https://doi.org/10.30857/2786-5371.2023.3.2

УДК 621.382

MAZIN M. Yu., ONYKIIENKO Yu. O.
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine

WAVELET TRANSFORM APPLICATION FOR IMAGE PROCESSING IN MICROCONTROLLER BASED INTERNET OF THINGS SYSTEMS

Purpose. *Investigating the efficiency of Haar Wavelet Transform as an additional transformation to JPEG image compression in the context of limited resources on microcontrollers.*

Methodology*. The aims are exploring the effectiveness of using Haar wavelet transform as an additional transformation to JPEG for image compression, considering the limited resources of microcontrollers. The research focuses on reducing of the image transmission time and energy consumption by utilizing the combined approach. The methodology involves experimental evaluation of image quality processed using Haar wavelet transform and compressing performing JPEG transformation on the microcontroller module ESP32-CAM, which captures images with an embedded camera. The evaluation includes comparison of images compressed with JPEG alone and with the combination of the wavelet transform and JPEG, calculating the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) coefficients.*

Findings. *A software module for implementing Haar wavelet transform on the ESP32-CAM microcontroller has been developed. The efficiency of using Haar wavelet transform as an additional method to JPEG for image compression and transmission is evaluated, considering the limited resources of the microcontroller. The size of compressed images and their quality was compared, assessing the size reduction of compressed images and the quality using PSNR and SSIM coefficients. The results showed that the additional use of Haar wavelet transform reduced the image size from 25% to times without significant loss in image quality, although the image processing time increased to 1 second.*

Originality. *A combined approach for image compression using multiple video processing methods, taking advantage of the capabilities of 32-bit microcontrollers with external memory, was proposed.*

Practical value. *The proposed image processing method can reduce the transmission time and energy consumption during transformations and image transmission, thereby extending the battery life of the sensor node. Moreover, this compression approach can maintain the quality of the original image with minimal data loss by reducing the amount of visual information lost during compression.*

Keywords: *image compression; wavelet transforms; Haar wavelet; PSNR; SSIM.*

Introduction. Modern sensor networks consist of nodes equipped with sensors, a computing unit, wireless communication modules with a low data transfer rate, and a battery. In order to provide economical monitoring of the environment, the nodes must be inexpensive and energy efficient. For this reason, the node computing unit is usually an inexpensive microcontroller with limited processing power and RAM [1]. A sensor node can be equipped with a small camera to track an object or monitor the environment, thus forming a camera sensor network [2, 3]. However, systems designed for image processing require more resources than a typical low-cost sensor node. Even if the image can be stored on the node itself using cheap, fast and large flash memory [4], transferring the image over the network may require quite high bandwidth and a lot of energy. Therefore, image processing or image data compression must be performed.

With modern data and signal processing methods that require very little RAM, low-cost sensor networks can be created by connecting small cameras and external flash memory to low-cost sensor nodes. One of the widely used techniques for image preprocessing, to view or compress an image, is the discrete wavelet transform or wavelet transform.

Until now, image processing technologies required a memory volume that significantly exceeds the resources of conventional microcontrollers. Therefore, in the sensor platforms of image processing, varieties of wave transformation in the form of fractional wave filters were used, which require very little memory in the processing process. For example, in work [5] a variant of the filter is proposed that requires less than 1.5 KB of RAM to convert an image of 256 × 256 pixels with 8-bit color depth using only 16-bit integer arithmetic. Thus, the fractional wavelet filter works well within the limitations of typical low-cost sensor nodes [6]. The disadvantage of transformations based on fractional filters is a sufficiently long processing time.

Currently, 32-bit microcontrollers, thanks to high performance and additional external memory, allow significantly reduce calculation time and increase the efficiency of wave transformations. In addition, there are opportunities to perform more complex, combined image processing.

Objectives. The purpose of the work is to investigate the effectiveness of using the Haar wavelet transform for image processing and then saving them in JPG format for further transmission within the network. To achieve the goal, it is necessary to capture and process the image using the ESP32-CAM microcontroller module, evaluate the effect of the number of iterations of the wavelet transformation and the quality factor of the JPG transformation on the size and quality of the obtained image. It is also necessary to estimate the processing time of the images.

Results of previous studies. Today, the main directions for improving image processing in sensor networks are to ensure a reduction in the time of transmission of the images themselves and a reduction in energy consumption during transformations and transmission of images. The image transfer time directly depends on the image size. In work [7], in order to reduce the energy consumption of the node due to the limited capacity of energy storage devices, an adaptive Haar wavelet transformation was used. An approach to grayscale compression is proposed, which ensures the quality of the original image with the least possible data loss. In this way, the battery life of the sensor is extended by reducing the amount of visual information that is lost during compression. The paper also compares the image size before and after compression. For networks with a large number of nodes located close together, redundant information in the form of images is a problem because it increases energy consumption. Therefore, the use of wavelet transformation along with the use of data prioritization together with information about the state of the channel between sensor nodes and a cluster node is proposed in [8]. This approach in the form of shared image transmission increases the service life of batteries in the network. Each sensor node receives images, which were decomposed into four sub-images using a 1-level wavelet transform and were transmitted through different channels. The cluster node manages the transmission process.

