Date of publication MAR-31, 2024, date of current version MAR-22, 2024.
www.computingonline.net / computing@computingonline.net

Print ISSN 1727-6209
Online ISSN 2312-5381
DOI 10.47839/ijc.23.1.3444

Image Transmission in WMSN Based on Residue Number System

ANATOLIY SACHENKO[1,2], VASYL YATSKIV[3], JÜRGEN SIECK[4] AND JUN SU[5]
[1]Research Institute for Intelligent Computer Systems, West Ukrainian National University, Ternopil, 46020,Ukraine
[2]Department of Informatics, Kazimierz Pulaski University of Technology and Humanities in Radom, Radom, 26600, Radom, Poland
[3]Department of Cyber Security, West Ukrainian National University, Ternopil, 46020, Ukraine
[4]Hochschule für Technik und Wirtschaft Berlin FKI, H201, 12459 Berlin, Germany
[5]School of Computer Science Hubei University of Technology, Wuhan 430068, China

Corresponding author: Vasyl Yatskiv (e-mail: vy@wunu.edu.ua).

ABSTRACT The paper considers the speedy images processing in Wireless Multimedia Sensor Networks using the Residue Number System (RNS) and the method of arithmetic coding. The proposed method has a two-stage frame: firstly, the RNS transformation is run to divide the data and obtain residues, and secondly, the parallel compression of the resulting residues is provided by employing the arithmetic coding. Within the implementation of binary code transformation in RNS one, the hardware complexity for block conversion is evaluated for various modulo sets and the results are illustrated. Authors employed the arithmetic coding for residue compression to provide the optimum of compression degree in terms of entropy assessment as well as a reduction in image redundancy without loss of quality. A research algorithm is proposed to run an experiment presented by the residues carried out on test images and other types of files. As a result, an increase in the speed of image compression of about 2.5 times is achieved by processing the small data as well as providing the parallel operation of the compression residue units by RNS selected moduli. Finally, the existing and proposed methods are compared and it has been shown the last one provides a better compression ratio of more than twice.

KEYWORDS Wireless Multimedia Sensor Networks; Residue Number System; Image Transmission.

I. INTRODUCTION

A large number of devices (sensors) connected to the IoT, which generates a significant amount of data and there is a tendency to increase it even more, especially for the mass growing of multimedia data [1] – [3]. The image is a common carrier among multimedia data, including pixels with a high correlation to each other. Due to this fact, the increasing number of compression methods are observed last decade [4] – [7].

A selection of the compression method is closely related to the type of operating platform, one of which is Wireless Multimedia Sensor Network (WMSN). The last one is able to deliver multimedia data using cheap CMOS cameras and microphones. Nowadays, the WMSNs can find a broad application, in particular Industry 4.0, military, agriculture, intelligent transport systems and traffic surveillance, cybersecurity monitoring, medicine, smart cities, space investigation programs, smart grid.

In comparison with conventional Wireless Sensor Networks, WMSN has more problems because of their limited resources in memory, processing, bandwidth, and power consumption. Hence, additional requirements are imposed on the compression process.

Since existing techniques and technologies of compression cannot provide an appropriate capability, improving the image transmission and compression over WMSN is a hot issue [8], [9].

A contribution of the paper is a proposal for using the Residue Number System (RNS) transformation for dividing the data (obtaining residues) and parallel compressing the resulting residues by employing arithmetic coding. The novelty of this work is that the proposed method enables reducing the compressing time of data by providing a parallel processing of remainders.

The rest of this paper has been divided into four sections. Section 2 is dedicated to State-of-the-Arts, Section 3 describes the proposed method, and Section 4 contents the experimental results. Section 5 summarizes the obtained results.

II. RELATED WORK

Now various methods and techniques of compression are designed, and JPEG is widely employed in CMOS chips [11] – [13]. The main disadvantage of most compression algorithms is their sequential data processing, which leads to an increase in data delay time. This disadvantage is especially noticeable in WSN, given the limited hardware resources of nodes and the low bandwidth of communication channels [14] – [15].

In Reference [16], the Lempel-Ziv-Welch (LZW) compression algorithm was improved by using a system of residual classes. In the encoding process, decimal representations of ASCII codes are converted to residuals using the direct conversion of RNS. The received balances are transmitted by a communication channel in a secret order. Decoding is the restoration of a decimal representation using a process known as the inverse conversion of RNS. It is shown the proposed scheme works more efficiently than the traditional LZW compression algorithm and provides error correction by using redundant RNS modules.

In Reference [17], a new model of data compression of sensors with accumulation of RBM (Stacked RBM-AE) was developed, which consists of four standard limited Boltzmann machines (RBM). An energy optimization method has also been developed, which further reduces the energy consumption by trimming the model parameters. The proposed method provides an average value of the error of temperature recovery of 0.2815 C, and the energy consumption during data transmission can be reduced by 90%. However, this method refers to data compression with losses, and accordingly cannot be applied to data that are sensitive to losses.

In Reference [18], a new algorithm for encrypting and decrypting data using RNS with four modules and the Huffman algorithm was developed. Huffman's coding algorithm uses the probability or frequency of occurrence of characters or strings in a particular data set. Data compression is achieved through RNS, which uses the remainder of the numbers instead of the numbers themselves. In addition, the proposed scheme enables detecting and correcting the errors due to two redundant modules. From a theoretical point of view, the proposed scheme improves the RNS coding technique with three modules in terms of security and data volume.

