filtering them could reduce the volume of RAM required for frequency filtering. And as a consequence of this, with parallel processing, the speed of frame processing as a whole could increase.

The high-pass filtering methods used were: ideal filter, Butterworth filter, and Gaussian filter [32].

The devised procedure for fragmenting astronomical frames was implemented in the C++ programming language. This code was used at the stage of preliminary processing of the Lemur software package (Ukraine) [33] for the automated detection of new and tracking of known objects within the CoLiTec project [34].

The devised procedure, implemented in Lemur software (Ukraine), was used during testing on various servers, including cloud ones. As you know, cloud servers have a limited volume of RAM and are quite expensive. Therefore, the devised procedure has confirmed its practical significance primarily for cloud servers.

5. Results of investigating the procedure for fragmenting astronomical frames

5. 1. Formal connection between frame and fragment pixel values

The use of DFT during high-pass filtering makes it possible to reduce the complexity of the filtering in the spatial domain [35]. The original frame A_{in} of size N_{CDDx}×N_{CDDy} is divided into M_x×M_y fragments A_{crkl} of size N_{crx}×N_{cry}, which depends on the allowable amount of memory used for frequency filtering (Fig. 1).

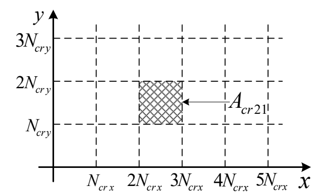

Fig. 1. Covering the original frame with fragments

When dividing the source frame into fragments, it is necessary to have a clear array of pixels belonging to each fragment of the source frame. To this end, it is proposed to introduce a formal connection between the pixel values of the original frame A_{in} and the pixel values of its *kl*-th fragment A_{crkl}:

$$A_{crkl}(m, n)=A_{in}(m+N_{crx}k, n+N_{cry}l), \quad (1)$$

where $A_{crkl}(m, n)$ – pixel brightness of the *kl*-th frame fragment with numbers *m*, *n*;

$A_{in}(m, n)$ – pixel brightness of the original frame with numbers *m*, *n*;

$m=\overline{0,N_{crx}-1}$ – pixel number of the frame fragment along the abscissa;

$n=\overline{0,N_{cry}-1}$ – number of pixels of a frame fragment along the ordinate;

$k=\overline{0,M_x-1}$ – frame fragment number along the abscissa;

$l=\overline{0,M_y-1}$ – frame fragment number along the ordinate;

M_x, M_y – number of frame fragments along the abscissa and ordinate;

N_{CDDx}×N_{CDDy} – size of the original frame along the abscissa and ordinate;

N_{crx}×N_{cry} – size of the frame fragment along the abscissa and ordinate.

After introducing a formal connection between the pixel values of the original astronomical frame and the pixel values of its fragment, it is necessary to determine the number of necessary fragments to cover the entire original astronomical frame.

5. 2. Determining the number of fragments in an astronomical frame

To devise a procedure for fragmenting the initial astronomical frame A_{in}, it is necessary to determine the number of fragments along the abscissa axis M_x and ordinate axis M_y, into which the frame will be divided.

It is assumed that the size of the original frame is N_{CDDx}×N_{CDDy}, and the selected fragment size is N_{crx}×N_{cry} along the abscissa and ordinate axes, respectively.

To solve this problem, it is proposed to determine the number of fragments along the abscissa M_x and ordinate M_y axes using the following expressions, respectively:

$$M_x=\begin{cases}\dfrac{N_{CDDx}}{N_{crx}}, \text{at } \dfrac{N_{CDDx}}{N_{crx}}=E\left[\dfrac{N_{CDDx}}{N_{crx}}\right],\\ E\left[\dfrac{N_{CDDx}}{N_{crx}}\right]+1, \text{at } \dfrac{N_{CDDx}}{N_{crx}}\neq E\left[\dfrac{N_{CDDx}}{N_{crx}}\right],\end{cases} \quad (2)$$

$$M_y=\begin{cases}\dfrac{N_{CDDy}}{N_{cry}}, \text{at } \dfrac{N_{CDDy}}{N_{cry}}=E\left[\dfrac{N_{CDDy}}{N_{cry}}\right],\\ E\left[\dfrac{N_{CDDy}}{N_{cry}}\right]+1, \text{at } \dfrac{N_{CDDy}}{N_{cry}}\neq E\left[\dfrac{N_{CDDy}}{N_{cry}}\right],\end{cases} \quad (3)$$

where $E[*]$ is the operation of obtaining an integer from a function argument.

