Journal of Al-Qadisiyah for Computer Science and Mathematics Vol.12(4) 2020 , pp Comp. 61–71 61

Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

The Numerical Implementation of the Image Filtering Method with Computations in the Residue Number System

Rusul Jabbar Alsaedi [a]

[a] *Al-Qasim University - Al-Qasim - Babylon – Iraq* , Email: *rusul@uoqasim.edu.iq*

ARTICLE INFO

Article history:
Received: 05 /06/2020
Rrevised form: 18 /06/2020
Accepted : 28 /06/2020
Available online: 11 /01/2021

Keywords:
Digital image processing, Digital filter, Laplace filter, Residue Number System (RNS), Module set.

ABSTRACT

The use of the Residue Number System (RNS) instead of the known binary arithmetic proposes, it is substantiated that the application of the non-position RNS will allow increasing the speed and reducing power consumption of digital image processing applications. Analyzed known schemes, based on which the process of image filtration occurs. The inefficiency of the application of known schemes is substantiated, connected with the possibility of errors due to the overflow of the dynamic range of the RNS. A new scheme of the filtration transformation process is given, which allows obtaining correct results of digital processing, independent of the size of the digital image being processed and the filter mask. The obtained results can be used in the construction of specialized digital image processing systems with low power consumption and high speed of operation.

MSC : 30C45 , 30C50

DOI : https://doi.org/10.29304/jqcm.2020.12.4.731

1. Introduction

Digital imaging has a wide range of applications in various fields of science and technology, such as machine vision, medicine, satellite communications, military and security affairs, automatic control of agricultural products [1], graphic and multimedia art [2], and etc. Developed digital image processing applications that are used in these areas should have, on the one hand, simplicity of hardware implementation, low power consumption of the system, high speed of information processing and a high level of security [3], and on the other hand, use mathematical processing methods that will improve the visual quality of the image that is obtained during processing [4].Since digital filters are the main tool for digital processing, an active search is underway for new effective digital image processing methods based on various numerical methods and algorithms [5]. The use of such methods will reduce

Corresponding author *Rusul Jabbar Alsaedi*

Email address: *rusul@uoqasim.edu.iq*

Communicated by : *Alaa Hussein Hamadi*

power consumption and increase the speed of data processing in digital image processing applications. The system of residual classes (RNS) is a promising tool in solving the tasks. Due to the property of parallel execution of operations, RNS can be effectively used in applications with a predominant share of addition, subtraction, and multiplication [6]. One of such applications is digital image processing, in particular digital filtering [4]. This article will show a numerical implementation of the filtering process on a specific image fragment. Known circuits of converters used in processing are considered, and a new one is proposed, which allows obtaining correct processing results.

NOMENCLATURE

Aradius of

Bposition of

Cfurther nomenclature continues down the page inside the text box

2. Materials and Methods

2.1 Introduction to Digital Image Processing

Digital filtering determines which cases digital filter. Speaking about the process of digital filtering (processing) of images, it should be noted that in this case, the digital filter allows you to apply various effects to the image; for example, blur, sharpness, deformation, noise, and etc. In this case, the digital filter is understood as a mathematical image processing algorithm. A large group of digital filters has the same calculation algorithm, but the effect of the filter on the image depends on the coefficients used in the algorithm [5]. Most often, in the process of the digital image processing, filters with a finite impulse response are used based on the theory of linear systems and the use of two-dimensional convolutions. Convolution is a way of representing any vector; values are scalar values. As applied to image processing, a vector value is the color of a group of pixels, and a scalar value obtained from convolution is the color of a pixel obtained by applying an effect to the original image. Image processing using such filters is described by the following formula [2].

$$C_{new}[i][j] = \sum_{k=0}^{n}\sum_{l=0}^{n} a_{k,l} C_{old}\left[i-\frac{m}{2}\right]\left[j-\frac{n}{2}\right] \quad (2.1.1)$$

Where m and n - are the filter constants that specify the two-dimensional filter size;
a - is the filter coefficient that determines the effect that the first imposes a filter.