In Reference [19], both arithmetic compression and Huffman compression, based on a hybrid combination of DWT-DCT methods, are presented. Simulation results show that arithmetic coding gets a higher compression ratio than Huffman coding, but lower PSNR values.

In Reference [20], the data compression algorithm with an error limitation mechanism based on artificial neural networks (ANN) in the form of an autoencoder (AE), which was further evaluated and compared with conventional approaches, is proposed. It is proved experimentally, the proposed algorithm has a smaller root mean square error (RMSE) and higher values of the coefficient of determination (R2).

In Reference [21], the latest developments in the design of high-energy-efficient WMSNs with QoS support are considered, in particular, the energy-efficient communication protocols with QoS support, including MAC protocols with focusing on their mechanisms of prioritization and service differentiation, as well as on protocols of discrete multipath routing.

Reference [11] presents an improved version of the LZW algorithm, which uses the characteristic samples of these sensors to reduce energy consumption by more than 1.5 times. The authors quantified the benefits of data compression for energy savings in WSN.

The performed analysis showed that most authors use coding in RNS to increase the reliability of data transmission. However, this coding is implemented sequentially and does not use the main advantage of parallel data processing in RNS [16] – [18], [22], [23].

Therefore, the goal of this paper is to develop a method of data transmission that would maximize the benefits of RNS, such as parallel data processing and processing of low-bit residues. That should increase the speed of image compression and other types of data.

III. MATERIALS AND METHODS

The proposed method has a two-stage frame. Firstly, the RNS transformation is run to divide the data (obtain residues) and secondly, the parallel compressing the resulting residues is provided by employing the arithmetic coding [27], [35], [36].

Residue Number System is determined by the selection of integers p_I $(i = 1, 2, \dots, k)$, called moduli [24]. If all moduli are pairwise mutually prime numbers, any integer X in the range $[0,\ P_k]$, where $P_k = \prod_{i=1}^{k} p_i$, can be presented by sequence of residues b_i:

$$X = (b_1,\ b_2, \dots, b_k), \tag{1}$$

where $b_i = X \pmod{p_i}, (i = 1, 2, \dots, k)$.

The b_i are the residues of the number X from dividing by p_i for $i = 1, 2, \dots, k$. In this case, $[0, P_k]$ is the working (allowed) range of X.

The integer X in the range can be restored from the k residues $(b_1, b_2, \dots, b_k)$ using the Chinese remainder theorem [24].

In order to detect and correct errors in the message submitted by the RNS, we have introduced additional moduli.

The excess RNS is obtained by adding a certain number $r = n - k$ of additional moduli $(p_{k+1},\ p_{k+2}, \dots, p_{k+r})$ to the previously selected modulo system. As a result, the excess RNS is formed as a code of n from positive two-way, mutually simple moduli.

Therefore, an integer X in the range $[0, P_n]$ is presented by the sequence of remainders in moduli $(p_1, p_2, \dots, p_n)$:

$$X = (b_1,\ b_2, \dots, b_n). \tag{2}$$

Hence, the interval $[0, P_k]$ is called the allowed (operating) range, and the interval $[0, P_k]$, formed from the additional moduli, is called the forbidden one.

The operating range represents the computing range of RNS, while the forbidden range is used to detect and correct errors [26], [27].

A. IMAGE REPRESENTATION IN RNS

Coding is generally refers to the process of converting input data into a form that is convenient for direct transmission, processing, and storage. In this paper, coding refers to the transformation of image pixels into a system of residual classes.

When moving to the representation of numbers in RNS, we obtain independent numbers of small bits (2 – 8 bits), which enable to increase the speed of running the arithmetic operations.

The color RGB image is presented as an array of colored pixels $M \times N \times 3$, in which M is the number of rows, N – the number of columns, and each pixel is a triplet corresponding to red, green, and blue color components. The most common version is the representation of each component in an 8-bit image. In such case, an RGB image has a depth of 24 bits [12], [28].

The proposed method is based on the transition from the representation of image pixels in a Binary Number System (BNS) to their representation in the system of residual classes. Since the pixels of the image take values in the range from 0 to 255, it is necessary to select in the RNS mutually simple moduli, which product will be larger than 255.

So, let us select the following sets of moduli: $\{p_1 = 3, p_2 = 7, p_3 = 13\}$, where $P = 3 \times 7 \times 13 = 273$ or $\{p_1 = 5, p_2 = 7, p_3 = 8\}$, $P = 5 \times 7 \times 8 = 280$. For further operations, let us select the moduli (5, 7, 8), because the residues of the supplied moduli in the binary code have the same digit capacity m=3.

A selection of moduli was carried out under two conditions: 1) the product of the moduli must be greater than the maximum value that is processed; 2) to simplify a further work with residues, it is advisable that their digit capacity was the same. Therefore, the values of P_1, P_2 and P_3 moduli are selected as $(5, 7, 8)$ because: (i) image processing will be performed by one byte, respectively, the product of the moduli must be greater than 256 (2^8); (ii) the digit capacity of residues in these moduli is the same and it equals 3 bits. The components of the RGB image from the digital camera go to a binary code converter into a RNS code, at the output of which we obtain the residues of bi obtained from the division of the pixel values of the selected modulo system p_i.

The resulting arrays of residues are fed to the encoder, which carries out the further processing (compression) of the image presented by the residues using arithmetic coding (Fig. 1).

