DOI: 10.2352/ISSN.2470-1173.2016.15.IPAS-196

# JPEG compression with recursive group coding

**Nadezhda Kozhemiakina[a], Vladimir V. Lukin[a], Nikolay N. Ponomarenko[a], Jaakko Astola[b] and Karen O. Egiazarian[b]**
[a] **National Aerospace University, 61070, Kharkov, Ukraine;**
[b] **Tampere University of Technology, FIN 33101, Tampere, Finland**

## Abstract

*A task of alternative (faster and more efficient in compression ratio sense) coding of discrete cosine transform (DCT) coefficients within JPEG based image compression approach is considered. In the data processing chain, it is proposed to apply a recursive group coding (RGC) as an alternative to arithmetic or Huffmann coding. In contrast to the aforementioned data coding techniques, the RGC method is able to efficiently code symbols of very large alphabets (each block of 8x8 pixels of quantized DCT coefficients can be represented as such 64-byte or 128-byte symbol). Comparative analysis of efficiency for the standard JPEG and its proposed modification (for three images of three different digital cameras) is carried out using six different quantization tables. It is shown that RCG possesses low computational complexity and a high speed of compression simultaneously with higher compression ratio (CR) compared to the standard JPEG. The benefit in CR appears to be larger for smaller quantization steps (QSs) that mainly correspond to SHQ (super high quality) mode. This benefit can reach up to 10%. It is also demonstrated that the benefits exists for uniform quantization tables. The proposed coding method can be used for an additional compression of JPEG-images in coding traffic of communication lines. For this purpose, data of JPEG-images have to be partly decoded (till the level of the quantized DCT coefficients) and then recoded by RGC.*

*Keywords: JPEG compression, additional JPEG compression, entropy coding, recursive group coding.*

## Introduction

The task of lossy image compression continues to be of a prime interest for several decades. Among a widely applied methods of lossy compression, it is worth mentioning techniques based on DCT [1] (used as a core of the standard JPEG [2]) and on discrete wavelet transform (DWT) put into a basis of the later introduced standard JPEG2000 [3]. Currently, the methods based on orthogonal transforms provide reasonable compromise between compression ratio, image quality, and speed of compression and decompression. Methods based on other approaches such as fractal compression or vector quantization have considerably less practical applications.

A popular modern trend in image compression is to take into account visual quality of compressed images, to provide a desired visual quality or invisibility of introduced distortions [4, 5]. These techniques deal with design and testing of special metrics (indices) of image visual quality [6, 7]. In this sense, compression methods based on DCT are also among the best [8]. The main reason is that the considered techniques are able to easily incorporate a knowledge on peculiarities of human visual system by using non-uniform quantization tables where DCT coefficients corresponding to higher spatial frequencies are quantized with larger QSs. Noise characteristics can be taken into account at noisy image compression stage as well to improve visual quality [9]. In particular, noise characteristics can be taken into consideration in compression of Bayer pattern color filter array (RAW) images available in many modern digital cameras [10].

There are also many specific applications of (lossy) image compression such as remote sensing of Earth and other planets, medical imaging, etc. For some of them, the main criterion of efficiency is how well data interpreting or information retrieval tasks contained in images are solved (often in automatic or semi-automatic modes). For such applications, it can be worth using uniform quantization tables with rather small values of QS.

Methods of compressing still images continue to be the basis of compressing multichannel images and video [11], where, alongside with accounting intra-frame redundancy, inter-frame or inter-component redundancy is incorporated to improve compression efficiency. DCT is widely used for these applications as well.

Although the standard JPEG was designed and introduced more than twenty years ago and later more efficient methods, including the standard JPEG2000, have been proposed, all these methods occurred to be unable to substitute JPEG in many applications including devices of everyday life. One reason is that JPEG is supported by numerous software tools and platforms. Another important factor is that JPEG is pure in patent sense that allows manufacturers to use it without a risk of patent conflicts. More than 80% of images in Internet are still stored and transferred using JPEG format. Moreover, JPEG continues to be the most widely used standard (format) of storing images acquired by digital cameras and other modern imaging devices. Due to this, a special attention is still paid to such tasks as additional compression of JPEG images [12] for their storage or transfer via communication networks, design of different modifications of JPEG oriented on compressing particular (specific) types of images represented as 16-bit data arrays.

