Lossless Compression of Medical and Natural High Bit Depth Sparse Histogram Images

Roman Starosolski(✉)

Institute of Informatics, Silesian University of Technology,
Akademicka 16, 44-100 Gliwice, Poland
roman.starosolski@polsl.pl

Abstract. In this paper we overview histogram packing methods and focus on an off-line packing method, which requires encoding the original histogram along with the compressed image. For a diverse set containing medical MR, CR and CT images as well as various natural 16-bit images, we report histogram packing effects obtained for several histogram encoding methods. The histogram packing improves significantly JPEG2000 and JPEG-LS lossless compression ratios of high bit depth sparse histogram images. In case of certain medical image modalities the improvement may exceed a factor of two, which indicates that histogram packing should be exploited in medical image databases as well as in medical picture archiving and communication systems in general as it is both highly advantageous and easy to apply.

Keywords: Image processing · Lossless image compression · High bit depth images · Medical images · Sparse histogram · Histogram packing · Histogram encoding · Image coding standards · JPEG2000 · JPEG-LS · DICOM

1 Introduction

Most single-frame single-band medical images, like MR, CR and CT and are of a high nominal bit depth, which usually varies from 12 to 16 bits per pixel. The number of active levels, i.e., intensity levels actually used by image pixels, may be smaller, than implied by the nominal bit depth, by an order of magnitude or even more. Furthermore, active levels are distributed throughout almost all the entire nominal intensity range, i.e., the images have sparse histograms of intensity levels. Also, the continuous tone natural (photographic) images of high bit depths may have sparse histograms. The image histogram is sparse, when the acquisition device quantizes analog image intensities to a number of levels, which is smaller than the nominally possible, and then distributes the quantized levels over a wider range. In case of multiple band (e.g., color or multispectral) images each band may use a different set of active levels. This makes viewing images easier–the brightest actual level gets closer to the brightest nominally possible. On the other hand, numeric values of originally consecutive quantization levels cease to be consecutive integers making compressing of the images less effective [4]. The acquisition device

S. Kozielski et al. (Eds.): BDAS 2015, CCIS 521, pp. 363–376, 2015.
DOI: 10.1007/978-3-319-18422-7_32

characteristic is not the only reason for image histogram sparseness. Some of the routine image processing methods, e.g., gamma correction or contrast adjustment, may make the histogram sparse. Histograms of some images are inherently sparse. Although this observation probably won't lead to improving the compression ratios for such images, we note the obvious fact: regardless of the nominal bit depth, the number of active levels cannot be greater, than the number of pixels. All in all, sparse histograms occur frequently in high bit depth images. The impact of histogram sparseness on image compression ratios is well known–applying to sparse histogram images, prior to regular compression, a histogram packing [23,12] leads to significant ratio improvement.

Image compression algorithms are based on sophisticated assumptions as to characteristics of images they process. Sparse histogram is clearly different from what is expected by most lossless image compression algorithms. An initial step of processing image data by these algorithms is aimed at making the data easier to compress. Predictive image compression algorithms (e.g., JPEG-LS [7]) use the predictor function to guess the pixel intensities and then the prediction errors, i.e., differences between actual and predicted pixel intensities, are encoded instead of pixel intensities. Even using extremely simple predictors, such as one that predicts that pixel's intensity is identical to its left-hand side neighbor, improves the resulting compression ratio. For typical single-band images, the pixel intensity distribution is roughly close to uniform. Prediction error distribution is close to Laplacian, i.e., symmetrically exponential. Therefore entropy of prediction errors is significantly smaller than entropy of pixel intensities and the resulting compression ratio is improved accordingly. However, since in sparse histogram images the pixel intensity distribution is not uniform the prediction may (and usually does) increase the entropy. Transformation image compression algorithms (e.g., JPEG2000 [5]) instead of pixel intensities encode a matrix of transformation coefficients (cosine or wavelet transformation), making probability distribution peaked and reducing the entropy of the data. In case of multiple-band color image compression, since bands are correlated, the first step is to apply a color space transformation that removes correlation. Most such transformations for color images [19] produce 3 bands: luminance band of entropy similar to original bands (provided that they were not sparse) and 2 chrominance bands of peaked distributions and reduced entropies. Again, if histograms of original bands are sparse, then the color space transformation increases entropy of all 3 bands. In the above mentioned cases, histogram packing should be applied before regular compression. Histogram packing does not reduce the entropy of the data (actually it may cause a small increase), but it allows entropy reduction in the first step of the regular algorithm (prediction, band transformation or color space transformation).