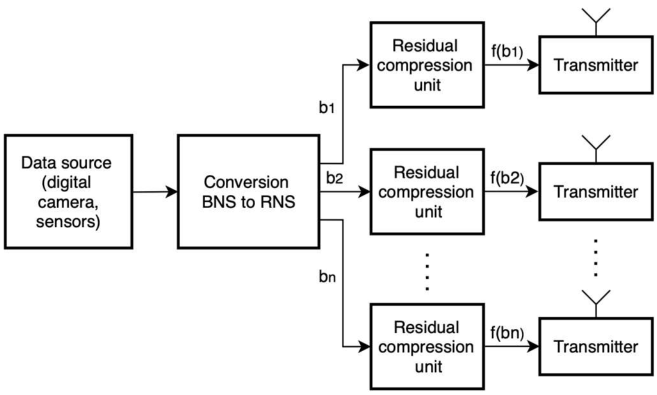


Figure 1. Scheme of image transformation and processing.

B. IMPLEMENTING THE TRANSFORMATION OF BINARY CODE INTO RNS CODE

Since the proposed method of image compression requires the conversion of data into a system of residual classes, an important task is to select a conversion way that would provide the low hardware complexity and high speed. The authors explored two procedures of implementing the conversion to FPGA, a first, using the Verilog language (Procedure 1), and a second, employing a combination based on the properties of moduli 2^n (Procedure 2).

The transformation of binary code into the RNS code in Verilog is presented according to a Procedure 1 [30], [31]:

modulo convert_8 (input [7: 0] A, output [2: 0] b1, b2, b3);

parameters $p1 = 5, p2 = 7, p3 = 8$ # (those parameters are justified in Subsection A);

assign b1 = A% p1;

assign b2 = A% p2;

assign b3 = A% p3.

Results of simulating the block conversion confirmed its correct operation, where A is the byte of the input data given in the decimal number system; b_1, b_2, b_3 – residues from the division of input data into moduli p_1, p_2, p_3 correspondingly.

For example, if $A = 128$ (a third byte on the left in Fig. 2) then $b_1 = 128 (\mathrm{mod}\ 5) = 3$; $b_2 = 128 (\mathrm{mod}\ 7) = 2$; $b_3 = 128 (\mathrm{mod}\ 8) = 0$.

The use of a 2^n type modulo simplifies the converter construction, since the remainder of this modulo is n junior bits of the binary number, which is reflected by the absence of additional blocks in the formation of residues modulo p3, the output b3 (Fig. 2). The advantages of using modules of this type are confirmed by the results of the work [29].

The block conversion of the binary code into the RNS code is described in the Verilog language and implemented on the FPGA of Altera Family Cyclone IV GX Device EP4CGX15BF14C6 (Procedure 2) (see Figure 3) with using the Altera platform for programming and developing under FPGA Altera Quartus II.

Hardware costs for converting the 8-bit binary code are 124 total logic elements; the conversion time is 2 ns.

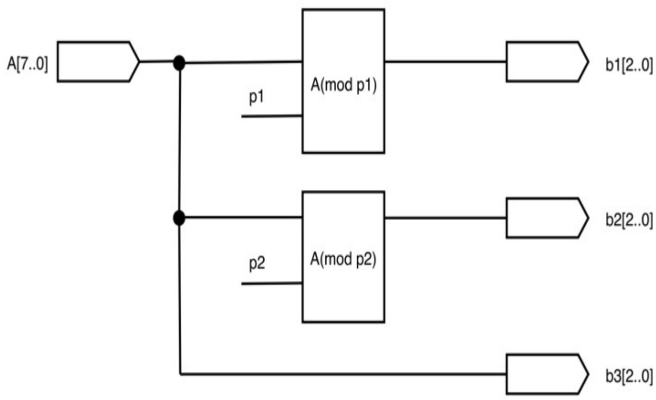


Figure 2. Functional diagram of conversion block.

Possible sets of moduli for various bits of input data are given in Table 1.

The experimental studies of hardware complexity for block conversion (see Fig. 1) are carried out on various modulo sets (see Table 1) and the results are illustrated in Fig. 3.

To select the residual compression algorithm, the frequency distribution of the received residues during the conversion of files into RNS was estimated.

Table 1. Modulo sets for converting data into RNS

Input data range	Modulo set	Modulo value	Working range
8	1	p1=3, p2=7, p3=13	Pn=273
	2	p1=5, p2=7, p3=8	Pn=280
16	1	p1=29, p2=43, p3=47	Pn=58609
	2	p1=3, p2=7, p3=11, p4=13, p5=23	Pn=69069
24	1	p1=229, p2=239, p3=307	Pn=16802417
	2	p1=7, p2=23, p3=29, p4=59, p5=61	Pn=16803731
32	1	p1=241, p2=257, p3=263, p4=269	Pn=4381856939
	2	p1=19, p2=29, p3=37, p4=43, p5=59, p6=83	Pn=4292910977