In this case, frame fragments with numbers $k=M_{x-1}$ and $l=M_{y-1}$ are supplemented to sizes $N_{crx}\times N_{cry}$ with the pixel values of previous fragments.

It is also necessary to take into account the peculiarities of the formation of initial astronomical frames of different sizes, which must be divided into fragments.

In the case when the frame size along the x-axis is not equal to the integer number of frame fragments along it, the last frame fragments (with number $k=M_{x-1}$) are supplemented with the pixel values of the previous frame fragments. In this case, the pixel values of the original frame A_{in} and the pixel values of the kl-th fragment A_{crkl} are related as follows:

$$A_{crkl}(m, n)=A_{in}(N_{CCDx}-N_{crx}+m+1, n+N_{cry}l), \quad (4)$$

where $m=\overline{0,N_{crx}-1}$, $n=\overline{0,N_{cry}-1}$, $l=\overline{0,M_y-1}$, $k=M_x-1$.

When the ordinate size of the frame is not equal to the integer number of fragments, the last fragments of the frame (with number $l=M_{y-1}$) are additionally determined by the pixel values of the previous fragments. In this case, the pixel values of the original frame A_{in} and the pixel values of the kl-th fragment A_{crkl} are related by the expression:

$$A_{crkl}(m, n)=A_{in}(m+N_{crx}k, N_{CCDy}-N_{cry}+n+1), \quad (5)$$

where $m=\overline{0,N_{crx}-1}$, $n=\overline{0,N_{cry}-1}$, $k=\overline{0,M_x-1}$, $l=M_y-1$.

In the case when the frame size is not equal to the integer number of frame fragments along two axes, then the frame fragment with numbers $k=M_{x-1}$ and $l=M_{y-1}$, located in the upper right corner of the frame, is supplemented with the pixel values of the previous fragments. In this case, the pixel values of the original frame A_{in} and the pixel values of the kl-th fragment A_{crkl} are related as follows:

$$A_{crkl}(m, n)=A_{in}(N_{CCDx}-N_{crx}+m+1, N_{CCDy}-N_{cry}+n+1), \quad (6)$$

where $m=\overline{0,N_{crx}-1}$, $n=\overline{0,N_{cry}-1}$, $k=M_x-1$, $l=M_y-1$.

The potential conditions described above must be taken into account during the development of an algorithm for the procedure for fragmenting astronomical frames.

5. 3. Development of an algorithm for the procedure for fragmenting astronomical frames

The proposed algorithm for the procedure for fragmenting astronomical frames includes the following sequence of operations:

1. Obtain the dimensions $N_{CDDx}\times N_{CDDy}$ of the original astronomical frame in pixels.
2. Select the default fragment size $N_{crx}\times N_{cry}$, which depends on the allowable amount of memory used for frequency filtering.
3. Determine the number of fragments of the astronomical frame along the abscissa (2) and ordinate (3) axes.
4. Form a discrete spectrum of the original astronomical frame S_{in} using direct DFT of the original astronomical frame A_{in}:

$$S_{in}(u,v)=\sum_{m=0}^{N_{CCDx}-1}\sum_{n=0}^{N_{CCDy}-1}A_{in}(m,n)\exp\left(\begin{matrix}-i\dfrac{2\pi}{N_{CCDx}}mu-\\ -i\dfrac{2\pi}{N_{CCDy}}nv\end{matrix}\right), \quad (7)$$

where $S_{in}(u, v)$is the uv-th harmonic of the discrete spectrum of the source frame;

$A_{in}(m, n)$ – brightness of the mnth pixel of the source frame.