The hardware resources of the sensor nodes are no less important than the image quality, which affects the cost of the equipment. Considerable attention is paid to reducing the amount of memory used in the wavelet transform process. In [9], a variant of the architecture for calculating the wavelet transform of an image using a fractional wavelet filter (FrWF) is proposed, which provides a memory amount reduction during image processing. The disadvantage of the approach is that the original image and transformed coefficients are stored on an external memory card. Another option for calculating the wavelet transform using a fractional wavelet filter is the so-called lifting-based implementation of the transform [10], which requires fewer calculations than FrWF. The evaluation results show that for high-resolution images (2048×2048), the complexity of the proposed implementation is about 40% less than that of FrWF without any additional memory requirements.

Features of the Haar wavelet transform. The Haar transform is one of the simplest and most basic wavelet transform methods. For example, a one-dimensional discrete signal $f(f_1,f_2,..,f_n)$ is decomposed by the Haar transform into two components of the same size. The first component is called the average value or approximation, and the second - the difference or detail. The formula for calculating the value of the average sub-signal (sub-signal), $a_1 = (a_1, a_2, ..., a_{N/2})$, at the first level for a signal of length N has the following form:

$$a_n = \frac{f_{2n-1} + f_{2n}}{\sqrt{2}}, \qquad (1)$$

where n = 1, 2, 3, …, N/2, and detailing sub-signal, $d_1 = (d_1, d_2, ..., d_{N/2})$, at the same level is represented as

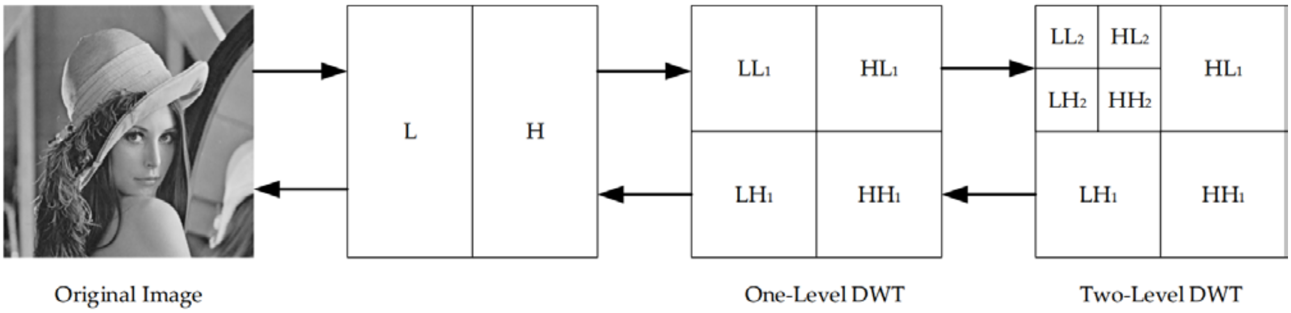
$$d_n = \frac{f_{2n-1} - f_{2n}}{\sqrt{2}}, \qquad (2)$$

These values form two new signals $a\{a_n\}$ $n \in Z$ and $d\{d_n\}$ n $\in Z$, one of which is a rough version of the initial signal (each pair of f elements corresponds to their arithmetic mean), and the other contains the information necessary to restore the initial signal. Really,

$$\begin{matrix} f_{2n-1} = a_n + d_n \\ f_{2n} = a_n - d_n \end{matrix} \qquad (3)$$

A similar operation can be applied to signal a and thereby two signals will be obtained, one of them is a rough version of a and the other contains the detailed information needed to reconstruct signal a.

Discrete two-dimensional wavelet transform is performed by sequentially applying one-dimensional wavelet transforms to the rows and columns of the image matrix. First, a one-dimensional wavelet transform is applied to each row, and the resulting results are written back into place. The wavelet transform is then applied to the columns. This process divides the image into four parts: a low-frequency component (LL), a vertical-high-frequency component (LH), a horizontal-high-frequency component (HL), and a high-frequency component (HH), as shown in Fig. 1.



Source: [11].

Fig. 1. **One-level and two-level Haar transform wavelet algorithm**

Applying an N-fold 2D wavelet transform means repeating this process N times. Each subsequent wavelet transformation is applied to the lower quarter of the matrix (quadrant LL). The result is a hierarchical image structure where each level represents a modified version of the previous level. This makes it possible to analyze the image at different scales and highlight its details depending on the requirements of the task.

The inverse wavelet transform is performed recursively, starting from the lowest level and reconstructing each level based on its respective constituents. This process allows you to reconstruct an image from different levels of decomposition and use it for further analysis or processing. To restore quadrant LL1, presented in fig. 1, quadrants LL2, LH2, HL2, HH2, etc. are used. Similarly, N-fold inverse wavelet transformation is performed. It should be noted that the specified transformation is hierarchical, i.e. if, when applying the inverse wavelet transformation, not all levels are calculated, but a smaller number of them, then a reduced copy of the image is formed in the LL quadrant, as shown in Fig. 1. In particular, if no inverse wavelet transform is used at all, then the youngest quadrant is also a reduced copy of the image. Thanks to this property, the inverse wavelet transform allows you to cut out fragments of images at different scales. It should be noted that the available scales are determined by the number of levels of the wavelet transformation and are not arbitrary, but differ by a factor of two. Therefore, the application of two-dimensional wavelet transformation allows obtaining a multi-scale representation of the image, where each level has its own information about the details of the image at the corresponding scale. It is convenient for various image processing tasks such as compression, filtering, edge detection and others.