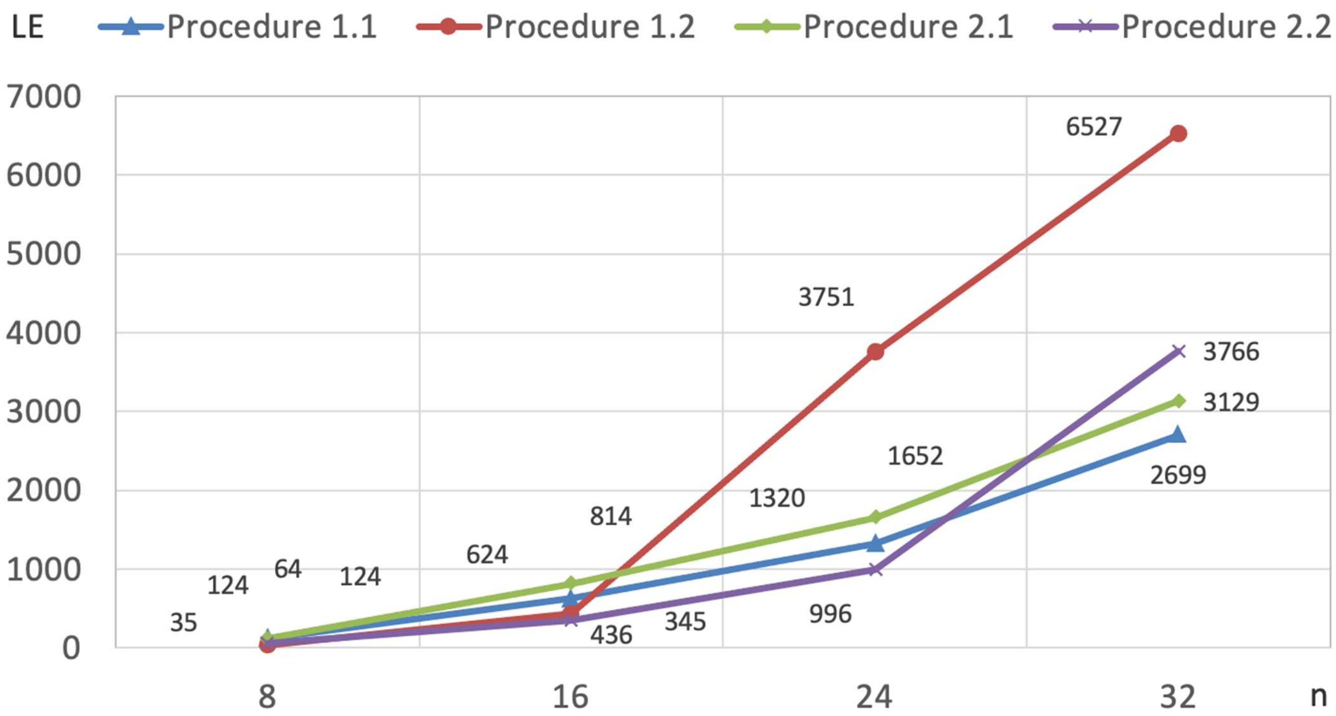


Figure 3. Hardware complexity of the converter from binary code into RNS when the modulo set 1 is using (see Table 1): as it was explained above, Procedure 1 is the operation to obtain the Verilog remainder (%); Procedure 2 is based on the properties of moduli 2^n.

The code of the program for calculating the frequency of residues in the file according to module 5 looks like this:

```
#Mod 5
freqreading = open('barbara_01.rrns', 'rb') # barbara_01.rrns – the name of the residuals file
p5 = freqreading.read()
freqreading.close()
from collections import defaultdict
chars = defaultdict(int)
f=list(p5)
For char in p5:
    chars[char] += 1
print("mod_0=",chars[0])
print("mod_1=",chars[1])
print("mod_2=",chars[2])
print("mod_3=",chars[3])
print("mod_4=",chars[4])
```

Table 2 shows an example of the residual frequency distribution when an image is transferred into the RNS

Residue value	Frequency of residues occurrence in different moduli		
	modulo 5	modulo 7	modulo 8
0	713439	531379	424943
1	640063	478764	404058
2	640048	481931	368485
3	649366	485959	424085
4	646360	457596	405393
5		432862	387914
6		420785	423650
7			450748

After image transfer of the RNS we got 8 received remainders from 0 to 7 of 3 bits with different frequencies of residues occurrence. The different frequency of residues occurrence proves the possibility of using data compression statistical methods, in particular arithmetic coding.

C. ARITHMETIC CODING

Authors selected arithmetic coding as the method of residue compression. That provides the optimum compression degree in terms of the entropy assessment due to Shannon coding. Each symbol requires about H bits, where H is the information entropy of the source. Unlike the Huffman algorithm, the arithmetic coding demonstrates a high efficiency for uneven intervals of probability distribution for coded symbols. However, in the case of coded symbols with the uniform distribution, the efficiency of the arithmetic coding is closer to Huffman prefix codes [5].

The compressed sequences are fed from encoder output to transmitters (see Fig. 1) and further they are transmitted through parallel channels [25], [33].

The decoder (removal recovery unit) restores residues using the arithmetic decoding algorithm (Fig. 4).

Then the restored residues are converted from the RNS into a binary code according to formulas [29], [34]:

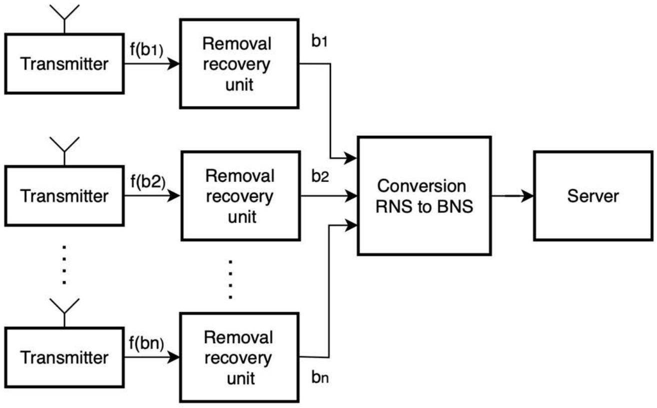


Figure 4. Scheme of image reception and recovery.

$$A = (\sum_{i=1}^{n} b_i \cdot B_i) mod\ P, \quad (3)$$

where $n = 3$ – the number of moduli;

$B_i = \frac{P}{p_i} \cdot \delta_i \equiv 1(\mathrm{mod}\ p_i)$ – orthogonal bases,

where δ_i – the weight of the orthogonal element, $0 < \delta_i < p_i$;

$P = \prod_{i=1}^{3} p_i$, p_i – are mutually prime numbers.