The simplest method of histogram packing is called the off-line histogram packing. This method simply maps all the active levels to the lowest part of the nominal intensity range (order-preserving one-to-one mapping). The off-line packing requires the information, describing how to expand the histogram after decompressing an image, to be encoded along with the compressed image–along with the compressed image we have to encode the original histogram. Below we briefly characterize other methods targeted at the sparse histogram image compression.

– The on-line preprocessing technique [14]–the histogram is built on-line, yet still prior to actual compression. The technique may be used as a preprocessing step for any image compression algorithm. The compression ratio improvement is reported to be about the same as for the off-line packing.
– The integrated on-line packing [12] is a variant of the on-line preprocessing technique integrated into the JPEG-LS algorithm. The compression ratio improvement is reported to be about the same as for the off-line packing.
– The extended prediction mode of the 2nd part of the JPEG-LS standard [8] is a technique designed for sparse histogram images. The compression ratio improvement reported for this technique [13] is several times smaller compared with the improvement obtained using the off-line packing.
– Embedded Image Domain Adaptive Compression (EIDAC) [24]–relatively complex, progressive compression algorithm targeted at the sparse histogram images.
– Piecewise Constant Image Model algorithm (PWC) [1] is designed for palette images having low number of colors (for images of up to 16 and up to 256 colors), for binary images, and for grayscale images of 8-bit depth. For low bit depth sparse histogram grayscale images it outperforms the EIDAC algorithm.

In the case of the off-line histogram packing, on-line preprocessing, integrated on-line packing, and EIDAC the histogram has to be encoded and transmitted explicitly as the side information to the decoder. In the 2nd part of the JPEG-LS, the extended prediction mode prohibits predicting intensity levels not found in the already processed part of the image. This way histogram is built on-line by the decoder based on the already decoded pixels. In the PWC algorithm, when the pixel intensity is different from intensities of it's neighbors, then the limited length LRU chain of active levels is used to guess the pixel intensity, and if this fails, then the intensity level is encoded in the predictive way–the difference between actual intensity and intensity predicted from the pixels neighborhood (using the standard predictor of JPEG-LS [7]) is encoded and transmitted to the decoder.

Standard algorithms, which along with the compressed image store the image palette (PNG [20]) or the level mapping table (JPEG-LS [7]), may be practically useful. For these algorithms, we use the off-line histogram packing to improve compression ratios. The decompression reconstructs image and its original histogram solely by means of the algorithm, i.e., no additional step of histogram expanding is required after decompressing images encoded using these algorithms. The support for histogram transformations is included in the 2nd part (annex K) of the JPEG2000 standard [6]. The standard describes two non-linear transformations, which may be applied to decoded pixel intensity levels: the piecewise linear function and the gamma-style function. These transformations are probably aimed mainly for lossy compression and for systems combining image transformations and compression rather than for lossless compression of images having sparse histograms. Fortunately, the level mapping table is a special case of the piece-wise linear function.

In the case of high bit depth sparse histogram images, the direct use of the above standards is not as straightforward as it seems–storing the 16-bit mapping table may be not supported by the algorithm itself, as in the case of PNG, or by the algorithm implementation. Popular implementations of JPEG2000 core coding system [5] may not support non-linear transformations, since these transformations are extensions defined in the 2nd part of the standard.

When the histogram of an image is not sparse globally, but the image contains sparse histogram areas, then the compression ratio may be improved by exploiting the local image characteristics [15]. Also, the histogram of an image containing uniform intensity areas or, after the initial prediction step, containing runs of equal prediction errors, may be considered locally (highly) sparse. Several modern algorithms use special mode of processing such data, usually improving this way both the compression ratio and the speed. In the JPEG-LS algorithm, instead of encoding each pixel separately, we encode, with a single codeword, the number of consecutive pixels of equal intensity. In the PWC, a more sophisticated method, the Skip-Innovation model [1], is used for encoding runs of equal intensity pixels. In the CALIC algorithm [22], we encode in a special way sequences of pixels that are of at most two levels.