PSNR (Peak Signal-to-Noise Ratio) is an indicator of the quality of signal reproduction, which is measured in decibels (dB). PSNR compares the maximum possible power of the signal with the power of the noise that distorts it. The higher the PSNR, the better the signal quality. *PSNR* is often used to evaluate the quality of compressed images or videos. *PSNR* can be calculated using the following formula:

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX^2}{MSE}\right), \tag{4}$$

where *MAX* is the maximum possible value of the signal and *MSE* is the root mean square error between the original and reproduced signals. For images consisting of 8-bit pixels, *MAX* = 255. *MSE* can be calculated using the following formula:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \ , \tag{5}$$

where *I*(*i*,*j*) is the pixel value of the original signal at position (*i*,*j*) and *K*(*i*,*j*) is the pixel value of the reproduced signal at the same position. *m* and *n* are the vertical and horizontal dimensions of the signal, respectively.

SSIM (Structural Similarity Index Measure) - is a method for evaluating the perceived quality of digital images and videos, and for measuring the similarity between two images. SSIM is based on the assumption that human visual perception is adapted to detect structural information in a scene, and takes into account important perceptual phenomena such as luminance and contrast masking. Mathematically, *SSIM* can be calculated using the following formula:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{6}$$

where *x* and *y* are the two images to be compared, μ_x and μ_y are the averages of pixels *x* and *y*, σ_x^2 and σ_y^2 are the variances of pixels *x* and *y*, σ_{xy} is the covariance of pixels *x* and *y*, and c_1 and c_2 are constants, which prevent division by zero. Typically, $c_1 = (k_1L)^2$ and $c_2 = (k_2L)^2$, where *L* is the dynamic range of pixels (255 for 8-bit images) and k_1 and k_2 are constants (0.01 and 0.03). *SSIM* can be computed locally on small portions of images to produce an *SSIM* map that shows regions of similarity and difference between two images. The mean value of the *SSIM* map is used as an overall measure of similarity between two images.

Average pixel values for each image:

$$\mu_X = \frac{1}{16}\sum_{i=0}^{3}\sum_{j=0}^{3}x(i,j)\,; \tag{7}$$

$$\mu_y = \frac{1}{16}\sum_{i=0}^{3}\sum_{j=0}^{3}y(i,j)\,. \tag{8}$$

Pixel variances for each image:

$$\sigma_x^2 = \frac{1}{16}\sum_{i=0}^{3}\sum_{j=0}^{3}[x(i, j) - \mu_x]^2; \tag{9}$$

$$\sigma_y^2 = \frac{1}{16}\sum_{i=0}^{3}\sum_{j=0}^{3}[y(i, j) - \mu_y]^2 . \tag{10}$$

Pixel covariance between two images:

$$\sigma_{xy} = \frac{1}{16}\sum_{i=0}^{3}\sum_{j=0}^{3}[x(i, j) - \mu_x][y(i, j) - \mu_y]. \tag{11}$$

An *SSIM* value close to or equal to 1 is considered good for images, meaning they are nearly identical or without distortion.

Description of hardware and software for investigating the effectiveness of wavelet transforms in image compression. 32-bit microcontrollers have a sufficiently high performance for processing signals of various types. However, image processing involves the use of large amounts of RAM, which most general-purpose microcontrollers do not have. The problem is solved by connecting external RAM with a serial interface to the microcontroller, although this approach reduces the speed of video processing. The ESP32-CAM module [12] was used for research, which consists of two boards: a microcontroller board with a camera and memory and a board with a USB interface and power circuits. The microcontroller board contains a dual-core ESP32 microcontroller with a Tensilica Xtensa LX6 microprocessor core and a built-in Wi-Fi channel, a 2-megapixel camera and a 4 MB PSRAM chip. The presence of Wi-Fi allows you to transfer the image directly to the web page opened in the browser. The main characteristics of the microcontroller board:

- The maximum frequency of the clock generator is 240 MHz
- Program memory capacity – 4 MB
- RAM capacity (internal + external) – 4+0.5 MB
- Camera resolution – 2 megapixels

The appearance of the ESP-CAM module with a connected camera is shown in Fig. 2.