For moduli 5, 7, 8, the orthogonal bases are $B_1 = 56, B_2 = 120, B_3 = 105$ correspondingly.

Using formula (3), the block conversion can be implemented on FPGA.

Simulation results of block conversion are shown in Fig. 6, and the code at the A0 output is given in the decimal notation. For example, if $b_1 = 3, b_2 = 4, b_3 = 4$ $b1 = 3, b2 = 4, b3 = 4$, and $B_1 = 56, B_2 = 120, B_3 = 105$ then we get by formula (1) in Verilog: $A_0 = (3 \cdot 56 + 4 \cdot 120 + 4 \cdot 105)\%280 = 228$.

Since the RNS reverse transformation restores the input data without loss, we may conclude that the proposed method provides a reduction in image redundancy without loss of quality (Fig. 5).

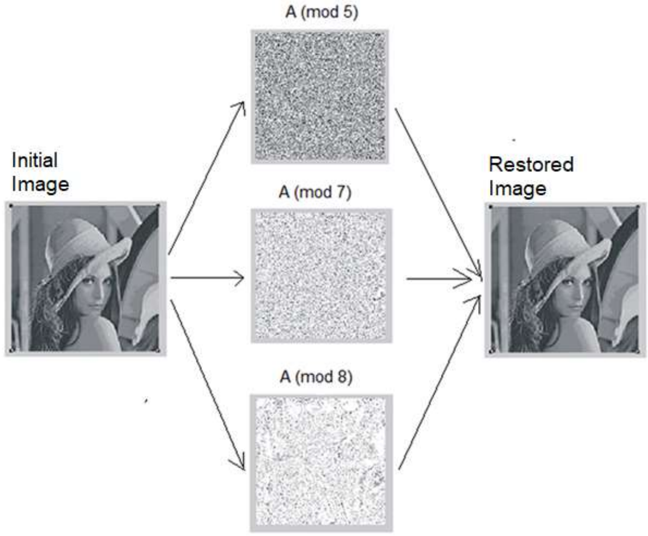


Figure 5. Example of image transformation

D. CASE STUDY

To compare the efficiency of compression algorithms, the compression ratio is used [12]:

$$k = \frac{V_0}{V_c}, \quad (4)$$

where V_0 is the input data amount, V_c is the data amount after compression.

To run the experiment, authors proposed the algorithm consisting of the following steps.

1. Data input file is read and converted to byte type (Python language).
2. Residues are read and calculated for moduli 5, 7, 8 per each data byte.
3. A point 2 above is repeated unless all data will read.
4. Resulting residues of moduli are recorded in separate files.

5. Files (step 4) are compressed using arithmetic coding.
6. Calculating the compression ratio with residues.
7. Transfer of compressed files with residues by separate routers
8. Unzipping the obtained files.
9. Conversion of residues into the position number system.
10. Data converting to a given type.
11. Calculating the compression ratio.

Then an entropy is calculated by the formula:

$$H(X) = -\sum_{i=1}^{n} P(x_i) \cdot log_2 P(x_i), \quad (5)$$

where $P(x_i)$ is the probability of residuals according to the corresponding module.

The experimental research of the compression ratio for image parts presented by the residues carried out on test images and other types of files (Table 3).

The conducted experimental research confirmed advantages of the proposed method for image transformation in RNS with a subsequent compression of residues by the arithmetic coding. The results showed that the proposed method provides a compression ratio from 10% to 14% depending on the image format (see Table 3).

The outcomes of an experimental study for the compression speed of the initial image file and the residual files obtained as a result of division modulo p1=5, p2=7, p3=8 (see a step 5 of the proposed algorithm above) are given in Table 4. A parameter 'Increase speed' is obtained as a result of dividing the value of the parameter 'Compression time of input file' by the value of the parameter 'Compression time of residuals file'.

Table 3. Results of data compression in RNS system.

File name and type	File size, bytes	Files size with residues by moduli, bytes	Compressed files size with residues, bytes			Entropy $H(X)$	Compression ratio, %
		p1, p2, p3	p1	p2	p3		
piza.jpeg	3289276	1233478	954155	1152704	1232481	2.71	9.8
lena.tiff	786576	294966	228295	276024	294927	2.71	9.7
lena.bmp	263229	98711	76399	92372	98709	2.71	9.7
barbara.tiff	239	90	67	79	86	2.58	13.8
barbara.bmp	263229	98711	76399	92356	98683	2.71	9.7
rgb.giff	14436	5414	4182	5058	5337	2.69	10.2
cmy.gif	20246	7592	5873	7097	7537	2.70	10.0
fruit.jpg	63780	23918	18487	22362	23914	2.71	9.7

Table 4. Compression time of input file and residuals files

File name and type	File size, bytes	Compression time of input file, s	Compression time of residuals file, s	Increase speed, times
piza.jpeg	3289276	47.302	15.7673458	2,99
lena.tiff	786576	11.3115006	3,77050019	3,00
lena.bmp	263229	3.82792251	2.27529238	1,68
barbara.bmp	263229	3.78541296	2.2142107	1,71
fruit.jpg	63780	0.9172	0.5365	1,71
cmy.gif	20246	0.3173	0.1766	1,79
rgb.giff	14436	0.20903	0.1263	1,65
barbara.tiff	239	0.0078	0.00586	1,33

As it can be seen from Table 4, we got an increase in the speed of image compression of about 1.9 times by processing the small data (the bit capacity of residues is 3 bits) as well as providing the parallel operation of the compression residue units by RNS selected moduli. Therefore, the coder can operate in 2.5 times faster than other existing algorithms.