Note, that JPEG-like compression can be conditionally divided into two stages: 1) removal of the spatial correlation in data and image simplification (compact representation) in the DCT domain in image blocks and further quantization of DCT coefficients; 2) removal of the statistical redundancy in obtained data by a procedure that includes Zig-Zag scanning, run length encoding (RLE) and arithmetic or Huffman coding. Zigzag scanning and RLE are useful for coding large number of zeroes in quantized DCT coefficients. Without these operations, further applied arithmetic or Huffman coding are not efficient since quantized DCT coefficients in image blocks are not heterogeneous in the statistical sense. There are much more zeroes for coefficients that relate to high spatial frequencies that in low frequency DCT coefficients. However, it is also possible to consider 8x8 blocks of the quantized DCT coefficients as a specific type of data (alphabet symbols). Then, they are quite similar to each other according to their statistical characteristics. At the beginning of each symbol, there is less zero bytes whilst there are more zero bytes at the end. Then, if quantized coefficients are coded not as separate bytes but as large symbols, it becomes possible to avoid Zig-Zag scanning

©2016 Society for Imaging Science and Technology
DOI: 10.2352/ISSN.2470-1173.2016.15.IPAS-196

and RLE which require considerable computations and restrict potential efficiency of encoding. However, neither arithmetic nor Huffman coding used in the JPEG allow efficient coding of large alphabet symbols.
Meanwhile, this task can be effectively solved by the recently proposed coding technique called RGC [13]. Computational complexity of this method is low since the main cycle of coding can be realized using only shift and logical "OR" operations. Due to the recursive scaling of data, it provides an efficient coding of the texts of large alphabet symbols without any a priori information on their statistical characteristics [13, 14].
The main goal of this work is to study modifications of JPEG where the second stage of compression is entirely carried out by RGC. An efficiency analysis and comparisons are performed for both cases of uniform and non-uniform quantization of DCT coefficients.

## RGC Use in JPEG Compression

The principle of RGC is illustrated in Fig. 1.

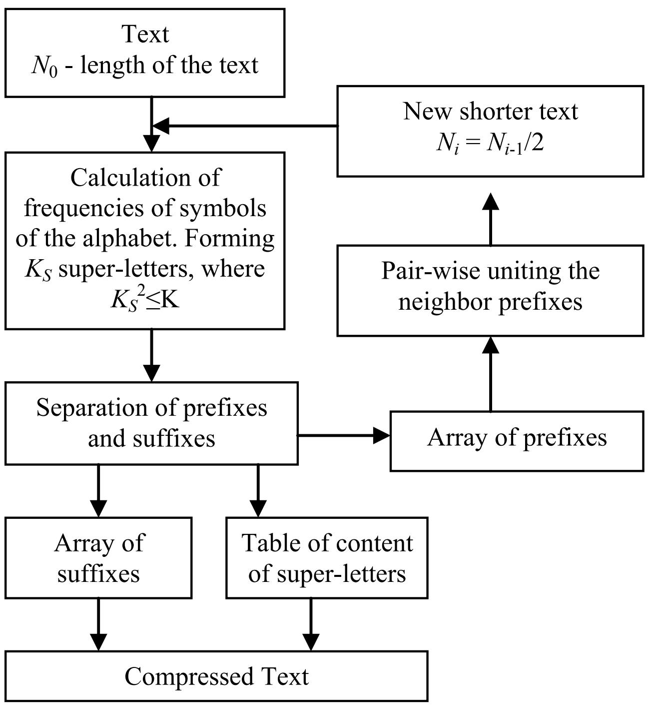


***Figure 1.*** *Block-diagram of RGC*

At each iteration of RGC, coding of $k$-th symbol of an alphabet (denote the total number of symbols in the considered alphabet as $K$) is performed by estimating frequency of this symbol appearance $p_k$ in a coded text. Then, all symbols of the alphabet are divided into $K_S$ super-letters (groups of symbols) with close values of $p_k$. Next, symbol suffixes (order numbers inside super-letters) are saved in a compressed file whilst prefixes (super-letter numbers) are united in the pairwise manner. After this, the formed new text (which is twice shorter than the original one) is passed to the next iteration of the coding process. To avoid increasing the alphabet size from iteration to iteration, the condition $K \ge K_S^2$ should be satisfied.
An increase of the code length due to combination of symbols into a super-letter is expressed in the relative units as

$$\Delta = -p_E(-\log_2 p_E + \log_2 M) / \sum_{k=1}^{M} (p_k \log_2 p_k), \qquad (1)$$

where $M$ is a number of symbols united in the super-letter, $p_E$ denotes a sum of probabilities of these symbols determined as $\sum_{k=1}^{M} p_k$.
The value of $\Delta$ depends on homogeneity (in the statistical sense) of symbols united into a super-letter: more homogeneous - less a value of $\Delta$ is.
Time $W_e$ spent by RGC for coding a text can be estimates by the following expression:

$$W_e = N_0(2W_a + 2W_t + W_s + W_o), \qquad (2)$$

where $W_a$ is a time necessary for summation operation (increased by unity), $W_t$ denotes a time of the operation of extraction from a table, $W_s$ is a time of logical shift operation, $W_o$ defines a time for executing logical "OR" operation.
In turn, execution time required for decoding is expressed as:

$$W_d = N_0(2W_t + W_s + W_o). \qquad (3)$$

As it is seen, the expression (3) (compared to the expression (2)) does not contain the term $2W_a$, which makes decoding to be about 30% faster than encoding (the benefit depends upon peculiarities of command and operations execution in the used processor).
Fig. 2 presents the proposed block-diagram of substituting the aforementioned part of JPEG compression by RGC. A specific feature of coding for the considered application is that a dynamic range of quantized DCT coefficients might exceed 256. Then, to use RGC we need to choose a strategy of coding such coefficients. First, it is possible to code 16-bit values of DCT coefficients (one symbol of the used alphabet occupies 128 bytes). Second, it is possible to carry out a preliminary separation of data into more significant and less significant bytes and to code these two data arrays separately.
We prefer the second strategy by using one more preliminary operation (before separating data into more significant and less significant parts). This is the operation of adding 128 to each quantized DCT coefficient. This operation makes most of quantized DCT coefficients positive with the mean value about 128. Then, most of information is concentrated in the array of less significant values.

## Numerical analysis

To carry out comparative analysis of the conventional and proposed modifications of JPEG, let us use a set of test color images (earlier exploited in the paper [15]) which is represented in Fig. 3 (all these images have been saved in the format TIFF without compression). Let us compress them using the conventional JPEG and the proposed modification (further referred as JPEG-RGC).
Table 1 presents the results of compressing the test images for three different quantization tables that correspond to high, middle and low quality of JPEG compressed images (the parameter "Quality" equals to 100, 80, and 50, respectively).

DOI: 10.2352/ISSN.2470-1173.2016.15.IPAS-196

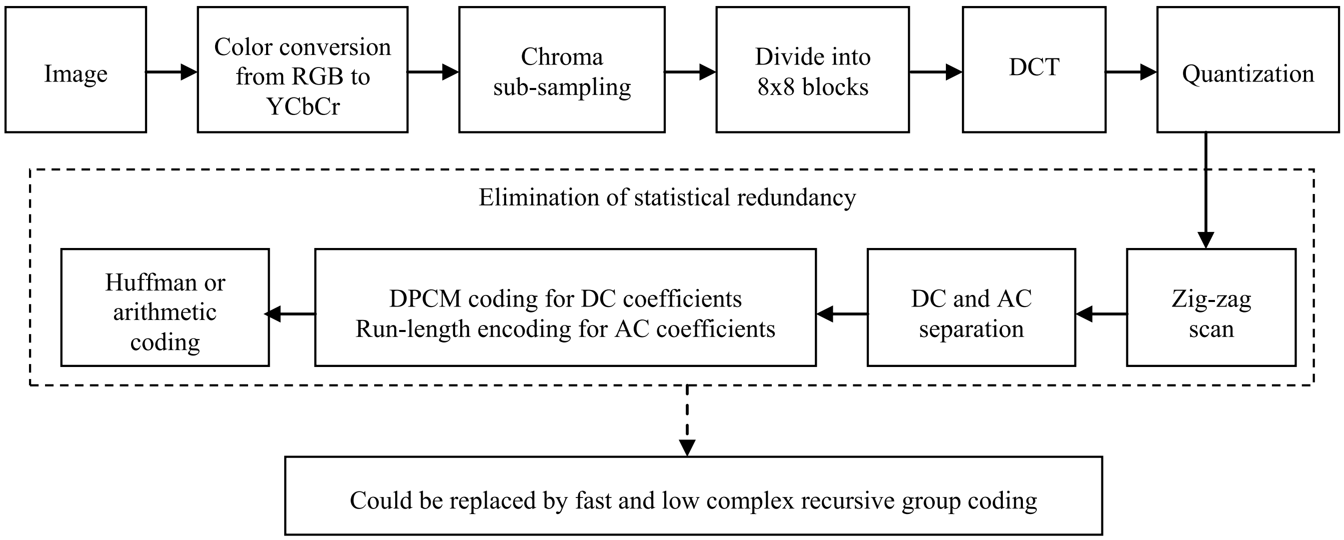


***Figure 2.*** *Block-diagram of using RGC in JPEG instead of the complex procedure of removing statistical redundancy in quantized DCT coefficients*