Therefore, to apply a filter to the image, it is necessary to carry out calculations for each of the pixel values. In this case, each of the pixels is considered together with the matrix, the central element of which it is. Multiply the corresponding values two matrices and their sum is assigned to the point under consideration. In this article, we will consider an image in shades of gray. In this format, the image is a rectangular array of integers (pixels). The number of gray levels is integer powers of 2 that are a pixel represents brightness or darkness [2, 4]. In this way, the larger number encoding a pixel, the brighter the image at that point. In standard image processing applications, the pixel values of the image in gray scale are encoded with 8-bit numbers and are in the range [0 - 255] where 0 is black, and 255 is white. If as a result of image processing, a negative number is obtained, then it is replaced by 0 (black color). In the case of obtaining a number greater than 255, it is replaced by 255 (white). Fig.1 is a schematic representation of the principle of mapping numbers to the color of a pixel in gray scale.

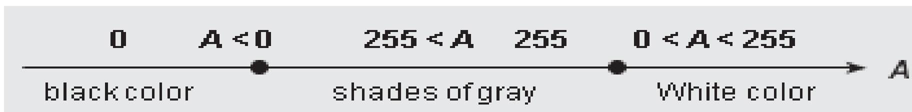

Fig. 1 - the dependence of the color of the pixel presented in shades of gray on the value of the code number.

As an illustrative example, the filter we look at the Laplace filter for sharpening, which we will also use in the future for numerical calculations. An example of a laplace filter mask is a matrix of the form.

$$[a, l] = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix} \qquad (2.1.2)$$

The filter mask is defined as the sum of the second derivatives and is calculated as the sum of the differences at the neighbors of the central pixel.

Applying the Laplace filter to a digital image allows you to increase the contrast of the image, increasing its sharpness. This type of filter is called high pass filters. The filter core has a value greater than 1 at the point (0, 0). The total amount of all values is 1. The effect of increasing sharpening is achieved due to the fact that the filter emphasizes the difference between the intensities of neighboring pixels, removing these intensities from each other [4]. Moreover, the sharpness effect will be the stronger, the greater the value of the central member of the nucleus. A sign of the use of such a filter is to obtain noticeable light and less noticeable dark halos on the image. Figure 2 shows the result of the sharpening filter. Figure 2b was shifted by 128 to improve the visual perception.

Fig. 2 – (a) image "girl" before processing; (b) the image "girl" after processing the filter Laplace for contouring.

2.2 Introduction to the System of Residual Classes

In RNS, numbers are represented in the basis of mutually prime numbers called modules *$\beta = \{m_1, ..., m_n\}$, $GCD(m_i, m_j) = 1$, for $i \neq j$*. The product of all modules RNS $M = \prod_{i=1}^{n} m_i$ is called the dynamic range of the system. Any integer $0 \leq X < M$ can be uniquely represented in the RNS in the form of a vector *$\{x_1, x_1, ..., x_n\}$, where $xi = | X | mi = X \bmod m_i$* [7].

In the case of using negative numbers, the number of relies on the following intervals:

$$-\frac{M-1}{2} \leq X \leq \frac{M-1}{2} \quad for\ odd\ M \qquad (2.2.1)$$

$$-\frac{M}{2} \leq X \leq \frac{M}{2} - 1 \ for\ even\ M, \qquad (2.2.2)$$

The recovery of the number X from the residuals *$\{x_1, x_2, ..., x_n\}$* is based on the Chinese Residue Theorem (RNS) [6].

$$X = \left| \sum_{i=0}^{n} \left| |M_{m_i}^{-1}|_{mi} X_{mi} \right|_{mi} M_i \right|_M \qquad (2.2.3)$$

where $M_i = M/m_i$

Item $|M_i^{-1}|_{m_i}$ means the multiplicative inverse element for M_i, modulo m_i. In the process of filtering images, we will use the Chinese Residue Theorem, as well as the Modified Chinese Residue Theorem. Using RNS in conjunction with LUT tables, you can perform operations on numbers with the least loss of information. The use of modified RNS will help reducing the dynamic range of the system and, as a result, increasing the efficiency of implementation.

3. Architectural Organization of the Image Filtering Process

The numerical organization of the filtering process is directly related to its structural scheme. Therefore, speaking of image filtering, a special place should be taken by the scheme of this process, which should include each process in stages. To perform operations in RNS, it is necessary to initially convert numbers from a positional number system to a system of residual classes; i.e., apply the direct conversion operation PSS-RNS [7]. Thus, the scheme on the basis of which this conversion should take place must first obtain the pixel values, then convert them to a representation of the RNS, and then perform the digital image processing, namely digital filtering. After passing through the described operation to the numbers represented in the RNS, the reverse operation should be applied to obtain information in the MSS. Proceeding from this, in [8], the authors present the image filtering operation as a process consisting of three stages. Each of these steps is described as follows.