Histogram packing techniques are also used in compression methods, which may be generally described as near lossless. In [10], for given grayscale image and for given tolerable error threshold, a minimum entropy sparse histogram is computed. Then image is transformed into sparse-histogram image, which is subject to histogram packing followed by lossless compression. For small error threshold, the scheme outperforms most of other tested near-lossless algorithms in terms of compression ratio. In [2], for biological micrographs, a model of noise being a function of signal in the imaging system is constructed using the measured acquisition device characteristics. Statistically insignificant intensity levels are discarded making the histogram of image sparse, then histogram packing and lossless compression is used. The scheme preserves image information content better than standard lossy algorithms; compared to lossless algorithms it obtains significantly better compression ratios.

First successful attempts to use histogram packing method in lossy image coding were recently reported. In [9] it was shown, that applying histogram packing and lossless compression to original image quantized using weighted median cut results in better reconstructed image quality (in terms of PSNR) than in case of JPEG2000; the method also allows for fine rate control for high bit rates.

Methods of image compression exploiting the histogram sparseness were found to be effective for low bit depth images. To our best knowledge, except for our previous work (see [16] and [18], of which this paper is a revised version) and the lossy method mentioned above [9], the compression of high bit depth images having sparse histograms has not been investigated. In this paper, we report effects of the off-line histogram packing in the case of high bit depth images. High bit depth images require the histogram to be encoded efficiently–simple off-line methods that are suitable for the 8-bit images, like encoding the histogram using the level mapping table, for 16-bit images may cause the data expansion. We analyze efficient methods of encoding this information.

The reminder of this paper is organized as follows. In section 2, we discuss methods of encoding histograms of high bit depth images. In section 3, for medical MR, CR and CT images as well as natural high bit depth grayscale images, we report effects of these methods and of applying the off-line histogram packing to JPEG2000, and JPEG-LS algorithms. Section 4 summarizes the research.

2 Histogram Encoding Methods

The off-line histogram packing method actually is an image transformation; we apply it to an image before the compression. It transforms sparse histogram image into the packed histogram image. The transformation is reversible if, along with the compressed image, we encode the original histogram. For the histogram expanding, it is enough to encode which of intensity levels are active–we do not need to know how many times the active level was used. The size of the encoded histogram, for some high bit depth images, is not negligible.

2.1 Mapping Table

For of 8-bit images, we may simply encode binary all the active levels. Following the JPEG-LS terminology, we call this method of histogram encoding the Mapping Table (MT). Actually, for the Mapping Table method, we have to store both the number of active levels and the levels, so for a histogram of an N-bit image containing L active levels we need $(L+1)N$ bits. This way, encoding the histogram of an 8-bit image requires not more than about a quarter of kilobyte. Typical size of an image after compression varies from a couple of dozens of kilobytes to several megabytes. So, even in the case of small images, the size of encoded histogram is a negligible factor in the overall compression ratio.

For 16-bit images, encoding the histogram using the Mapping Table method may lead to significant worsening of the compression ratio. In the worst case, when the histogram is not sparse, we would need 128 kilobytes to encode the histogram–the nominal number of image intensity levels is 216 and we need two bytes to encode binary the specific level. Therefore we need more efficient methods of encoding histograms of high bit depth images. Below we describe a couple of them.

2.2 Bit-Array

Instead of encoding the intensity level of each active level, we encode, for all nominally available levels, the information whether the specific level is active. Therefore, we need 2^N bits to encode the histogram of an N-bit image, regardless of the number of active levels. This method of histogram encoding was used for 8-bit images in the EIDAC algorithm starting from its first version [23]. We call a histogram encoded in this manner the Bit-Array of the histogram. For a 16-bit image, the Bit-Array requires 8 kilobytes, for an 8-bit image, 32 bytes only.