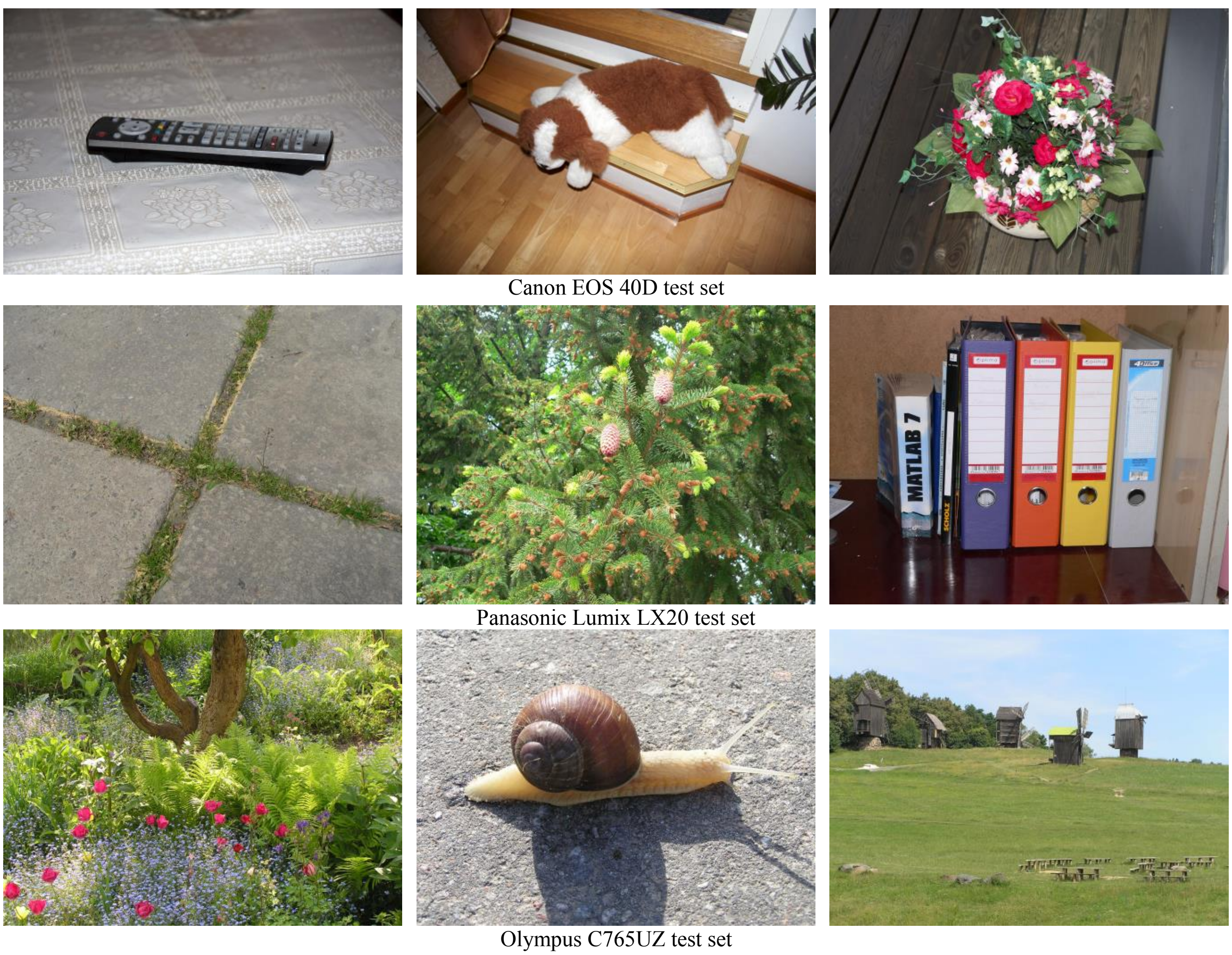


***Figure 3.*** *Test set for numerical analysis*

©2016 Society for Imaging Science and Technology
DOI: 10.2352/ISSN.2470-1173.2016.15.IPAS-196

Analysis of these data shows that JPEG-RGC provides the maximal benefit (on the average, 10%) compared to the conventional JPEG for "Quality"=100 and for highly textural images (the maximal benefit is observed for the first test image from the set Olympus C765UZ). For larger CR, the method RGC-JPEG provides CR comparable to the conventional JPEG. Table 2 represents similar data for uniform quantization tables. As it is seen, the method RGC-JPEG again has certain benefits compared to JPEG for "Quality"=100 although the benefits in this case are smaller and are about 5%.