1. Converting pixel values from a positional numbering system to a system of residual classes. This stage includes the solution of several tasks. Reading the pixel value of the input digital image in the form of decimal or binary numbers; selecting an appropriate set of RNS modules that satisfies the constraint imposed on the dynamic range of the system; creation of a reference LUT table for faster processing; assignment of new pixel values in accordance with the RNS obtained at this stage.
2. The operation of image filtering with calculations in the RNS.
3. The inverse transformation of the obtained pixel values from the RNS to MSS. The step includes getting values pixels as residues in the RNS; determination of the number of residues; transformation of each residue into its original form using LUT tables.

A special place in the described process of direct and inverse transformations from MSS to RNS and vice versa is assigned to viewing LUT tables since they increase the overall system performance. Tables are used during the transition of pixel values from the MSS to the RNS and vice versa, which naturally lead to a faster conversion of values. We show our generalized modified scheme, adhering to which we can obtain a processed digital image (Fig. 3).

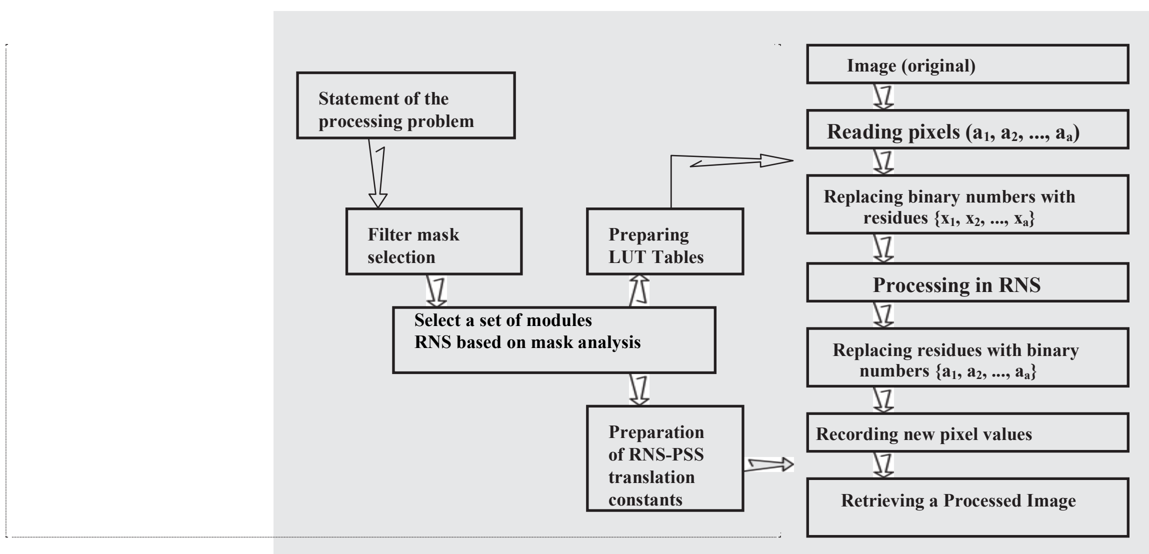

Fig. 3 - the proposed scheme of the image filtering process.

The distinctive feature of the proposed scheme is the fact that it takes into account the preparatory stage of filtering, during which the filter mask is selected and the set of RNS modules is selected, which occupies one of the important places in the image processing process. Moreover, these operations depend on the originally posed processing task. In [1], the choice of a set of RNS modules occurs after reading the pixel values of the images, which can lead to incorrect results of the system operation and image processing [9, 10].