2.3 Run Length Encoding

Some images, like MR images used for experiments in this paper, use below 1% of all the nominally possible levels. A histogram of such image, encoded using the Bit-Array method, contains long runs of 0s separated by single 1s. Such histogram could be represented more compactly if we encoded lengths of runs of 0s. If, on the other hand, the histogram is not sparse, then it contains long runs (or just one long run) of 1s. Therefore we encode the Bit-Array of the histogram using the Run Length Encoding (RLE) variant described in the table 1 [16]. Note, that the RLE variant cannot be used if the last run of bits in the Bit-Array is not followed by single bit of value opposite to bits of run. In such a case we assume that a single bit being negation of the last bit in the Bit-Array follows the array–we encode this extra bit, but do not decode any bits after having decoded 2^N bits. Encoding the histogram using the RLE method is most efficient when the number of levels is close to 0 or close to 2^N. In the worst case, i.e., when every second level is used, we need 2^{N+2} bits for the RLE encoded histogram–32 kilobytes for the worst case histogram of a 16-bit image.

Table 1. Run Length Encoding of histograms of images of bit depths up to 16 bits

RLE codeword	Sequence
0 b_6 b_5 b_4 b_3 b_2 b_1 b_0	run of $r+1$ 0s followed by single 1, $r = b_6...b_0, r < 126$
0 1 1 1 1 1 1 0 $b_7...b_0$	run of $r+127$ 0s followed by single 1, $r = b_7...b_0$
0 1 1 1 1 1 1 1 $b_{15}...b_0$	run of $r+383$ 0s followed by single 1, $r = b_{15}...b_0$
1 b_6 b_5 b_4 b_3 b_2 b_1 b_0	run of $r+1$ 1s followed by single 0, $r = b_6...b_0, r < 126$
1 1 1 1 1 1 1 0 $b_7...b_0$	run of $r+127$ 1s followed by single 0, $r = b_7...b_0$
1 1 1 1 1 1 1 1 $b_{15}...b_0$	run of $r+383$ 1s followed by single 0, $r = b_{15}...b_0$

2.4 Further Compression of the Encoded Histogram

The Bit-Array is inefficient when the number of active levels is low; the RLE may be inefficient for certain numbers of intensity levels. Fortunately, both the Bit-Array of the histogram and the RLE encoded histogram may be further compressed. In the cases, when the above methods are most inefficient, the histograms encoded using them are likely to contain multiple repetitions of long sequences of symbols (bits or RLE codewords). For compressing such data we may use a universal compression algorithm capable of capturing long contexts, like the LZ77 universal dictionary compression algorithm [25].

3 Experimental Results and Discussion

We have compared experimentally the presented methods of histogram encoding. We also evaluated effects of packing histograms of high bit depth images on compression ratios of standard image compression algorithms. In experiments, we used MR, CR, and CT medical images as well as various 16-bit natural images, i.e., images acquired from scenes available for the human eye (photographic images).

In order to evaluate the impact of histogram sparseness on compression ratio for typical medical image of a certain modality, we used all the MR, CR, and CT medical images from a test image set described in another study [17]. There were 12 images of each of the modalities. Not all the medical images are of 16-bit depth and not every medical image has sparse histogram. Obviously, for the 10- or 12-bit images the method of histogram encoding gets less important for the overall compression ratio. Natural continuous tone grayscale images of 16-bit depth were included in experiments to evaluate effects of histogram packing on various non-medical images. These images included unprocessed images of various sizes as well as processed ones. Following groups of non-medical images were evaluated, each containing 4 images:

- Medium–natural (photographic) images of 16-bit depth classified in the above-mentioned set [17] as medium-sized;
- Contrast–Medium images with contrast increased by 25%;
- Gamma–Medium images with gamma (value 1.25) correction applied;
- Small–small images, which are reduced size (ninefold) Medium images.