***Table 1. bpp for conventional JPEG quantization tables***

| Test set | Image № | Quality=100 | | Quality=80 | | Quality=50 | |
|---|---|---|---|---|---|---|---|
| | | JPEG | JPEG-RGC | JPEG | JPEG-RGC | JPEG | JPEG-RGC |
| Canon EOS 40D | 1 | 6.25 | 5.63 | 1.06 | 1.04 | 0.49 | 0.49 |
| | 2 | 5.91 | 5.38 | 0.91 | 0.92 | 0.39 | 0.43 |
| | 3 | 6.05 | 5.52 | 1.01 | 1.01 | 0.47 | 0.49 |
| Panasonic Lumix LX20 | 1 | 7.41 | 6.61 | 1.71 | 1.70 | 1.00 | 0.94 |
| | 2 | 8.11 | 7.19 | 1.97 | 1.94 | 1.18 | 1.16 |
| | 3 | 5.55 | 5.17 | 0.95 | 0.95 | 0.49 | 0.52 |
| Olympus C765UZ | 1 | 9.50 | 8.09 | 2.68 | 2.63 | 1.63 | 1.63 |
| | 2 | 7.73 | 6.86 | 2.05 | 2.05 | 1.25 | 1.25 |
| | 3 | 5.86 | 5.50 | 1.25 | 1.30 | 0.73 | 0.75 |
| Average | | 6.93 | 6.22 | 1.51 | 1.50 | 0.85 | 0.85 |

***Table 2. bpp for uniform quantization tables***

| Test set | Image № | Quality=100 | | Quality=80 | | Quality=50 | |
|---|---|---|---|---|---|---|---|
| | | JPEG | JPEG-RGC | JPEG | JPEG-RGC | JPEG | JPEG-RGC |
| Canon EOS 40D | 1 | 9.48 | 9.21 | 6.63 | 6.44 | 3.55 | 3.53 |
| | 2 | 9.20 | 9.05 | 6.31 | 6.27 | 3.28 | 3.36 |
| | 3 | 9.45 | 9.23 | 6.58 | 6.43 | 3.50 | 3.54 |
| Panasonic Lumix LX20 | 1 | 10.23 | 9.66 | 7.23 | 7.00 | 4.64 | 4.50 |
| | 2 | 11.93 | 11.27 | 8.82 | 8.42 | 6.02 | 5.83 |
| | 3 | 8.30 | 7.90 | 5.35 | 5.23 | 3.01 | 3.05 |
| Olympus C765UZ | 1 | 12.71 | 12.14 | 9.67 | 9.31 | 7.00 | 6.79 |
| | 2 | 9.63 | 9.22 | 6.85 | 6.61 | 4.71 | 4.62 |
| | 3 | 7.69 | 7.47 | 4.92 | 4.92 | 3.09 | 3.19 |
| Average | | 9.85 | 9.46 | 6.93 | 6.74 | 4.31 | 4.27 |

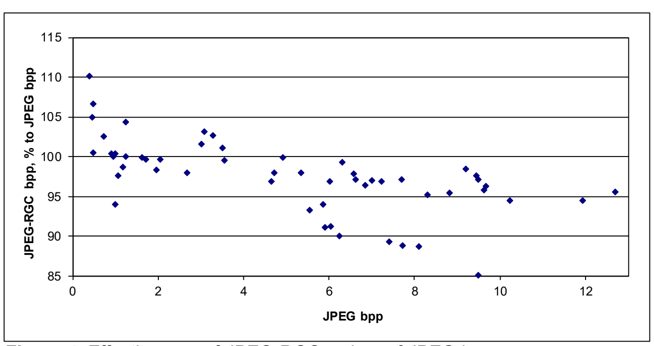


***Figure 4.*** *Effectiveness of JPEG-RGC vs bpp of JPEG image*

Fig. 4 presents the scatter-plot that jointly represents data for both Tables. Obviously, the proposed modification has advantages for large bpp values, i.e. for relatively small CR. The conventional JPEG, in turn, is slightly better for bpp smaller than 4.

To estimate efficiency of image compression by the proposed method, we have also employed a special test set of images that consists of 100 images randomly chosen at photo-hosting "Яндекс.Фотки" (Yandex-Photos). Average size of JPEG-compressed image is 2 MBytes.

Compression results have demonstrated the following. Among 100 considered test images, better compression has been gained for 68 images. Total decrease of memory needed for storing these 100 images is 3.5%.

Fig. 5 presents the scatter-plot that illustrates efficiency of compression by JPEG-RGC compared to JPEG for the considered test image set.