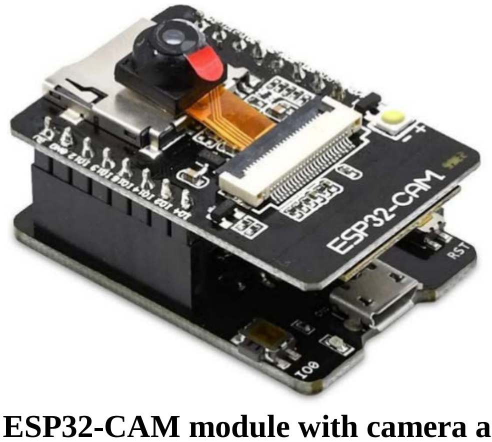


Fig. 2. **External view of ESP32-CAM module with camera and USB interface board**

The Haar wavelet transform is known for its simplicity and efficiency and is suitable for implementation on a microcontroller with limited resources. The main purpose of the experiments was to determine how much the Haar transform wavelet increases the efficiency of image compression from the camera when saved in JPG format. That is, the image was saved in JPG format after discrete wavelet transform (WDT), as shown in Fig. 3b and compared with the image saved in JPG format without additional wavelet transform (Fig. 3a). Additional compression is necessary so that the image is transferred from the microcontroller to other devices in the shortest possible time. To simplify and speed up the process of comparing results, an html page was created, which was stored in the memory of the microcontroller and to which the processed image and the size of the processed file were transferred. In addition, the number of wavelet transformation iterations and the quality level of the processed JPG file were specified through the html page. Converting the image to JPG format was performed using the functions included in the image processing library for the ESP32 microcontroller [13]. The great advantage of the library is the support of a large number of input data formats, as well as the ability to set the quality level of the image converted to JPG. In order to evaluate the quality of image compression using the JPEG + WDT method, the inverse transformation of the received data was performed on the computer with the output of the final image on the screen.

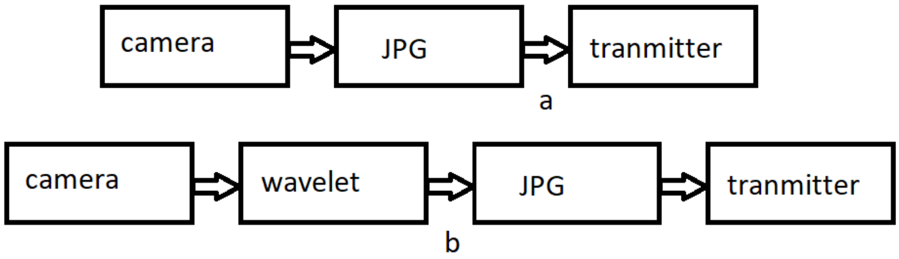


Fig. 3. **Image processing and transmission algorithm: a) conventional, b) proposed**

During the experiments, the following was studied:

1. Size ratio of black-and-white and color images processed using the combined JPEG + WDT conversion and black-and-white and color images of the same quality, compressed and saved in JPEG format. The experiment was performed for single and double wavelet transformation at the following quality values of JPEG and JPEG + WDT images: 70, 80, 90 and 100%.

2. Size ratio of black-and-white and color images processed using the combined JPEG + WDT conversion and the same quality black-and-white and color images saved only in JPEG format with a fixed image quality value of 100%. The experiment was performed for single and double wavelet transformation at the following quality values of JPEG + WDT images: 70, 80, 90 and 100%.

Table 1

Comparison of the results of JPEG conversion and JPEG + WDT conversion

Size	*Iterations*	*Quality, %*	*Size JPEG + WDT, B*	*Size JPEG, B*	$t_{compression\ JPEG}$, *ms*	$t_{wavelet}$, *ms*	*PSNR, дБ*	*SSIM*
160x120	1	100	11987	15762	57	502	37,15	0,98
160x120	1	90	3333	4760	42	491	35,01	0,97
160x120	1	80	2242	3270	39	474	35,92	0,98
160x120	1	70	1894	2634	35	473	34,88	0,98
160x120	2	100	9621	15793	57	934	27,78	0,85
160x120	2	90	2514	4668	42	910	26,91	0,82
160x120	2	80	1695	3228	34	908	26,33	0,79
160x120	2	70	1425	2631	39	908	28,49	0,90

Analysis of the results of the experiment. Tables 1 and 2 show for all experiments the following data: input image size in pixels, Haar wavelet iteration number, JPEG quality in percent, JPEG + WDT image size and JPEG image size, time JPEG image compression, Haar wavelet image compression time, *PSNR* peak signal-to-noise ratio for Haar transform, and Haar transform *SSIM* structural similarity index, which can range from 0 to 1, where 1 indicates complete similarity between original and reconstructed /compressed image to evaluate the compression quality. *PSNR* values are measured in decibels (dB) and are typically in the range of 20 to 50 dB.

The results of the first experiment are shown in Table 1, where the data is obtained for images with the same level of JPEG quality. As can be seen from the table, an additional wavelet transformation with one iteration at a quality of 100% reduces the image size by approximately 25%; for two iterations of the wavelet transformation the image is compressed by approximately 50%. *PSNR* values are close to 40dB, which is considered very well, as well as SSIM values close to one.