5. Introduce a formal connection (1) between the pixel values of the original frame A_{in} and the pixel values of its kl-th fragment A_{crkl}.
6. Take into account the peculiarities of the formation of fragments depending on the size of the original frame (4) to (6).
7. Select the transfer function H_{Ghp} of the high-pass filter [36] that will be used during the study.
8. Perform a high-pass filtering procedure for each calculated fragment.

5. 4. Verification of the procedure for fragmenting astronomical frames

To verify the devised procedure for fragmenting astronomical frames, testing was carried out on a series of frames obtained from various telescopes. The devised procedure for brightness alignment of the background frame was tested using the following high-pass filters: ideal, Butterworth, and Gaussian.

To confirm the hypothesis, processing time and RAM utilization were assessed during high-pass filtering using the devised fragmentation procedure and without it.

It is known that the optimal implementation of high-pass filter procedures, for example in the C++ programming language, uses a different number of arrays to store data in memory [32]. For example, an ideal filter is 6 arrays, a Butterworth filter is 7 arrays, and a Gaussian filter is 9 arrays. Source images are 64-bit. Thus, to process a frame with a resolution of 512×512 pixels with an ideal filter, you need to allocate only 12 MB of RAM. But to process a frame with a resolution of 4008×2672 pixels with a Gaussian filter, you already need as much as 735 MB. More detailed information is given in Table 1.

Table 1

Required volume of RAM for processing frames

High Pass Filter	Frame width, pixels	Frame height, pix.	Number of arrays	Memory, MB
Gaussian filter	512	512	9	18
Butterworth filter			7	14
Ideal filter			6	12
Gaussian filter	768	512	9	27
Butterworth filter			7	21
Ideal filter			6	18
Gaussian filter	3,056	3,056	9	641
Butterworth filter			7	498
Ideal filter			6	427
Gaussian filter	4,008	2,672	9	735
Butterworth filter			7	572
Ideal filter			6	490

The average processing time of the entire original astronomical frame was obtained for each high-pass filter used. For example, for a source frame of 512×512 pixels, the average time for an ideal filter is 0.32 seconds, for a Butterworth filter it is 0.36 seconds, and for a Gaussian filter it is 0.41 seconds.

A multiprocessor system on a server was considered, which uses different numbers of cores to process entire raw astronomical frames. Detailed information about the processing time of the initial series of 50 frames with a resolution of 512×512 pixels is given in Table 2.

Table 2

Processing time for entire frames on a multiprocessor system

High Pass Filter	Number of threads	Total memory, MB	Processing time, s
Gaussian filter	4	72	5.1
Butterworth filter		56	4.5
Ideal filter		48	4.0
Gaussian filter	6	108	3.4
Butterworth filter		84	3.0
Ideal filter		72	2.7
Gaussian filter	8	144	2.5
Butterworth filter		112	2.2
Ideal filter		96	2.1
Gaussian filter	10	180	2.0
Butterworth filter		140	1.8
Ideal filter		120	1.6

Now consider the same multiprocessor system on a server to process an initial series of 100 frames with a resolution of 4008×2672 pixels. The average processing time for such a frame for an ideal filter is 3.1 seconds, for a Butterworth filter – 3.7 seconds, and for a Gaussian filter – 4.2 seconds. Detailed information about processing time is given in Table 3.

Table 3

Processing time for entire frames on a multiprocessor system

High Pass Filter	Number of threads	Total memory, MB	Processing time, s
Gaussian filter	4	2,940	105.1
Butterworth filter		2,288	92.5
Ideal filter		1,960	77.5
Gaussian filter	6	4,410	70.2
Butterworth filter		3,432	61.7
Ideal filter		2,940	51.7
Gaussian filter	8	5,880	52.5
Butterworth filter		4,576	46.3
Ideal filter		3,920	38.8
Gaussian filter	10	7,350	42.1
Butterworth filter		5,720	37.3
Ideal filter		4,900	31.2

Using the devised frame fragmentation procedure, the following number of fragments was obtained for each resolution (Table 4). The region of 512×512 pixels was chosen as the fragment size.