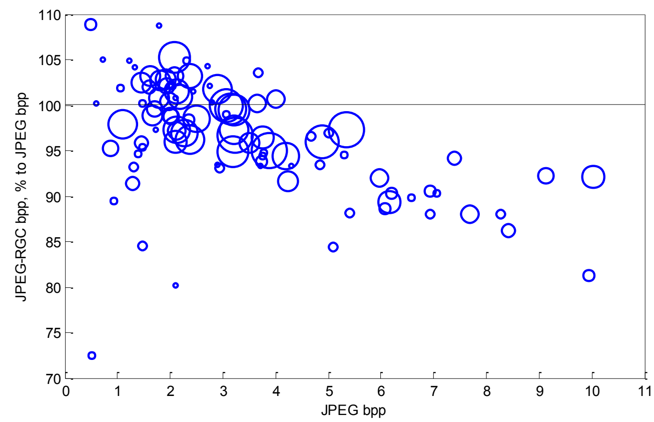


***Figure 5.*** *Efficiency of compression by JPEG-RGC depending on bpp of JPEG file for images of the analyzed test set*

Each circle at this scatter-plot corresponds to one test image where a circle diameter shows size of JPEG file for this image (it varied from 40 kBytes to 7 MBytes). One can see that efficiency of the proposed method practically does not depend on image file size. For example, large diameter circles are both over and under the horizontal line 100%. Meanwhile, there is an obvious dependence on bpp. If bpp exceeds 4, the proposed method always provides better CR compared to JPEG. This is in the good agreement with data in Fig. 4.

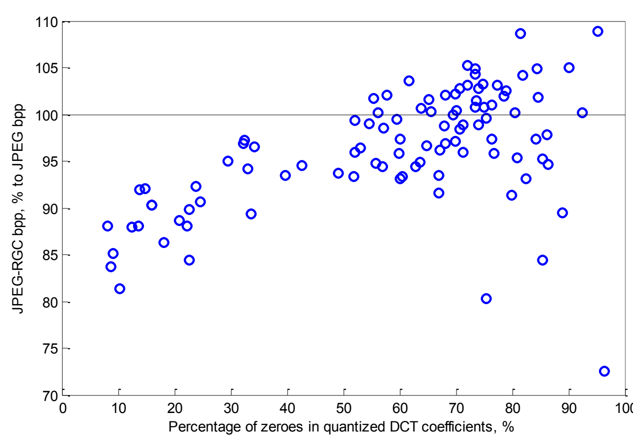


***Figure 6.*** *Efficiency of compression by JPEG-RGC depending upon percentage of zeros in quantized DCT coefficients*

DOI: 10.2352/ISSN.2470-1173.2016.15.IPAS-196

Usually compression ratio can also depend upon other factors. Let us study what other parameters (alongside with bpp) allow predicting when the proposed compression technique is more efficient. For this purpose, let us obtain and analyze the corresponding scatter-plots. Fig. 6 presents the scatter-plot of compression efficiency of JPEG-RGC depending on percentage of zeros in quantized DCT coefficients. The study has been performed for the same test set as for data in Fig. 5.

Analysis of data in Fig. 6 clearly shows that if the percentage of zeros in quantized DCT coefficients is smaller than 55%, the method JPEG-RGC occurs to be preferable compared to JPEG. Contrary, if number of zeros exceeds 55%, the methods are comparable.

Another factor able to influence coder performance could be image size expressed as number of pixels. It is possible to assume that, to provide benefits of the proposed coder, total number of quantized DCT coefficients should exceed some limit. Fig. 7 shows the scatter-plots of dependence of JPEG-RGC performance (compared to JPEG) on image size expressed in Mega pixels.

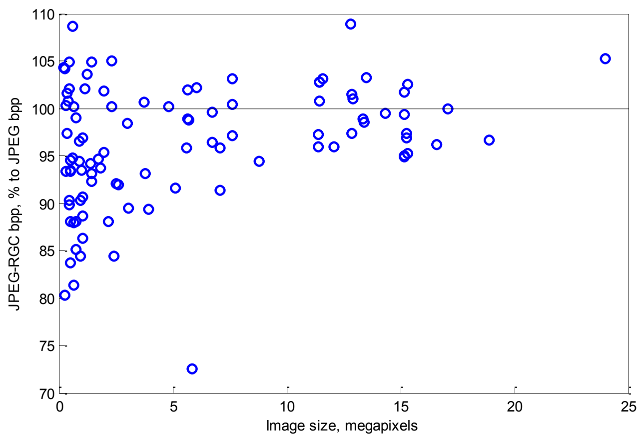


***Figure 7.** Efficiency of compression by JPEG-RGC depending upon image size expressed as number of pixels*

Analysis of the scatter-plot in Fig. 7 shows that, most probably, there is no connection between the proposed method benefits and the compressed image size. Perhaps, benefits of RGC-JPEG compared to JPEG appear themselves more often for small size images.

A next factor that might have impact is statistics of quantized DCT coefficients. To analyze it, let us calculate an estimate of DCT coefficients entropy for image blocks as:

$$E = \frac{1}{U}\sum_{n=1}^{U} log_2(1+|D_n|), \qquad (4)$$

where U is the number of image pixels (and, respectively, the total number of DCTY coefficients in 8x8 pixel blocks), $D_n$ denotes a value of the $n$-th DCT coefficient. Fig. 8 presents the scatter-plot characterizing JPEG-RGC performance depending on the parameter $E$ (4).