Table 2

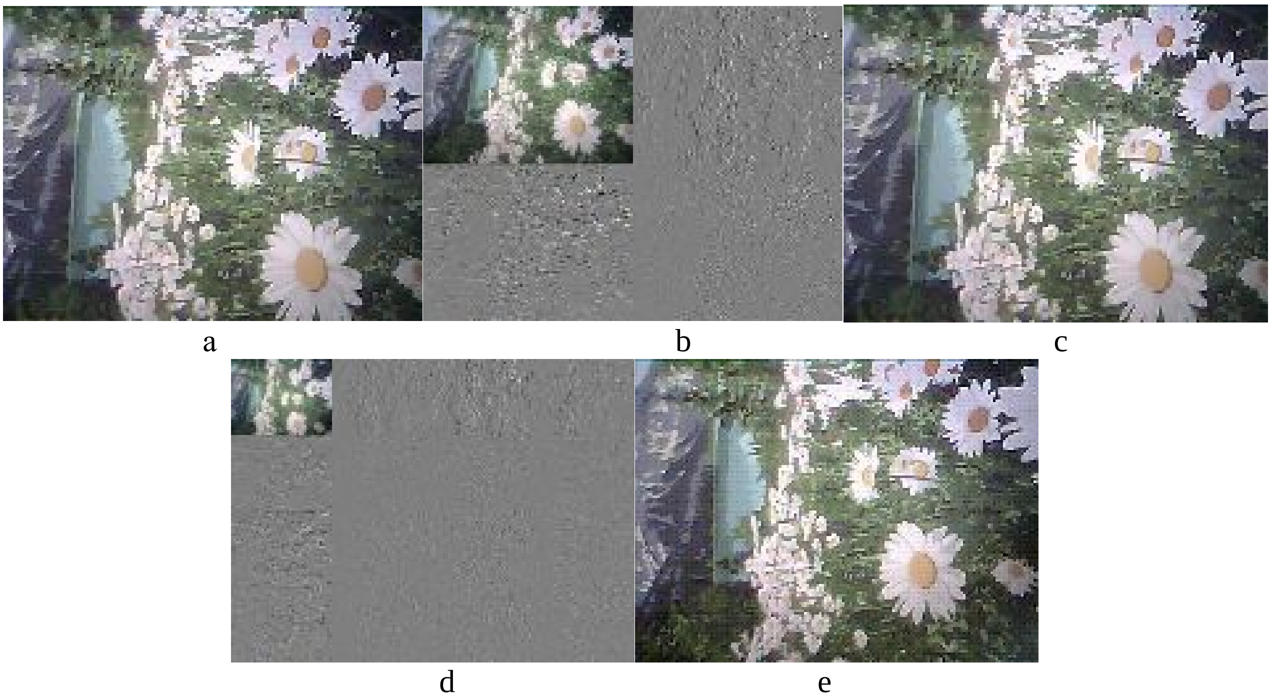
Comparison of JPEG conversion results at 100% constant quality and JPEG + WDT conversion

Size	***Iterations***	***Quality JPEG + WDT, %***	***Quality JPEG, %***	***Size JPEG + WDT, Б***	***Size JPEG, Б***	$t_{compression\ JPEG}$, ***ms***	$t_{wavelet}$, ***ms***	***PSNR***	***SSIM***
160x120	1	90	100	3293	15769	58	491	32,68	0,94
160x120	1	80	100	2251	15753	59	474	31,91	0,94
160x120	1	70	100	1865	15721	54	473	30,86	0,93
160x120	1	60	100	1589	15468	56	489	30,12	0,92
160x120	2	90	100	2490	15629	58	925	27,07	0,81
160x120	2	80	100	1674	15583	56	924	26,49	0,79
160x120	2	70	100	1421	15563	55	907	28,28	0,89
160x120	2	60	100	1284	15551	58	923	27,87	0,88

The results of the second experiment for a picture with JPEG compression at 100% quality, and for a picture with JPEG + WDT compression at 90% to 60% quality are listed in Table 2. Quality values below 60% are inappropriate for conducting the experiment. The results show that reducing the quality of JPEG has almost no effect on the legibility of the image, as the PSNR and SSIM coefficients remain at a satisfactory level but with smaller picture size parameters. For the lowest quality, the difference in file sizes is about 10 times for one iteration of the Haar wavelet, while for two iterations of the Haar wavelet, the difference in image size is almost 12 times.

Figure 4 shows an image saved in JPEG format (a), processed using the forward Haar transform + JPEG (b), reconstructed using the inverse Haar transform (c) for one iteration, and the direct (d) and inverse (e) Haar transform for two iterations.

The image from the camera in RGB888 format, saved in JPEG format, is presented in Fig. 4a. The same image in RGB888 format was processed using the Haar wavelet transform, which made it possible to reduce the size of the image itself and save only the most important details, and saved in JPEG format, presented in Fig. 4b. The Haar wavelet function was used to highlight important functional elements of the image. The inverse Haar transform process restores the image to near-original quality by restoring the details of the compressed image based on the reduced wavelet transform coefficients, as shown in Fig. 4 in. Although the quality of the restored image will be worse in any case due to the loss of information due to JPEG compression. Two iterations of the Haar transformation make it possible to reduce the size of the compressed file several times, but the artifacts on the reconstructed image become more noticeable, which confirms the values of the PSNR and SSIM coefficients.


a b c

d e

Fig. 4. **Image comparison at 80% JPEG quality. Image saved in JPEG format (a), transformed image for one iteration (b), reconstructed image for one iteration (c) transformed image for two iterations (d) reconstructed image for two iterations (e)**