The characteristics of images are reported in the table 2 (for brevity we report averaged results only). To characterize numerically image sparseness, we define the image level utilization $U = L/(1+l_{hi}-l_{lo})$, where l_{lo} and l_{hi} are respectively the lowest and the highest active level, and L is number of active levels. In the table, images are characterized by the image name, size (number of pixels), nominal depth (N), nominal (2^N) and actual (L) number of intensity levels, and by the level utilization (U). Results of encoding histograms, expressed as sizes (in bytes) of encoded histograms averaged for image groups, are reported in the table 3 for the following methods:

- MT–Mapping Table method;
- BA–Bit-Array method;
- BA+LZ77–Bit-Array of the histogram, compressed using LZ77;
- RLE–RLE method;
- RLE+LZ77–histogram encoded using RLE method followed by LZ77.

For the LZ77 compression we used the popular gzip compression utility, in the case of the RLE method it was applied directly to the encoded histogram. For the Bit-Array, since gzip is byte-wise, prior to compression, each bit was expanded to a byte.

Table 2. Characteristics of test images (averages for groups)

Images	Pixels	N	2^N	L	U
MR	196608	16.0	65536	1104	1.7%
CR	3527076	12.5	23296	7878	59.5%
CT	257569	14.7	45056	1951	17.3%
Medium	440746	16.0	65536	55839	87.1%
Contrast	440746	16.0	65536	23737	36.4%
Gamma	440746	16.0	65536	28076	44.4%
Small	48776	16.0	65536	25174	39.7%

Table 3. Encoded histogram size [B] (averages for groups)

Images	MT	BA	BA+LZ77	RLE	RLE+LZ77
MR	2210	8192	550	1127	102
CR	15184	2912	285	7071	179
CT	3592	5632	541	1852	219
Medium	111681	8192	4358	6231	3528
Contrast	47475	8192	1251	23737	909
Gamma	56154	8192	1546	28080	1314
Small	50350	8192	8447	13195	7288

The RLE+LZ77 method appears to be the most efficient. It obtains the shortest encoded histogram length for nearly all tested images. In the case of medical images, on average, it results in the encoded histogram length about 3 times shorter, than the second best BA+LZ77 method, for non-medical images the difference in favor of RLE+LZ77 is about 20%. Therefore, for evaluating effects of histogram packing on compression ratios of standard image compression algorithms, we use the RLE+LZ77 method.

The compression ratios obtained for images before histogram packing (Norm.), after packing (Pack.), and the ratio improvements due to histogram packing are reported in the table 4. The compression ratio is expressed in bits per pixel [bpp]: $8e/n$, where n is the number of pixels in the image, e–the size in bytes of the compressed image (including the size of the histogram encoded using the RLE+LZ77 method in the case of ratio after packing). We performed experiments for JPEG-LS [7,21] and JPEG2000 [5,3] standard image compression algorithms.

We notice, that effects of packing histograms on the compression ratios of tested algorithms are, for both algorithms, highly similar (see Fig. 1). Therefore we discuss results obtained for the more frequently used JPEG2000 algorithm only. As expected, the histogram packing does not improve compression ratios for Small images. For these images we observe noticeable worsening of compression ratios. Histograms of Small images are sparse ($U = 39.7\%$) and actually the packed histogram images would compress better by about 7% if we did not consider the encoded histogram size. For these images, the histogram sparseness is caused by the image size and also because of the image size the encoded

Table 4. Effects of histogram packing on compression ratios of JPEG-LS, and JPEG2000; results obtained for histograms encoded using RLE+LZ77 method (averages for groups)

Images	U	JPEG-LS			JPEG2000		
		Norm. [bpp]	Pack. [bpp]	Improvement	Norm. [bpp]	Pack. [bpp]	Improvement
MR	1.7%	10.009	4.944	50.6%	10.024	4.849	51.6%
CR	59.5%	6.343	5.398	14.9%	6.394	5.426	15.1%
CT	17.3%	7.838	4.557	41.9%	8.044	4.630	42.4%
Medium	87.1%	11.829	11.844	-0.1%	12.058	12.082	-0.2%
Contrast	36.4%	11.416	9.992	12.5%	11.951	10.558	11.7%
Gamma	44.4%	11.950	10.676	10.7%	12.183	10.965	10.0%
Small	39.7%	12.414	12.813	-3.2%	12.712	13.180	-3.7%