The calculated values of the total memory and the time for complete processing of 100 frames of different resolutions in a multiprocessor system using the devised fragmentation procedure are given in Table 5.

All our research results in Tables 1–3, 5 have averaged values obtained on a sample of 10 iterations for each high-pass filter, each frame, and each fragment.

Given in Table 5, the results of the study indicate successful verification of the devised procedure and confirmation of the working hypothesis put forward. Performing fragmenting reduces RAM allocation and usage during high-pass filtering by 7 to 10 times.

Table 4

Required number of fragments to process frames

Frame width, pixels	Frame height, pixels	Number of fragments
512	512	1
768	512	2
3,056	3,056	36
4,008	2,672	40

Table 5

Processing time for frame fragments in a multiprocessor system

High Pass Filter	Number of threads	Number of fragments	Total memory, MB	Frame processing time, s	Total processing time, s
Gaussian filter	4	1	72	0.10	2.56
Butterworth filter			56	0.09	2.25
Ideal filter			48	0.08	2.03
Gaussian filter	6	2	108	0.14	2.28
Butterworth filter			84	0.12	2.03
Ideal filter			72	0.11	1.78
Gaussian filter	8	36	144	1.85	23.06
Butterworth filter			112	1.62	20.25
Ideal filter			96	1.44	18.02
Gaussian filter	10	40	180	1.64	16.40
Butterworth filter			140	1.44	14.40
Ideal filter			120	1.28	12.80

6. Discussion of results of investigating the procedure for fragmenting astronomical frames

Within the framework of the CoLiTec project and Lemur software (Ukraine), a study was carried out on the procedure for fragmenting astronomical frames for subsequent use during high-frequency filtering.

Solving the problem of introducing a formal connection between the pixel values of the original astronomical frame and the pixel values of its fragment made it possible to generate arrays of pixels for each fragment of the original frame for submitting them to the filtering procedure. The following high-frequency filters were chosen: ideal, Butterworth, and Gaussian.

Owing to solving the problem of determining the number of necessary fragments to cover the entire initial astronomical frame, two-dimensional arrays of pixels along the abscissa and ordinate axes were formed. The generated two-dimensional arrays for each frame fragment were submitted to high-frequency filtering procedures for three types of filters. Each of these filters was tested in the devised procedure and analyzed. During the study, different multiprocessor systems with 4, 6, 8, 10 processing threads were used. The required volume of RAM was obtained for each high-frequency filter under consideration for different frame sizes (Table 1). For example, to process a frame of 512×512 pixels you need from 12 to 18 MB, but for a frame of 4008×2672 pixels from 490 to 735 MB depending on the selected filter. The processing time of series of frames consisting of 50 and 100 frames of different resolutions was also obtained (Tables 2, 3). For example, to process a series of 50 frames of 512×512 pixels it takes from 1.6 to 5.1 seconds, but for a series of 100 frames of 4008×2672 pixels from 31.2 to 105.1 seconds depending on the selected filter and the number processing threads.

The study showed that the use of the devised procedure reduces RAM utilization by 7–10 times (Table 5). For example, to process a series of 100 frames of 4008×2672 pixels in 10 threads without the devised procedure with a Gaussian filter, 7.35 GB were simultaneously required, while with a devised procedure only 180 MB.

In addition, the speed of the highest frequency filtering was increased by 2–3 times. For example, to process a series of 100 frames of 3056×3056 pixels in 8 threads without the devised procedure with a Butterworth filter, it took 46.3 seconds, while with a devised procedure and division into 36 fragments it took only 20.25 seconds.

Statistical and full-scale modeling [37] have proven that this also significantly affects the speed of performing a number of tasks of brightness equalization, accumulation, and data acquisition [38]. It also has a positive impact on computer vision methods [39, 40] using supercomputers [41].

The limitations of this study are the computing power of the equipment and the number of cores in multiprocessor systems. Another possible limitation may be the set requirements for the speed of simultaneous frame processing during parallel processing from different telescopes. However, these requirements directly depend on the power of multiprocessor systems themselves. The issue of security [42] of frames, namely the encryption of input data, is also important. In this case, an additional decryption algorithm will be required.