4. Results and Discussion

4.1 Numerical Implementation of Filtering by Example Gray Scale Images

Let us consider in more detail the effect of the digital filter image on the example of the well-known filters of Privit, Sobel, Gauss, Laplacian. Filtering will take place in the spatial domain using a gray scale image. To represent one image, we will use a two-dimensional matrix of size M × N. The value of each element from this matrix shows the degree of brightness, while each element takes an 8-bit value, which can vary range from 0 to 255. In the MATLAB mathematical package, such a representation of the image is written as follows [11].

$$F(x,y)=\begin{bmatrix} F(1,1) & F(1,2) & \dots & F(1,N) \\ F(2,1) & F(2,2) & \cdots & F(2,N) \\ \vdots & \vdots & \vdots & \vdots \\ F(M,1) & F(M,2) & \dots & F(M,N) \end{bmatrix} \quad (4.1.1)$$

The combination of function and mask changes in gray levels, written in the form of formula (8) is called a filter. In order to apply a linear filter to the image array, it is necessary to multiply all the image coefficients by the corresponding adjacent elements from the filter (mask) and summarize all the obtained values [2]. In the case of filtering in the spatial domain, the mask is superimposed on the corresponding part of the image, moves on it, and the corresponding pixel values that coincided when applying the mask are used as calculated values [4]. Thus, after such an operation, a new image in shades of gray is obtained, calculated in accordance with the proposed mask. The mathematical interpretation of the described operation corresponds to the formula (7).

$$g(x,y)=\sum_{s=-a}^{a}\sum_{t=-b}^{b} W(s,t)f(x+s,y+t) \quad (4.1.2)$$

Where a and b - are positive numbers;

f (x, y) - is the function for the given filter;

w (s, t) - is a function for the values of the output image.

Assume 3 × 3 mask coefficients can be represented as:

$$\begin{bmatrix} w(-1,1) & w(-1,0) & w(-1,-1) \\ w(0,1) & w(0,0) & w(0,-1) \\ w(1,1) & w(1,0) & w(1,-1) \end{bmatrix} \quad (4.1.3)$$

The coefficients of pixel values have the following meanings.

$$\begin{bmatrix} f(x-1,y+1) & f(x-1,y) & f(x-1,y-1) \\ f(x,y+1) & f(x,y) & f(x,y-1) \\ f(x+1,y+1) & f(x+1,y) & f(x+1,y-1) \end{bmatrix} \quad (4.1.4)$$

We multiply the two presented matrixes of coefficients (8) and (9) with each other and summarize the obtained values [11].

$$\sum_{s=-1}^{1}\sum_{t=-1}^{1} W(s,t)f(x+s,y+t)$$

Thus, the filtering process can be divided into 3 stages.

1. The location of the specified mask on the image fragment.
2. Multiplication of filter coefficients by adjacent image coefficients.
3. Finding the total amount of the obtained values.

All the many filters that are used to obtain improved digital image quality can be divided into two categories: low-pass and high-pass filters. Processing a digital image by a low-pass filter that passes through a strip of low-frequency pixels and changes high-frequency pixels can result in hidden images. A high-pass filter that passes through high-frequency pixels and makes changes in the low-frequency band of pixels leads to noise at the edges of the image [5]. The most famous filters used to solve the indicated problems, which are also used to solve problems of increasing or decreasing image sharpness, highlighting a path, noise reduction, and etc. are the following filters Prewitt filters for selecting a path.

$$p_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, p_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Sobel filters to select a path.

$$p_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, p_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Because the image filtering operation according formula (1) requires a large number of arithmetic operations of addition and multiplication of pixel values with the corresponding mask, we can conclude that for operations related to digital image processing it is necessary to increase the speed of these operations. In addition, increasing the speed of the system will lead to lower power consumption of the digital application, suitable solution.

The problem is the use of non-positional number system. In addition to the high parallelism property that RNS possesses, the architecture of such a system can use combination schemes in its work, as well as lookup tables (LUT tables). These advantages, consisting in the development of a sufficiently high system speed and low power consumption, emphasize the advantages of using RNS in the applications of the digital image processing [8]. In [9, 10, 12, 13], the actual issue that arises in the digital image processing is described, namely the choice of the correct set of modules.

Therefore, based on the results obtained earlier, we apply the set of modules that we proposed in [10] for solving this problem. The proposed set of modules has the form {5, 7, 9, 16}.