VI. CONCLUSIONS

Authors developed a method, which decreases significantly the file compression time in the two-stage frame: (i) the RNS transformation is running to divide the data and obtain residues, and (ii) the resulting residues are compressed in parallel using arithmetic coding. Therefore, the coder can operate in 2.5 times faster than other existing algorithms

The application of arithmetic coding for compression of residuals ensures a lossless compression ratio from 10% to 14% depending on the image format. Besides, the proposed method provides an additional (approximately 13%) data compression in comparison with existing archivers.

In a case of employing the proposed method in IoT systems built using the WSN technology enabling decreasing the bandwidth requirements of communication channels by transmitting residues on different routes. Due to this fact, the data are protected from attacks in the middle (Man in the Middle).

The proposed approach can be extended to secure (encrypt) images during transmission over open communication channels.

In future, authors are going to investigate the compression ratio of the data presented in the RNS when using modules with a bit size of 5-8 bits. Moreover important is the study of hardware complexity in the implementation of the proposed algorithms for data compression in IoT systems.

VII. DISCUSSION

The paper represents a significant extension of Reference [10] with further details regarding the schemes of image conversion and processing, the results of modelling the operation of the image conversion block in RNS, and the scheme of image reception and recovery. Moreover, the research methodology was developed and the results of the data compression study in RNS are presented (see Table 3). The studies of the compression time for the input file and residual files are given in Table 4.

In principle, it is possible to extend the system of RNS modules up to four instead of three (see as Fig. 5), for example, it enables to restore the image in case of loss or distortion for one of the parts (residues in one of the modules). However, that requires additional costs and didn't consider in this paper.

The proposed approach can be extended to secure (encrypt) images during transmission over open communication channels.

Acknowledgments

A second author would like to thank German Academic Exchange Service (DAAD) for their support of a research project on this topic at HTW Berlin during the term October 7-December 7, 2019.

References

[1] M. El-Hajj, A. Fadlallah, M. Chamoun, A. Serhrouchni, "A survey of internet of things (IoT) authentication schemes," *Sensors*, vol. 19, issue 5, pp. 1141, 2019. https://doi.org/10.3390/s19051141.

[2] C. Puliafito, E. Mingozzi, F. Longo, A. Puliafito, O. Rana, "Fog computing for the internet of things: A survey," *ACM Transactions on Internet Technology (TOIT)*, vol. 19, issue 2, pp. 1-41, 2019, https://doi.org/10.1145/3301443.

[3] H. Fesenko, V. Kharchenko, A. Sachenko, R. Hiromoto and V. Kochan, "An Internet of Drone-based multi-version post-severe accident monitoring system: Structures and reliability," *In book Dependable IoT for Human and Industry – Modeling, Architecting, Implementation*, Editors: Vyacheslav Kharchenko, Ah Lian Kor and Anrzej Rucinski, River Publishers, 2018, pp. 197-218. https://doi.org/10.1201/9781003337843-12.

[4] E. Satir, H. Isik, "A Huffman compression based text steganography method," *Multimed Tools Appl*, vol. 70, pp. 2085–2110, 2014, https://doi.org/10.1007/s11042-012-1223-9.

[5] O. Ghorbel, I. Jabri, W. Ayedi, M. Abid, "Experimental study of compressed images transmission through WSN," *Proceeding of the IEEE International Conference ICM'2011*, Hammamet, 2011, pp. 1-6, https://doi.org/10.1109/ICM.2011.6177378

[6] M. Hosseini, D. Pratas, A. J. Pinho, "A survey on data compression methods for biological sequences," *Information,* 2016, vol. 7, 56, https://doi.org/10.3390/info7040056.

[7] Uthayakumar J., Elhoseny M., Shankar K. Highly Reliable and Low-Complexity Image Compression Scheme Using Neighborhood Correlation Sequence Algorithm in WSN. *IEEE Transactions on Reliability*, 2020, vol. 69, issue 4, pp. 1398-1423, https://doi.org/10.1109/TR.2020.2972567.

[8] F. Bouakkaz, W. Ali, & M. Derdour, "Forest fire detection using wireless multimedia sensor networks and image compression," *Instrum. Mes. Métrologie*, vol. 20, pp. 57-63, 2021, https://doi.org/10.18280/i2m.200108.

[9] S. Jun, K. Przystupa, M. Beshley, et al., "A cost-efficient software based router and traffic generator for simulation and testing of IP network," *Electronics*, vol. 9, issue 1, pp. 40, 2020, https://doi.org/10.3390/electronics9010040

[10] V. Yatskiv, A. Sachenko, N. Yatskiv, P. Bykovyy, A. Segin, "Compression and transfer of images in wireless sensor networks using the transformation of residue number system," *Proceedings of the 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2019)*, Metz, France, 18-21 September 2019, vol. 2, pp. 1111-1114. https://doi.org/10.1109/IDAACS.2019.8924372.