A drawback of the study is that the proposed fragmentation procedure can only be used after the astronomical frame has been completely formed from the telescope. There are frame saving systems that gradually save parts of frames during online processing in the case of very large frame sizes, and access to the file is blocked.

It would be advisable to focus further research on the simultaneous application of the devised procedure during training and calibration service frames, and not only to the initial target frames. This will be necessary to accumulate and obtain data [43] for further identification of SSO with astronomical catalogs. To this end, it will be necessary to design the correct processing pipeline, connection of astronomical catalogs and appropriate methods for fetching services/modules [44]. It is also necessary to evaluate the usefulness of the devised procedure for other mathematical models [45], which will be used in methods for recognizing objects and detecting their movement. To this end, one can use Wavelet analysis [46], machine learning, or the forecasting method [47] to calculate statistical indicators.

7. Conclusions

1. A formal connection has been introduced between the pixel values of the original frame and the pixel values of its fragment. The peculiarity of this connection is the fact that when dividing the source frame into fragments, it is necessary to have a clear array of pixels belonging to each fragment of the source frame. This will allow us to get rid of confusion of pixels, directions of the coordinate axes of each fragment, and also assemble the filtered frame from the fragments correctly.

2. For the procedure of fragmenting astronomical frames, the fragment size was chosen in accordance with the allowable amount of allocated RAM. Owing to the proposed expressions, an equal number of fragments of the astronomical frame along the abscissa and ordinate axes was determined. Various features were also taken into account when the frame size along the axes is not equal to the whole number of fragments. In this case, it was proposed to supplement the incomplete fragment with the pixel values of the previous frame fragment. Owing to this, the formal connection between the pixel values of the original frame and the pixel values of its fragment was clarified.

3. An algorithm for the procedure for fragmenting astronomical frames has been developed. The key point is to take into account the peculiarities of the formation of fragments depending on the size of the frame with the subsequent formation of the spectrum of the high-frequency component of the image based on the selected transfer function. This made it possible to pre-fragment the original image to perform high-pass filtering.

4. The devised procedure for fragmenting astronomical frames using various high-frequency filters was verified based on many series of frames of different sizes. We simulated different numbers of threads on multiprocessor systems and measured RAM usage and processing time. By taking into account the size features of the original astronomical frames, as well as clarifying the formal connection between the values of the pixels of the frame and its fragments, the use of the devised procedure reduces the RAM utilization by 7–10 times. This solves the first part of the identified problem. The second part is the speed of the filtering procedure itself. Verification of the devised procedure showed that when dividing the original frame into fragments, the speed of the highest frequency filtering was also increased by 2–3 times. Accordingly, the time for complete processing of each astronomical frame was reduced by 2–3 times. When processing series consisting of a large number of astronomical frames (up to 100), the devised procedure can significantly speed up the overall processing of the series.

Conflicts of interest

The authors declare that they have no conflicts of interest in relation to the current study, including financial, personal, authorship, or any other, that could affect the study and the results reported in this paper.

Funding

The study was conducted without financial support.

Data availability

All data are available in the main text of the manuscript.

Use of artificial intelligence

The authors confirm that they did not use artificial intelligence technologies when creating the current work.