As it follows from the analysis of data in Fig. 8, the proposed coder JPEG-RGC is always preferable compared to JPEG if the parameter $E$ exceeds 3.

Finally, let us study a dependence of efficiency for JPEG-RGC on quantization tables used in compression. For this purpose, introduce a parameter $T$ calculated in the following manner. From quantization table (for intensity), let us select 4x4 fragment that corresponds to low spatial frequencies. Then, select the same fragment from quantization table for color components Cb and Cr. After this, average all the values for these two fragments. Fig. 9 shows the scatter-plot of JPEG-RGC performance depending on the introduced parameter $T$.

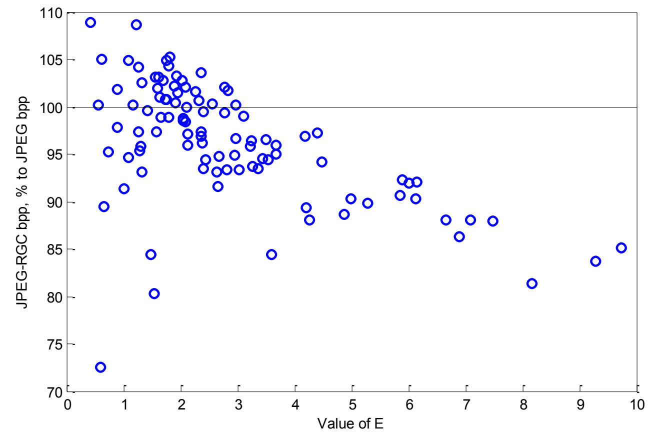


***Figure 8.** Performance (efficiency) pf JPEG-RGC depending on the parameter E*

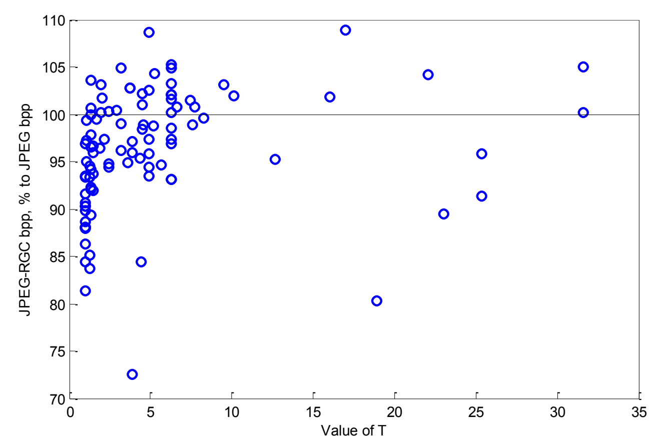


***Figure 9.** Efficiency of compression by JPEG-RGC depending on parameter T for images of the test set*

Analysis of the data in Fig. 9 allows concluding the following. JPEG-RGC is mostly preferable compared to JPEG for small quantization steps (when $T$ is quite small as well).

It is interesting that for the point corresponding to smallest JPEG-RGC/JPEG bpp ratio only this criterion is appropriate to predict a prefer ability of JPEG-RGC.

Fig. 10 presents enlarged fragment of the scatter-plot in Fig. 9 for $T$ values from 0 to 2.5.

It follows from analysis of data in Fig. 10 that advantage in performance of JPEG-RGC compared to JPEG is practically guaranteed for quantization steps close to unity, i.e., if an image is compressed preserving maximal quality.

Therefore, it is possible to conclude the following. First, if the task is to achieve additional compression of JPEG images providing better CR (e.g., in compressing specific traffic in channels of data transmission), the decision on re-compression using JPEG-RGC can be undertaken based on analysis of image bpp or quantization tables used (quantization tables can be extracted from compressed JPEG images without decoding them).

Second, if one deals with a necessity to compress images with high quality (i.e., using quantization steps close to unity), then the method JPEG-RGC clearly has certain advantages.

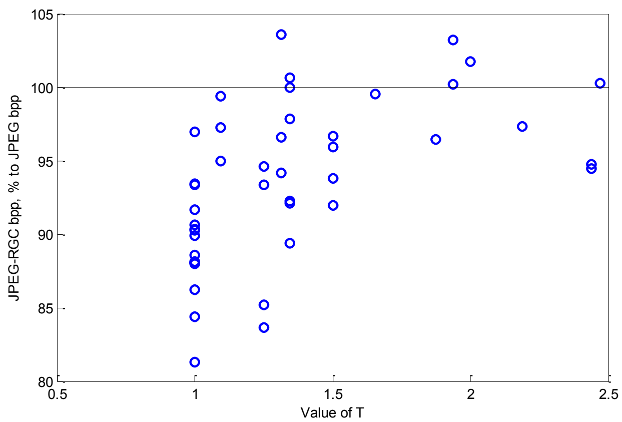


***Figure 10.*** *Efficiency of compression by JPEG-RGC depending on parameter T (enlarged fragment)*

Third, for middle and large values of quantization steps in blocks, it is possible to quickly estimate percentage of zero-valued quantized DCT-coefficients or the parameter $E$. If this percentage is smaller than 55% or $E$>3, then the use of the proposed JPEG-RGC is preferable.