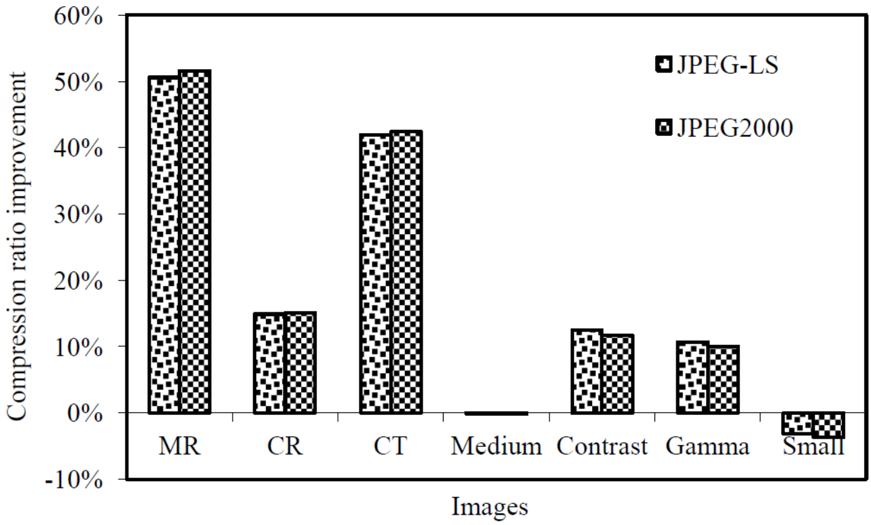

Fig. 1. Average compression ratio improvement due to histogram packing (RLE+LZ77)

histogram size is an important factor in the overall compression ratio–if compression ratios of such images could be improved by histogram packing, then we could try to improve the compression ratio for any image by splitting it into several smaller ones and compressing them separately.

Histograms of most Medium images are non-sparse. Only in the case of one of these images the histogram may be considered sparse, since for this image $U = 70.6\%$. The impact of histogram packing on compression ratios is similar for all Medium images–it negligibly worsens the compression ratios. Based on the image level utilization we'd rather expect compression ratio improvement for the above mentioned image. Analyzing histogram of this image we found, that almost all image pixels are of low intensities and the histogram is sparse

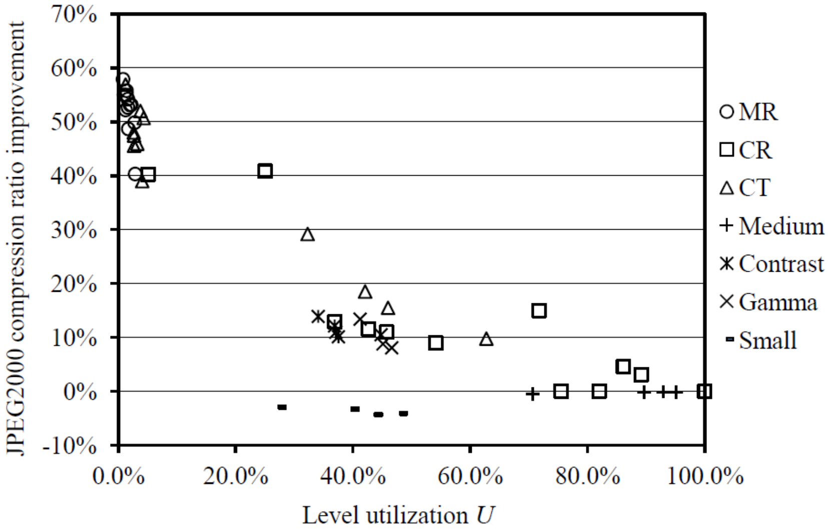

Fig. 2. JPEG2000 improvements for individual images due to histogram packing (RLE+LZ77)