References

1. Wheeler, L., Dotson, J., Aftosmis, M., Coates, A., Chomette, G., Mathias, D. (2024). Risk assessment for asteroid impact threat scenarios. Acta Astronautica, 216, 468–487. https://doi.org/10.1016/j.actaastro.2023.12.049
2. Khlamov, S. V., Savanevych, V. E., Briukhovetskyi, O. B., Pohorelov, A. V. (2016). CoLiTec software - detection of the near-zero apparent motion. Proceedings of the International Astronomical Union, 12 (S325), 349–352. https://doi.org/10.1017/s1743921316012539
3. Savanevych, V. E., Khlamov, S. V., Akhmetov, V. S., Briukhovetskyi, A. B., Vlasenko, V. P., Dikov, E. N. et al. (2022). CoLiTecVS software for the automated reduction of photometric observations in CCD-frames. Astronomy and Computing, 40, 100605. https://doi.org/10.1016/j.ascom.2022.100605
4. Khalil, M., Said, M., Osman, H., Ahmed, B., Ahmed, D., Younis, N. et al. (2019). Big data in astronomy: from evolution to revolution. International Journal of Advanced Astronomy, 7 (1), 11–14. https://doi.org/10.14419/ijaa.v7i1.18029
5. Adam, G. K., Kontaxis, P. A., Doulos, L. T., Madias, E.-N. D., Bouroussis, C. A., Topalis, F. V. (2019). Embedded Microcontroller with a CCD Camera as a Digital Lighting Control System. Electronics, 8 (1), 33. https://doi.org/10.3390/electronics8010033
6. Vavilova, I., Pakuliak, L., Babyk, I., Elyiv, A., Dobrycheva, D., Melnyk, O. (2020). Surveys, Catalogues, Databases, and Archives of Astronomical Data. Knowledge Discovery in Big Data from Astronomy and Earth Observation, 57–102. https://doi.org/10.1016/b978-0-12-819154-5.00015-1
7. Zhang, Y., Zhao, Y., Cui, C. (2002). Data mining and knowledge discovery in database of astronomy. Progress in Astronomy, 20 (4), 312–323.
8. Chalyi, S., Levykin, I., Biziuk, A., Vovk, A., Bogatov, I. (2020). Development of the technology for changing the sequence of access to shared resources of business processes for process management support. Eastern-European Journal of Enterprise Technologies, 2 (3 (104)), 22–29. https://doi.org/10.15587/1729-4061.2020.198527
9. Khlamov, S., Savanevych, V., Tabakova, I., Trunova, T. (2022). The astronomical object recognition and its near-zero motion detection in series of images by in situ modeling. 2022 29th International Conference on Systems, Signals and Image Processing (IWSSIP). https://doi.org/10.1109/iwssip55020.2022.9854475
10. Troianskyi, V., Kankiewicz, P., Oszkiewicz, D. (2023). Dynamical evolution of basaltic asteroids outside the Vesta family in the inner main belt. Astronomy & Astrophysics, 672, A97. https://doi.org/10.1051/0004-6361/202245678
11. Oszkiewicz, D., Troianskyi, V., Gal d, A., Hanu , J., urech, J., Wilawer, E. et al. (2023). Spins and shapes of basaltic asteroids and the missing mantle problem. Icarus, 397, 115520. https://doi.org/10.1016/j.icarus.2023.115520
12. Savanevych, V., Khlamov, S., Briukhovetskyi, O., Trunova, T., Tabakova, I. (2023). Mathematical Methods for an Accurate Navigation of the Robotic Telescopes. Mathematics, 11 (10), 2246. https://doi.org/10.3390/math11102246