4.2 Modeling a Numerical Method

Let us show the operation of this set of modules using a specific image as an example. As an input image, we will consider the image of a “girl” in shades of gray. The input pixel values in this image are presented as:

$$\begin{bmatrix} 252 & 254 & 250 & 250 \\ 250 & 253 & 254 & 251 \\ 254 & 253 & 252 & 251 \\ 254 & 252 & 253 & 254 \end{bmatrix} \qquad (4.2.1)$$

Convert all available pixel values of the image (10) and filter mask values (2) to the values presented in the RNS. As the selected set of modules, we will use the set {5, 7, 9, 16}. The dynamic range of the set M = 5 · 7 · 9 · 16 = 5040 is suitable for the digital images presented in the shades of gray. Then, for the selected pixel values, we get:

$$\begin{bmatrix} \{2,0,0,12\} & \{4,2,2,14\} & \{0,5,7,10\} & \{0,5,7,10\} \\ \{0,5,7,10\} & \{3,1,1,13\} & \{4,2,2,14\} & \{1,6,8,11\} \\ \{4,2,2,14\} & \{3,1,1,13\} & \{2,0,0,12\} & \{1,6,8,11\} \\ \{4,2,2,14\} & \{2,0,0,12\} & \{3,1,1,13\} & \{4,2,2,14\} \end{bmatrix}$$

The filter mask values are rewritten in the form:

$$\begin{bmatrix} \{4,6,8,15\} & \{4,6,8,15\} & \{4,6,8,15\} \\ \{4,6,8,15\} & \{4,2,0,9\} & \{4,6,8,15\} \\ \{4,6,8,15\} & \{4,6,8,15\} & \{4,6,8,15\} \end{bmatrix}$$

Considering the use of the LUT table in the inverse transformation during the transition from the system of residual classes to the positional number system needs to build a lookup table for the pixel values of the digital images (Table 1).

Table 1 - The pixel values of the digital images in the MSS and RNS.

The pixel values in the set of modules {5,7,9,16}	Pixel values in a positional number system
{0,5,7,10}	250
{1,6,8,11}	251
{2,0,0,12}	252
{3,1,1,13}	253
{4,2,2,14}	254

Let us consider in more detail the process of filtering values Images. As the processed data array, we will use an array of the same 3 × 3 size as the filter mask. When multiplying the image values by the corresponding filter values, we get:

$$\begin{bmatrix} \{2,0,0,12\} & \{4,2,2,14\} & \{0,5,7,10\} \\ \{0,5,7,10\} & \{3,1,1,13\} & \{4,2,2,14\} \\ \{4,2,2,14\} & \{3,1,1,13\} & \{2,0,0,12\} \end{bmatrix} \times \begin{bmatrix} \{4,6,8,15\} & \{4,6,8,15\} & \{4,6,8,15\} \\ \{4,6,8,15\} & \{4,2,0,9\} & \{4,6,8,15\} \\ \{4,6,8,15\} & \{4,6,8,15\} & \{4,6,8,15\} \end{bmatrix} = \begin{bmatrix} \{8,0,0,180\} & \{16,12,16,210\} & \{0,30,56,150\} \\ \{0,30,56,150\} & \{12,2,0,117\} & \{16,12,16,210\} \\ \{16,12,16,210\} & \{12,6,8,195\} & \{8,0,0,180\} \end{bmatrix}$$

Next, we add all the corresponding coefficients to each other, after which we get the remainder of each modulo. The obtained result {3, 6, 6, 2} corresponds to the value of the first pixel of the image represented in the RNS. We perform a similar operation for the remaining pixel values, after we obtain:

$$\begin{bmatrix} \{3,6,6,2\} & \{2,6,2,0\} \\ \{0,3,3,15\} & \{2,2,4,7\} \end{bmatrix} \tag{4.2.2}$$

To translate the found values from the view to the RNS back to the MSS, apply CTO. Since the found dynamic range of the RNS is $P = 5 \cdot 7 \cdot 9 \cdot 16 = 5040$, then by the formula $P = P / P_i$, we find the ranges for each of the four modules {5, 7, 9, 16} and the remainder in the division.

$P_i = 1008, \quad 1008 \bmod 5 = 3$

$P_2 = 720, \quad 720 \bmod 7 = 6$

$P_3 = 560, \quad 560 \bmod 9 = 6$

$P_4 = 315, \quad 315 \bmod 16 = 1$

Applying the inverse operation, we obtain the following values:

$3 \cdot x_1 = 1 \bmod 5 \to x_1 = 2$

$6 \cdot x_2 = 1 \bmod 7 \to x_2 = 6$

$2 \cdot x_3 = 1 \bmod 9 \to x_3 = 5$

$11 \cdot x_4 = 1 \bmod 16 \to x_4 = 3$

Find the B_i array:

$B_i = P_i \cdot x_i = 1008 \cdot 2 = 2016;$

$B_2 = P_2 \cdot x_2 = 720 \cdot 6 = 4320;$

$B_3 = P_3 \cdot x_3 = 560 \cdot 5 = 2800;$

$B_4 = P_4 \cdot x_4 = 315 \cdot 3 = 945;$

Then the obtained pixel value after the reverse transformation will be written as:

$A = 3 \cdot 2016 + 6 \cdot 4320 + 6 \cdot 2800 + 2 \cdot 945 = 50658,$

$50658 \bmod 5040 = 258.$

After performing similar operations for each of the values (11), get the converted pixel values from the RNS to the MSS:

$$\begin{bmatrix} 258 & 272 \\ 255 & 247 \end{bmatrix}$$

All calculations were performed using the package MATLAB application programs. The simulation results are obtained histograms for the digital images for the input image and the image obtained after filtering (Figures 4 and 5, respectively). The horizontal axis of each graph represents the values of the brightness levels r_k, and the vertical axis represents the histogram values $h(r_k) = n_k$. By the way, these graphs express the dependences $h(r_k) = n_k$ *on* r_k *or* $p(r_k) = n_k / n$ on r_k (if the histogram values are normalized) [2]. It can be seen from these graphs that non-zero histogram levels cover a wide part of the brightness range, and also that the distribution of pixel values is not too different from the uniform, with the exception of the number of peaks that raise above the remaining values. Intuitively, we can conclude that an image whose distribution of element values is close to uniform and occupies the entire range of possible brightness values will look high contrast and will contain a large number of mid tones [8]. Thus, based only on the information contained in the histogram of the source image, it is possible to build a conversion function that will automatically achieve this effect.

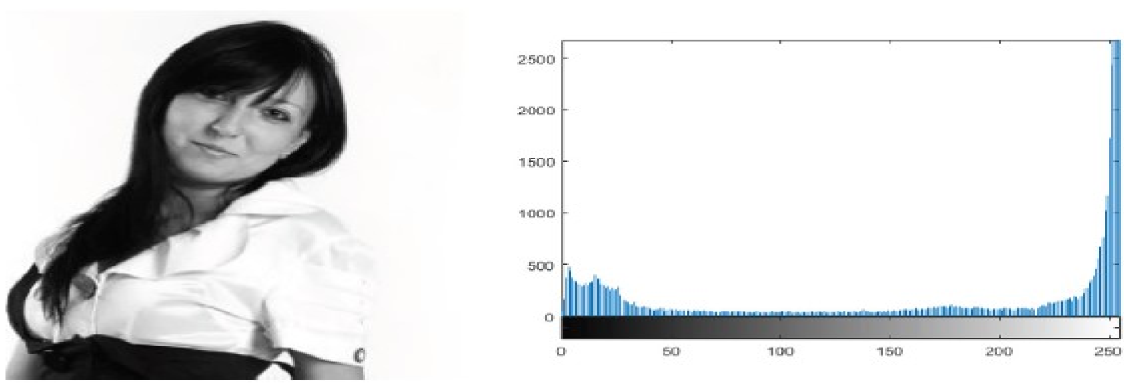

Fig. 4 - the input image “girl” before processing by the filter and its histogram of image values.