[11] C. M. Sadler, M. Martonosi, "Data compression algorithms for energy-constrained devices in delay tolerant networks," *Proceedings of the 4th ACM International Conference on Embedded Networked Sensor Systems*, Boulder, Colorado, USA, 2006, pp. 265-278, https://doi.org/10.1145/1182807.1182834

[12] J. Uthayakumar, T. Vengattaraman, P. Dhavachelvan, "A new lossless neighborhood indexing sequence (NIS) algorithm for data compression in wireless sensor networks," *Ad Hoc Networks*, vol. 83, pp. 149-157, 2019. https://doi.org/10.1016/j.adhoc.2018.09.009.

[13] D. Birvinskas, V. Jusas, I. Martisius, I. Damasevicius, "Fast DCT algorithms for EEG data compression in embedded systems," *Computer Science and Information Systems*, vol. 12, issue 1, pp. 49–62, 2015, https://doi.org/10.2298/CSIS140101083B.

[14] D. Kandris, C. Nakas, D. Vomvas, G. Koulouras, "Applications of wireless sensor networks: an up-to-date survey," *Applied System Innovation*, vol. 3, issue 1, 14, 2020. https://doi.org/10.3390/asi3010014

[15] M. L. Kaddachi, A. Soudani, I. Nouira, V. Lecuire, K. Torki, "Efficient hardware solution for low power and adaptive image-compression in WSN," *Proceedings of the 2010 17th IEEE International Conference on Electronics, Circuits and Systems*, Athens, 2010, pp. 583-586, https://doi.org/10.1109/ICECS.2010.5724579.

[16] A. Abdul-Barik, M. Ibrahim, S. Akobre, "Improved Lempel-Ziv-Welch's error detection and correction scheme using redundant residue number system," *Circulation in Computer Science*, vol. 2, issue 6, pp. 25-30, 2017, https://doi.org/10.22632/ccs-2017-252-33

[17] J. Liu, F. Chen, D. Wang, "Data compression based on stacked RBM-AE model for wireless sensor networks," *Sensors*, vol. 18, issue 12, 4273, 2018. https://doi.org/10.3390/s18124273.

[18] A. Alhassan, I. Saeed, P. A. Agbedemnab, "The Huffman's method of secured data encoding and error correction using residue number system (RNS)," *Communication on Applied Electronics (CAE) Journal*, *Foundation of Computer Science (FCS)*, New York, USA, vol. 2, no. 9, pp. 14-18, 2015, https://doi.org/10.5120/cae2015651844.

[19] G. Kumar, R. Kumar, "Analysis of arithmetic and Huffman compression techniques by using DWT-DCT," *International Journal of Image, Graphics and Signal Processing (IJIGSP)*, vol. 13, no. 4, pp. 63-70, 2021, https://doi.org/10.5815/ijigsp.2021.04.05.

[20] B. A. Lungisani, et al., "Image compression techniques in wireless sensor networks: a survey and comparison," *IEEE Access*, vol. 10, pp. 82511-82530, 2022, https://doi.org/10.1109/ACCESS.2022.3195891.

[21] S. Li, J. G. Kim, D. H. Han, K. S. Lee, "A survey of energy-efficient communication protocols with QoS guarantees in wireless multimedia sensor networks," *Sensors*, vol. 19, 199, 2019. https://doi.org/10.3390/s19010199

[22] S. A. Dehkordi, K. Farajzadeh, J. Rezazadeh, R. Farahbakhsh, K. Sandrasegaran, and M. A. Dehkordi, "A survey on data aggregation techniques in IoT sensor networks," *Wireless Networks*, vol. 26, issue 2, pp. 1243-1263, 2020, https://doi.org/10.1007/s11276-019-02142-z.

[23] E. D. Pitchika, S. Bharadwaj, "Fast base extension using single redundant modulus in a residue number system," *Proceedings of the 2019 IEEE International Conference on Power Electronics, Control and Automation (ICPECA)*, New Delhi, India, 2019, pp. 1-5, https://doi.org/10.1109/ICPECA47973.2019.8975450.

[24] H. Z. Eldin, M. A. Elhosseini, H. A. Ali, "Image compression algorithms in wireless multimedia sensor networks: A survey," *Ain Shams Engineering Journal*, vol. 6, issue 2, pp. 481-490, 2015. https://doi.org/10.1016/j.asej.2014.11.001.

[25] M. Zeidan, & B. Kamiloğlu, "Three-dimensional imaging technique to compare digital impression CEREC Omnicam intraoral camera (CAD) and tri-dimensional cone-beam computed tomography, to measure maxillary casts: Unilateral and bilateral cleft lip and palate up to 6 months of age, applied in nanotechnology," *Applied Nanoscience*, vol. 13, pp. 1753–1759, 2021.

[26] P. V. Ananda Mohan, "Reverse converters for the moduli set {2n+1-3, 2n-1, 2n-1-1}," *IETE Journal of Education*, vol. 57, pp. 31-43, 2016, https://doi.org/10.1080/09747338.2015.1115378.

[27] T. Sheltami, M. Musaddiq, E. Shakshuki, "Data compression techniques in wireless sensor networks," *Future Generation Computer Systems*, vol. 64, pp. 151-162, 2016, https://doi.org/10.1016/j.future.2016.01.015

[28] A. Drozd, J. Drozd, S. Antoshchuk, V. Antonyuk, K. Zashcholkin, M. Drozd, O. Titomir, "Green experiments with FPGA," *In book: Green IT Engineering: Components, Networks and Systems Implementation*. V. Kharchenko, Y. Kondratenko, J. Kacprzyk (Eds.), vol. 105, Berlin, Heidelberg: Springer International Publishing, 2017, pp. 219–239. https://doi.org/10.1007/978-3-319-55595-9_11.