13. Bellanger, M. (2024). Digital Signal Processing. Wiley. https://doi.org/10.1002/9781394182695
14. Savanevych, V., Khlamov, S., Vlasenko, V., Deineko, Z., Briukhovetskyi, O., Tabakova, I., Trunova, T. (2022). Formation of a typical form of an object image in a series of digital frames. Eastern-European Journal of Enterprise Technologies, 6 (2 (120)), 51–59. https://doi.org/10.15587/1729-4061.2022.266988
15. Klette, R. (2014). Concise Computer Vision. Springer London. https://doi.org/10.1007/978-1-4471-6320-6
16. Khlamov, S., Tabakova, I., Trunova, T. (2022). Recognition of the astronomical images using the Sobel filter. 2022 29th International Conference on Systems, Signals and Image Processing (IWSSIP). https://doi.org/10.1109/iwssip55020.2022.9854425
17. Bodyanskiy, Y., Popov, S., Brodetskyi, F., Chala, O. (2022). Adaptive Least-Squares Support Vector Machine and its Combined Learning-Selflearning in Image Recognition Task. 2022 IEEE 17th International Conference on Computer Sciences and Information Technologies (CSIT). https://doi.org/10.1109/csit56902.2022.10000518
18. Dhanalakshmi, R., Bhavani, N. P. G., Raju, S. S., Shaker Reddy, P. C., Mavaluru, D., Singh, D. P., Batu, A. (2022). Onboard Pointing Error Detection and Estimation of Observation Satellite Data Using Extended Kalman Filter. Computational Intelligence and Neuroscience, 2022, 1–8. https://doi.org/10.1155/2022/4340897
19. Savanevych, V., Akhmetov, V., Khlamov, S., Dikov, E., Briukhovetskyi, A., Vlasenko, V. et al. (2019). Selection of the Reference Stars for Astrometric Reduction of CCD-Frames. Advances in Intelligent Systems and Computing, 881–895. https://doi.org/10.1007/978-3-030-33695-0_57
20. Lösler, M., Eschelbach, C., Riepl, S. (2018). A modified approach for automated reference point determination of SLR and VLBI telescopes. Tm - Technisches Messen, 85 (10), 616–626. https://doi.org/10.1515/teme-2018-0053
21. Shan, W., Yi, Y., Qiu, J., Yin, A. (2019). Robust Median Filtering Forensics Using Image Deblocking and Filtered Residual Fusion. IEEE Access, 7, 17174–17183. https://doi.org/10.1109/access.2019.2894981
22. Hu, Z., Bodyanskiy, Y. V., Tyshchenko, O. K., Tkachov, V. M. (2017). Fuzzy Clustering Data Arrays with Omitted Observations. International Journal of Intelligent Systems and Applications, 9 (6), 24–32. https://doi.org/10.5815/ijisa.2017.06.03
23. Kirichenko, L., Saif, A., Radivilova, T. (2020). Generalized Approach to Analysis of Multifractal Properties from Short Time Series. International Journal of Advanced Computer Science and Applications, 11 (5). https://doi.org/10.14569/ijacsa.2020.0110527
24. Dadkhah, M., Lyashenko, V. V., Deineko, Z. V., Shamshirband, S., Jazi, M. D. (2019). Methodology of wavelet analysis in research of dynamics of phishing attacks. International Journal of Advanced Intelligence Paradigms, 12 (3/4), 220. https://doi.org/10.1504/ijaip.2019.098561