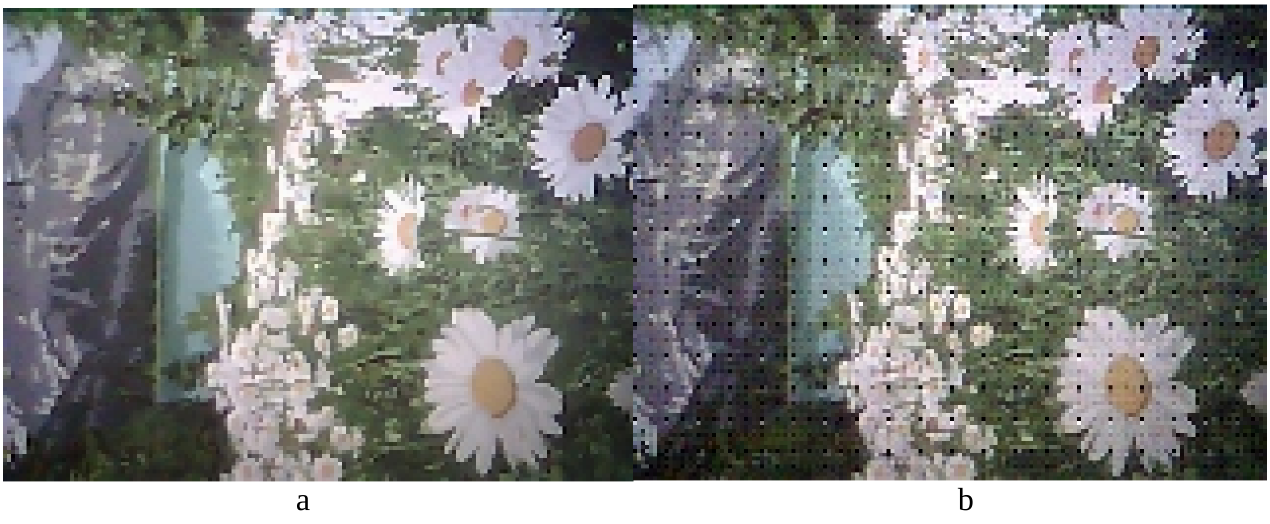

a b

Рис. 5. **Image saved in JPEG format (a) and restored image for four iterations of the Haar transform (b)**

In fig. 5 presents images that clearly show the appearance of artifacts in the image after the inverse Haar transform, because black points or artifacts may appear after the inverse Haar transform due to some factors, such as the number of iterations or incorrect data encoding. One of the reasons is the incorrect restoration or loss of some details during the inverse Haar transform. Iterative application of the transformation can lead to loss of information, especially in small details or noisy components of the image. This can result in black dots or artifacts. In addition, incorrect data encoding or inverse transformation errors can also affect the quality of the restored image. For example, if the pixel values were not correctly stored or encoded, this could result in the pixels being incorrectly restored after the inverse transformation. This is due to the errors that accumulate in the calculations for the selected data types used in the Haar wavelet transform algorithm.

Advantages of using wavelet compression in conjunction with JPEG over regular JPEG compression include better preservation of detail, as wavelet transforms preserve image features more accurately, such as fine lines, textures, and contours. Due to the using of wavelet transform, it is possible to provide higher image quality with less loss of details. In addition using of wavelet provides file size reduction, because the wavelet transform allows achieving a higher degree of Huffman compression, which is part of the JPEG format, and a file size reduction compared to ordinary JPEG. This is especially important for processing and transmitting images from systems with limited resources. Wavelet compression can reduce artifacts such as block structure and JPEG compression loss, resulting in cleaner, smoother images with less distortion. It is worth noting that wavelet compression has its limitations, such as the complexity of processing and setting parameters, which leads to an increase in image conversion time, because wavelet conversion has a more complex mathematical algorithm. In addition, the effectiveness of wavelet compression depends on the properties of a specific image and quality requirements. Therefore, before using wavelet compression, it is important to conduct experiments and evaluate its effectiveness in a specific application.

Conclusions. As a result of experimental studies, it was determined that the use of wavelet transformations for image processing for the purpose of transmission in microcontroller systems has its advantages, namely, it allows to achieve a high degree of image compression with minimal loss of quality. As determined by the experiments, the compression can reach 2–9 times, and the *SSIM* parameters are approximately equal to 0.94–0.92, as well as the *PSNR* is approximately equal to 30dB, which means an acceptable image quality, compared to the use of the usual JPEG algorithm. This is especially important in sensor systems on microcontrollers, where energy resources and time for image transmission are limited. The wavelet transform allows you to preserve a significant amount of information while reducing the file size, namely, better preserving image details, including textures, contours, and fine details. Which allows you to transmit images with high resolution and accuracy, which is also important for microcontroller systems with cameras. In addition, wavelet transforms performed in real time on microcontrollers can be improved with appropriate algorithms and optimization, which is the subject of further research.

References

1. Doorwar, M., Malathi, P. (2018). Comparison of Protocol Based on Architecture in Wireless Multimedia Sensor Network. *2018 IEEE Global Conference on Wireless Computing and Networking* (GCWCN), P. 39–43. DOI: 10.1109/GCWCN.2018.8668650.