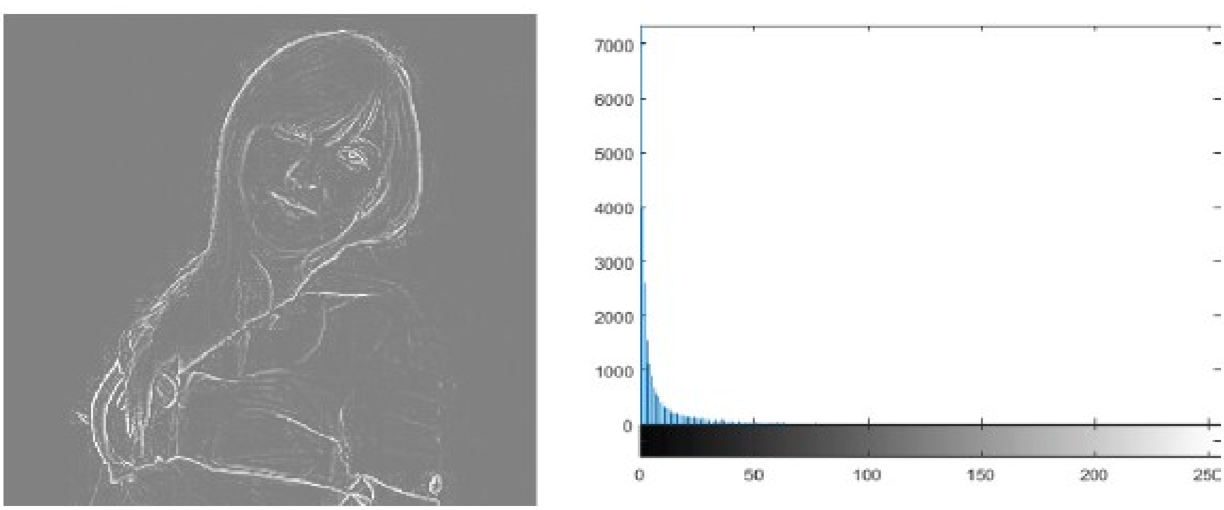

Fig. 5 - the “girl” image processed by the filter and its histogram of image values.

5. Conclusions

The article explores the issue of the numerical implementation of the method Yes digital image filtering using calculations in the system of residual classes. It is shown that the proposed numerical method used in the digital image filtering allows us to reduce the computational costs that arise when performing arithmetic operations of addition and multiplication. The effectiveness of the method is achieved through the use of a system of residual classes, which is able to parallelize the calculation process. The developed architectural organization of the digital filtering process allows you to get the correct processing result, regardless of the image size and filter mask. Benefits the schemes are due to the fact that it takes into account the preparatory stage of filtering, during which the selection of the set of RNS modules and the filter mask occur right up to the filtering process itself.

References

[1] A.I.Galyshkin, N.C. Tomashevih, and E.I. Ryabcev,"Neurocomputers for Image Processing," in Kn. 25, 2007, 74–109 c.

[2] R. Gonsales, and R. Vyds, "Digital Image Processing," in M.: Texnosfera, 2012, 1104 c.

[3] V. Damyanovski, "Bible Guarding Television," in Moskva, 2003, 344 c.

[4] S. U. Jeltov, "Image Processing and Analysis in Computer Vision Problems," in M.: Fizmatkniga, 2010, 672 c.

[5] M. V. Belodedov, "Methods for Designing Digital Filters," in Volgograd, 2004, 60 c.

[6] N.I. Chervyakov, P.A. Shaposhnikov, and A.N. Makoxa, "Neurocomputers in the Residual Classes," in M.: Radiotexnika, 2003, 272 c.

[7] I.Y. Akyshskii, and D.I. Udickii, "Machine Arithmetic in the Residual Classes," in M.: Sovetskoe radio, 1968, 440 c.

[8] D.K. Taleshmekaeil, and A. Mousavi, "The Use of Residue Number System for Improving the Digital Image Processing," in Proc. of 10th International Conference on Signal Processing, 2010, pp. 775-780.

[9] N.I. Chervyakov, P.A. Lyaxov, and D.I. Kalita, "About the Choice of the Range of the Residue Number System for Digital Image Processing," in Infokommynikacionnye texnologii, 2016, T. 14. №2. C. 111–122.

[10] N.I. Chervyakov, P.A. Lyakhov, D.I. Kalita, and K.S. Shulzhenko, "Effect of RNS Dynamic Range on Grayscale Image Filtering," in XV International Symposium Problems of Redundancy in Information and Control Systems (REDUNDANCY), St. Petersburg, 2016, pp.

[11] M.I. Kuryachii, "Digital Signal Processing," in Tomsk. gos Universitet sisten upr. i radioelektronoki. Tomsk, 2009, 190 c.

[12] W. Wang, M.N. Swamy, M.O. Ahmad, and Y. Wang, "A High-Speed Residue-to-Binary Converter for Three-Moduli ($2k$, $2k$, $1,2k–1$ – 1) RNS and a Scheme for Its VLSI Implementation," in IEEE Trans. On Circuits and Systems-II: Analog and Digital Signal Processing, 2000, 47, No. 12. Pp. 1576–1581.

[13] W. Wang, M.N. Swamy, M.O. Ahmad, and Y. Wang, "A Study of the Residue-to-Binary Converters for the Three Moduli Sets," in IEEE Trans. Circuits Syst. I: Fundamental Theory Appl., 50: 235–243.