25. Kirichenko, L., Pichugina, O., Radivilova, T., Pavlenko, K. (2022). Application of Wavelet Transform for Machine Learning Classification of Time Series. Lecture Notes on Data Engineering and Communications Technologies, 547–563. https://doi.org/10.1007/978-3-031-16203-9_31
26. Khlamov, S., Vlasenko, V., Savanevych, V., Briukhovetskyi, O., Trunova, T., Chelombitko, V., Tabakova, I. (2022). Development of computational method for matched filtration with analytical profile of the blurred digital image. Eastern-European Journal of Enterprise Technologies, 5 (4 (119)), 24–32. https://doi.org/10.15587/1729-4061.2022.265309
27. Khlamov, S., Savanevych, V., Vlasenko, V., Briukhovetskyi, O., Trunova, T., Levykin, I. et al. (2023). Development of the matched filtration of a blurred digital image using its typical form. Eastern-European Journal of Enterprise Technologies, 1 (9 (121)), 62–71. https://doi.org/10.15587/1729-4061.2023.273674
28. Kirillov, A., Wu, Y., He, K., Girshick, R. (2020). PointRend: Image Segmentation As Rendering. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr42600.2020.00982
29. Minaee, S., Boykov, Y. Y., Porikli, F., Plaza, A. J., Kehtarnavaz, N., Terzopoulos, D. (2021). Image Segmentation Using Deep Learning: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1–1. https://doi.org/10.1109/tpami.2021.3059968
30. Kudzej, I., Savanevych, V. E., Briukhovetskyi, O. B., Khlamov, S. V., Pohorelov, A. V., Vlasenko, V. P. et al. (2019). CoLiTecVS – A new tool for the automated reduction of photometric observations. Astronomische Nachrichten, 340 (1-3), 68–70. https://doi.org/10.1002/asna.201913562
31. Troianskyi, V., Kashuba, V., Bazyey, O., Okhotko, H., Savanevych, V., Khlamov, S., Briukhovetskyi, A. (2023). First reported observation of asteroids 2017 AB8, 2017 QX33, and 2017 RV12. Contributions of the Astronomical Observatory Skalnaté Pleso, 53 (2). https://doi.org/10.31577/caosp.2023.53.2.5
32. Burger, W., Burge, M. J. (2022). Digital Image Processing. In Texts in Computer Science. Springer International Publishing. https://doi.org/10.1007/978-3-031-05744-1
33. Lemur software. CoLiTec. Available at: https://colitec.space/
34. Khlamov, S., Savanevych, V., Tabakova, I., Kartashov, V., Trunova, T., Kolendovska, M. (2024). Machine Vision for Astronomical Images using The Modern Image Processing Algorithms Implemented in the CoLiTec Software. Measurements and Instrumentation for Machine Vision, 269–310. https://doi.org/10.1201/9781003343783-12
35. Dougherty, E. R. (2020). Digital Image Processing Methods. CRC Press, 504. https://doi.org/10.1201/9781003067054
36. Gonzalez, R., Woods, R. (2018). Digital image processing. Pearson. Available at: https://dl.icdst.org/pdfs/files4/01c56e081202b62bd7d3b4f8545775fb.pdf
37. Shvedun, V. O., Khlamov, S. V. (2016). Statistical modeling for determination of perspective number of advertising legislation violations. Actual Problems of Economics, 184 (10), 389–396.
38. Perova, I., Brazhnykova, Y., Miroshnychenko, N., Bodyanskiy, Y. (2020). Information Technology for Medical Data Stream Mining. 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET). https://doi.org/10.1109/tcset49122.2020.235399
39. Ulrich, M., Steger, C., Wiedemann, C. (2018). Machine vision algorithms and applications. John Wiley & Sons, 516.
40. Khlamov, S., Tabakova, I., Trunova, T., Deineko, Z. (2022). Machine Vision for Astronomical Images Using the Canny Edge Detector. CEUR Workshop Proceedings, 3384, 1–10.
41. Ruban, I., Martovytskyi, V., Lukova-Chuiko, N. (2016). Designing a monitoring model for cluster super–computers. Eastern-European Journal of Enterprise Technologies, 6 (2 (84)), 32–37. https://doi.org/10.15587/1729-4061.2016.85433
42. Buslov, P., Shvedun, V., Streltsov, V. (2018). Modern Tendencies of Data Protection in the Corporate Systems of Information Consolidation. 2018 International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T). https://doi.org/10.1109/infocommst.2018.8632089
43. Cavuoti, S., Brescia, M., Longo, G. (2012). Data mining and knowledge discovery resources for astronomy in the web 2.0 age. Software and Cyberinfrastructure for Astronomy II. https://doi.org/10.1117/12.925321
44. Petrychenko, A., Levykin, I., Iuriev, I. (2021). Improving a method for selecting information technology services. Eastern-European Journal of Enterprise Technologies, 2 (2 (110)), 32–43. https://doi.org/10.15587/1729-4061.2021.229983
45. Grebennik, I., Chorna, O., Urniaieva, I. (2022). Distribution of Permutations with Different Cyclic Structure in Mathematical Models of Transportation Problems. 2022 12th International Conference on Advanced Computer Information Technologies (ACIT). https://doi.org/10.1109/acit54803.2022.9913183
46. Baranova, V., Zeleniy, O., Deineko, Z., Bielcheva, G., Lyashenko, V. (2019). Wavelet Coherence as a Tool for Studying of Economic Dynamics in Infocommunication Systems. 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T). https://doi.org/10.1109/picst47496.2019.9061301
47. Dombrovska, S., Shvedun, V., Streltsov, V., Husarov, K. (2018). The prospects of integration of the advertising market of Ukraine into the global advertising business. Problems and Perspectives in Management, 16 (2), 321–330. https://doi.org/10.21511/ppm.16(2).2018.29