2. Mateen, A., Abbas, K., Sehar, M., Akbar, M. A. (2017). Comparative Analysis of Wireless Sensor Networks with Wireless Multimedia Sensor Networks. *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering* (ICPCSI-2017), P. 80–83. DOI: 10.1109/ICPCSI.2017.8391847.

3. Hussein, W. A., Ali, B. M., Rasid, M. F. A., Hashim, F. (2017). Design and Performance Analysis of High Reliability-optimal Routing protocol for Mobile Wireless Multimedia Sensor Networks. *IEEE 13th Malaysia International Conference on Communications* (MICC), 28–30 Nov. 2017, The

Література

1. Doorwar M., Malathi P. Comparison of Protocol Based on Architecture in Wireless Multimedia Sensor Network. *2018 IEEE Global Conference on Wireless Computing and Networking* (GCWCN). 2018. P. 39–43. DOI: 10.1109/GCWCN.2018.8668650.

2. Mateen A., Abbas K., Sehar M., Akbar M. A. Comparative Analysis of Wireless Sensor Networks with Wireless Multimedia Sensor Networks. *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering* (ICPCSI-2017). 2017. P. 80–83. DOI: 10.1109/ICPCSI.2017.8391847.

3. Hussein W. A., Ali B. M., Rasid M. F. A., Hashim F. Design and Performance Analysis of High Reliability-optimal Routing protocol for Mobile Wireless Multimedia Sensor Networks. *IEEE 13th Malaysia International Conference on Communications* (MICC), 28–30 Nov. 2017. The

Puteri Pacific, Johor Bahru, Malaysia, P. 136–140. DOI: 10.1109/MICC.2017.8311747.

4. Inaba, S. (2018). 3D Flash Memory for Data-intensive Applications. *2018 IEEE International Memory Workshop* (IMW). 2018. P. 1–15. DOI: 10.1109/IMW.2018.8388775.

5. Rein, S., Reisslein, M. (2011). Performance evaluation of the fractional wavelet filter: A low-memory image wavelet transform for multimedia sensor networks. *Ad Hoc Networks*, 2011. DOI: 10.1016/j.adhoc.2010.08.004.

6. Tausif, M., Jain, A., Khan, E., Hasan, M. (2020). Low Memory Architectures of Fractional Wavelet Filter for Low-Cost Visual Sensors and Wearable Devices. *IEEE Sensors Journal*, Vol. 20, No. 13, P. 6863–6871. DOI: 10.1109/JSEN.2019.2930006.

7. Ahmed, A. J., Hamdi, M. M., Mustafa, A. S., Rashid, S. A. (2022). WSN Application Based on Image Compression Using AHAAR Wavelet Transform. *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications* (HORA), Ankara, Turkey, 2022, P. 1–4. DOI: 10.1109/HORA55278.2022.9799842.

8. Sharma, A., Tandan, H., Goel, P. (2017). Energy efficient handling of DWT-image transmission over wireless sensor network. *2017 Fourth International Conference on Image Information Processing* (ICIIP), Shimla, India, 2017, P. 1–5. DOI: 10.1109/ICIIP.2017.8313689.

9. Tausif, M., Jain, A., Khan, E., Hasan, M. (2018). Efficient Architectures of Fractional Wavelet Filter (FrWF) for Visual Sensors and Wearable Devices. *2018 IEEE SENSORS*, New Delhi, India, 2018, P. 1–4. DOI: 10.1109/ICSENS.2018.8630300.

10. Tausif, M., Khan, E., Hasan, M. (2019). Lifting-based fast and low memory DWT computation for IoT platform. *TENCON 2019 – 2019 IEEE Region 10 Conference* (TENCON), Kochi, India, 2019, P. 435–440. DOI: 10.1109/TENCON.2019.8929533.

11. Wang, C., Zhang, Y., Zhou, X. (2018). Robust Image Watermarking Algorithm Based on ASIFT against Geometric Attacks. *School of Mechanical, Electrical and Information Engineering*, Shandong University, Weihai 264209, China, 2018, P. 1–20. DOI: 10.3390/APP8030410.

12. ESP32-CAM camera development board. *Ai-Thinker Technology*. URL: https://docs.ai-thinker.com/en/esp32-cam.

13. ESP32 Camera Driver. *espressif.com*. URL: https://github.com/espressif/esp32-camera/tree/master/conversions.

Puteri Pacific, Johor Bahru, Malaysia, 2017. P. 136–140. DOI: 10.1109/MICC.2017. 8311747.

4. Inaba S. 3D Flash Memory for Data-intensive Applications. *2018 IEEE International Memory Workshop* (IMW). 2018. P. 1–15. DOI: 10.1109/IMW.2018.8388775.

5. Rein S., Reisslein M. Performance evaluation of the fractional wavelet filter: A low-memory image wavelet transform for multimedia sensor networks. *Ad Hoc Networks*. 2011. DOI: 10.1016/j.adhoc.2010.08.004.

6. Tausif M., Jain A., Khan E., Hasan M. Low Memory Architectures of Fractional Wavelet Filter for Low-Cost Visual Sensors and Wearable Devices. *IEEE Sensors Journal*. 2020. Vol. 20, No. 13. P. 6863–6871. DOI: 10.1109/JSEN.2019.2930006.