[29] V. Yatskiv and T. Tsavolyk, "Improvement of data transmission reliability in wireless sensor networks on the basis of residue number system correcting codes using the special module system," *Proceedings of the 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*, Kyiv, Ukraine, 2017, pp. 890-893, doi: 10.1109/UKRCON.2017.8100376.

[30] Z. Hu, V. Yatskiv, A. Sachenko, "Increasing the data transmission robustness in WSN using the modified error correction codes on residue number system," *Elektronika ir Elektrotechnika*, vol. 21, issue 1, pp. 76-81, 2015. https://doi.org/10.5755/j01.eee.21.1.6657.

[31] M. Nasri, A. Helali, H. Sghaier, H. Maaref, "Adaptive image transfer for wireless sensor networks (WSNs)," *Proceedings of the 5th International Conference on Design & Technology of Integrated Systems in Nanoscale Era*, Hammamet, pp. 1-7, 2010. https://doi.org/10.1109/DTIS.2010.5487597.

[32] H. Kasban, S. Nassar, & M. A. M. El-Bendary, "Medical images transmission over wireless multimedia sensor networks with high data rate," *Analog Integr Circ Sig Process*, vol. 108, pp. 125–140, 2021, https://doi.org/10.1007/s10470-021-01854-7.

[33] B. Al Hayani, H. Ilhan, "Image transmission over decode and forward based cooperative wireless multimedia sensor networks for rayleigh fading channels in Medical Internet of Things (MIoT) for remote health-care and health communication monitoring," *Journal of Medical Imaging and Health Informatics*, vol. 10, no. 1, pp. 160-168(9), 2020, https://doi.org/10.1166/jmihi.2020.2691.

[34] S. Suseela, S. Jeevanantham, A. Kavitha, M. Saravanan, A. E. Narayanan, "Challenging issues for handling multimedia data in wireless multimedia sensor networks," *Journal of Computational and Theoretical*

Nanoscience, vol. 16, no. 4, pp. 1215-1220(6), 2019, https://doi.org/10.1166/jctn.2019.8022.

[35] Y. A. Ur Rehman, M. Tariq, "Chapter 3 – Visual information processing and transmission in wireless multimedia sensor networks: a deep learning based practical approach," Editor(s): S. Shukla, A. K. Singh, G. Srivastava, F. Xhafa, *In Intelligent Data Centric Systems, Internet of Multimedia Things (IoMT)*, Academic Press, 2022, pp. 47-66, https://doi.org/10.1016/B978-0-32-385845-8.00008-3.

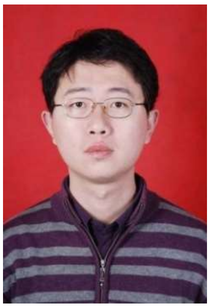
[36] A. Mateen, M. Sehar, K. Abbas and M. A. Akbar, "Comparative analysis of wireless sensor networks with wireless multimedia sensor networks," *Proceedings of the 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, Chennai, India, 2017, pp. 80-83, https://doi.org/10.1109/ICPCSI.2017.8391847.

***ANATOLY SACHENKO**, a Professor, Research Institute for Intelligent Computer Systems, Director, West Ukrainian National University, Ternopil, Ukraine.*

Research interests: Distributed Sensor Networks, Artificial Intelligence Application, Image Processing, Wireless Sensor Networks, Project Management etc.

***VASYL YATSKIV**, Eng. Automation of technological processes. Ivano-Frankivsk Technical Univercity of Oil and Gas, Ukraine (1996). Ph.D. of Computers, system and network. Lviv Polytechnic Univercity, Ukraine (2001). Professor, head of Department of Cybersecurity at Faculty of Computer Information Technologies of West Ukrainian National University. Research Interests are Wireless Sensor Networks, Residue Number System, Correcting Codes.*

***PROF. DR. JÜRGEN SIECK** received his degree in mathematics in 1981 and his PhD in Computer Science in 1989 from the Humboldt University Berlin, Germany. Now he is the senior researcher at the research group "Informations- und Kommunikation-sanwendungen" (INKA) and professor for computer sciences with a specialisation on algorithms, multi-media and mobile application for the degree programme Applied Computer Science at the University of Applied Sciences HTW Berlin. Previously, he was visiting lecturer at Monash University Melbourne (Australia), Yuanpei University Hsinchu (Taiwan) and at the University of Cape Town (South Africa) and was visiting researcher at the International Computer Science Institute in Berkley (USA), Polytechnic of Namibia and at Old Dominion University Norfolk in Virginia (USA). In February 2013 he was awarded an honorary doctorate from Odessa National Polytechnic University (Ukraine). Since 2013, he is Principal investigator of the cluster of excellence "Bild Wissen Gestaltung" at the Humboldt-University Berlin. Since 2015, he is also a professor of computer science at Namibia University of Science and Technology.*



Su Jun received the B.S. degree from National Technical University of Ukraine, Ukraine, in 2002, and the M.S. degree from National Technical University of Ukraine, Ukraine, in 2004, all in Computer System and Network. He defended his Ph.D. degree at Ternopil National Economic University (now West Ukrainian national University), Ukraine. Now he works as Associate professor At School of Computer Science, Hubei University of Technology, Wuhan, China. His main Areas of Research Interests Wireless Sensor Networks, Network Management, Information Security and Software Engineering.