## Conclusions

The carried out studies demonstrate that RGC can be efficiently used in compression schemes where it is needed to code blocks of quantized DCT coefficients with rather small values of quantization steps. This happens if requirements of compressed image quality (in particular, visual quality) are strict. Taking this into account, it seems perspective to apply RGC for other applications of data coding where one deals with the quantized DCT coefficients as this happens e.g. in audio and video coding.

## References

[1] N. Ponomarenko, V. Lukin, K. Egiazarian, J. Astola, "ADCT: A new high quality DCT based coder for lossy image compression" in Proceedings of LNLA, pp. 33-40, 2008.

[2] G.K. Wallace, "The JPEG Still Picture Compression Standard", Comm. of the ACM, Vol. 34, pp. 30-44, 1991.

[3] D. Taubman, M. Marcellin, JPEG 2000: Image Compression Fundamentals, Standards and Practice. Boston: Kluwer, 2002.

[4] N. Ponomarenko, S. Krivenko, V. Lukin, K. Egiazarian, J. Astola. "Lossy Compression of Noisy Images Based on Visual Quality: A Comprehensive Study", EURASIP Journal on Advances in Signal Processing, doi:10.1155/2010/976436, 13 p., 2010.

[5] V. Lukin, M. Zriakhov, N. Ponomarenko, S. Krivenko, M. Zhenjiang, "Lossy compression of images without visible distortions and its application", in Proceedings of IEEE International Conference on Signal Processing, pp. 698-701, 2010.

[6] N. Ponomarenko, L. Jin, O. Ieremeiev, V. Lukin, K. Egiazarian, J. Astola, B. Vozel, K. Chehdi, M. Carli, "Image database TID2013: Peculiarities, results and perspectives", Signal Processing: Image Communication, Vol. 30, pp. 55-77, 2015.

[7] N. Ponomarenko, S. Krivenko, V. Lukin, K. Egiazarian, "Visual quality of lossy compressed images", Proceedings of the 10th International Conference, CADSM 2009, pp. 137-142, 2009.

[8] V. Lukin, N. Ponomarenko, M. Zriakhov, A. Zelensky, K. Egiazarian, J. Astola, "Quasi-optimal compression of noisy optical and radar images", Proceedings of Image and Signal Processing for Remote Sensing XII, doi:10.1117/12.689557, 2006.

[9] V. Lukin, S. Abramov, N. Ponomarenko, M. Uss, M. Zriakhov, B. Vozel, K. Chehdi, J. Astola, "Methods and automatic procedures for processing images based on blind evaluation of noise type and characteristics", Journal of Applied Remote Sensing, vol. 5, doi:10.1117/1.3539768, 2011

[10] A. Bazhyna, K. Egiazarian, N. Ponomarenko, V. Lukin, "Compression of noisy Bayer pattern color filter array images", Proceedings of the SPIE conference Computational Imaging V, SPIE Vol. 6498, doi:10.1117/12.715588, 2007.

[11] N. Ponomarenko, V. Lukin, M. Zriakhov, A. Kaarna, J. Astola, "Automatic approaches to on-land/on-board filtering and lossy compression of aviris images", Proceedings of IGARSS, 4 p., 2008.

[12] N. Ponomarenko, K. Egiazarian, V. Lukin, J. Astola, "Additional lossless compression of JPEG images", in Proceedings of 4th Symposium on Image and Signal Processing and Analysis, pp. 117-120, 2005.

[13] N. Ponomarenko, V. Lukin, K. Egiazarian, J. Astola, "Recursive coding: A new fast and simplem alternative of arithmetical coding", Telecommunications and Radio Engineering, 68 (20), pp. 1857-1863, 2009.

[14] N. Kozhemiakina, N. Ponomarenko, V. Lukin, K. Egiazarian, J. Astola, "Means and results of efficiency analysis for data compression methods applied to typical multimedia data", in Proceedings of International Scientific-Practical Conference Problems of Infocommunications Science and Technology, art. no. 6992281, pp. 12-14, 2014.

[15] N. Ponomarenko, V. Lukin, K. Egiazarian, E. Delp, "Comparison of lossy compression technique performance for real life color photo images", in Proceedings of Picture Coding Symposium, 4 p., 2009.