7. Ahmed A. J., Hamdi M. M., Mustafa A. S., Rashid S. A. WSN Application Based on Image Compression Using AHAAR Wavelet Transform. *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications* (HORA). Ankara, Turkey, 2022. P. 1–4. DOI: 10.1109/HORA55278.2022.9799842.

8. Sharma A., Tandan H., Goel P. Energy efficient handling of DWT-image transmission over wireless sensor network. *2017 Fourth International Conference on Image Information Processing* (ICIIP). Shimla, India, 2017. P. 1–5. DOI: 10.1109/ICIIP.2017.8313689.

9. Tausif M., Jain A., Khan E., Hasan M. Efficient Architectures of Fractional Wavelet Filter (FrWF) for Visual Sensors and Wearable Devices. *2018 IEEE SENSORS*. New Delhi, India, 2018. P. 1–4. DOI: 10.1109/ICSENS.2018.8630300.

10. Tausif M., Khan E., Hasan M. Lifting-based fast and low memory DWT computation for IoT platform. *TENCON 2019 – 2019 IEEE Region 10 Conference* (TENCON). Kochi, India, 2019. P. 435–440. DOI: 10.1109/TENCON.2019.8929533.

11. Wang C., Zhang Y., Zhou X. Robust Image Watermarking Algorithm Based on ASIFT against Geometric Attacks. *School of Mechanical, Electrical and Information Engineering*. Shandong University, Weihai 264209, China. 2018. P. 1–20. DOI: 10.3390/APP8030410.

12. ESP32-CAM camera development board. *Ai-Thinker Technology*. URL: https://docs.ai-thinker.com/en/esp32-cam.

13. ESP32 Camera Driver. *espressif.com*. URL: https://github.com/espressif/esp32-camera/tree/master/conversions.

MAZIN M. Yu.
Post graduate student,
Department of Acoustic and
Multimedia Electronic Systems,
National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnic Institute", Ukraine
https://orcid.org/0000-0001-9566-6662
E-mail: maksimalnuu@ukr.net

ONYKIIENKO Yu.O.
PhD, Associate Professor,
Department of Acoustic and Multimedia Electronic
Systems, National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnic Institute", Ukraine
https://orcid.org/0000-0001-7508-8391
Scopus Author ID: 55819618700
Researcher ID: J-6589-2017
E-mail: yurionik@gmail.com

МАЗІН М. Ю., ОНИКІЄНКО Ю. О.
Національний технічний університет України "Київський політехнічний інститут
імені Ігоря Сікорського" Київ, Україна

ВИКОРИСТАННЯ ВЕЙВЛЕТ-ПЕРЕТВОРЕНЬ ДЛЯ ОБРОБКИ ЗОБРАЖЕНЬ В СИСТЕМАХ ІНТЕРНЕТУ РЕЧЕЙ НА МІКРОКОНТРОЛЕРАХ

Мета*: Дослідження ефективності використання вейвлет-перетворення Хаара, як додаткового до JPEG перетворення, для стиснення зображень у контексті обмежених ресурсів мікроконтролерів, подальшого зменшення часу передачі зображення та, відповідно, зменшення витрат енергії.*

Методика*: Експериментальне дослідження якості зображень, оброблених за допомогою вейвлет-перетворення Хаара та стиснених за допомогою JPEG перетворення мікроконтролерним модулем ESP32-CAM, який здійснює захоплення зображень вбудованою камерою. Оцінка якості отриманих зображень, порівняння зображень стиснутих за допомогою JPEG та з вейвлет перетворенням плюс JPEG, розрахунок коефіцієнтів Peak Signal-to-Noise Ratio та Structural Similarity Index Measure.*

Результати*: Створено програмний модуль для реалізації вейвлет-перетворення Хаара на мікроконтролері ESP32-CAM. Досліджена ефективність використання вейвлет-перетворення Хаара, як додаткового методу до JPEG, для стиснення та передачі зображень використовуючи обмежені ресурси мікроконтролера, оцінені розмір та якість зображень шляхом порівняння розміру стиснутих зображень, а також якості зображень візуально та за допомогою розрахунку коефіцієнтів PSNR та SSIM. За результатами встановлено, що додаткове використання вейвлет-перетворення Хаара дає зменшення розміру зображення від 25% до декількох разів в залежності від коефіцієнта якості JPEG без значних втрат якості зображення, хоча час обробки зображення збільшується до 1 с.*

Наукова новизна*: Запропоновано комбінований підхід для стиснення зображень за допомогою декількох способів обробки відео зображень з урахуванням можливостей 32-х бітних мікроконтролерів з додатковою зовнішньою пам'яттю.*

Практична значимість*: Запропонований спосіб обробки зображень може забезпечити зменшення часу передачі самих зображень та зменшити споживання енергії при виконанні перетворень і передачі зображень, що дасть змогу подовжити термін служби батареї сенсорного вузла. При цьому даний підхід до стиснення зображення може забезпечити якість вихідного зображення з найменшими можливими втратами даних за рахунок зменшення кількості візуальної інформації, як втрачається під час стиснення.*

Ключові слова*: стиснення зображення; вейвлет-перетворення; вейвлет Хаара; PSNR; SSIM.*