only in the high intensity range. Similar image characteristics and similar lack of impact of histogram packing on compression ratio may be observed for two CR medical images. These observations show, that image level utilization is not a perfect histogram sparseness measure. Except for the cases described above, the histogram packing improves compression ratios for high bit depth sparse histogram images. The improvement varies depending on the image level utilization U, which we use as a measure of the histogram sparseness (see Fig. 2). For $U < 1/4$ the compression ratio improvement is roughly 50%, i.e., the size of the compressed image gets halved by applying the histogram packing method. For $U \approx 1/2$ we get the compression ratio improvement of about 10–20%; this level of improvement is not negligible for lossless image compression algorithm–the difference in compression ratio between algorithms obtaining best ratios and algorithms obtaining best speeds usually does not exceed 10% for the images used [17]. For $U > 3/4$ the histogram packing improves ratios for some images only, however, it does not worsen noticeably ratios for the remaining ones.

The RLE+LZ77 histogram encoding method outperforms others. It is interesting, however, whether it is practically justified to use it instead of some simpler one when we consider the overall image compression ratio, not the encoded histogram size alone. In the table 5, we report the JPEG2000 compression ratios for packed histogram images (calculated assuming, that the histogram is encoded using 0 bytes) and the cost of encoding histogram for the described histogram encoding methods. The histogram encoding cost is expressed in bits per image pixel and as relative to the above compression ratio. For brevity we report averaged results only.

Table 5. The cost of histogram encoding per image pixel [bpp] and relative to JPEG2000 compression ratio of packed histogram images. JPEG2000 ratio calculated excluding the encoded histogram size.

Images	JPEG2000	MT		BA	
	ratio [bpp]	[bpp]	relative	[bpp]	relative
MR	4.938	0.141	2.8%	0.500	10.1%
CR	5.398	0.044	0.8%	0.017	0.3%
CT	4.550	0.121	2.7%	0.168	3.7%
Medium	12.018	2.027	16.9%	0.149	1.2%
Contrast	10.542	0.862	8.2%	0.149	1.4%
Gamma	10.941	1.019	9.3%	0.149	1.4%
Small	11.618	8.258	71.1%	1.344	11.6%

Images	BA+LZ77		RLE		RLE+LZ77	
	[bpp]	relative	[bpp]	relative	[bpp]	relative
MR	0.032	0.6%	0.072	1.5%	0.006	0.1%
CR	0.001	0.0%	0.021	0.4%	0.001	0.0%
CT	0.017	0.4%	0.061	1.3%	0.007	0.2%
Medium	0.079	0.7%	0.113	0.9%	0.064	0.5%
Contrast	0.023	0.2%	0.431	4.1%	0.017	0.2%
Gamma	0.028	0.3%	0.510	4.7%	0.024	0.2%
Small	1.385	11.9%	2.163	18.6%	1.195	10.3%

In the case of some images, the effects of histogram packing vary significantly depending on the method of encoding the histogram. The best method for all image groups is the RLE+LZ77. The BA+LZ77 method also obtains good ratios, however in the case of certain modalities, namely MR and CT, the cost of encoding histograms using this method is several times greater than the cost of RLE+LZ77. Compared to the RLE+LZ77 method, the simplest methods, which were successfully used for low bit depth images (MT and BA) are, for some modalities, highly inefficient.

The RLE code is constructed ad-hoc; the algorithm for encoding the RLE sequence was selected without thorough analysis of the RLE sequence structure. Also, we did not analyze other algorithms that could be used instead of LZ77. Experiments were done using gzip, a popular general purpose compression utility not adjusted to characteristics of data produced by our RLE variant–it expands the shortest RLE sequences. Therefore there is certainly a possibility to encode the histogram more efficiently, than using our RLE+LZ77 method. From practical point of view, however, there is no need to encode the histogram more efficiently. Except for the Medium and Small images, which do not benefit from histogram packing, by finding a better method we could get further ratio improvement of no more than about 0.2% only.

For medical images, the cost of encoding histograms using the MT method is a small factor in the ratio improvement obtained due to histogram packing. Decoding of images with packed histograms encoded using the MT method is already supported by the JPEG-LS [7] standard and by the 2nd part (annex K) of the JPEG2000 standard [6]. The JPEG-LS standard is included in the

DICOM standard [11] commonly used in medical picture archiving and communication systems. Therefore the MT method may in practice be very useful for medical devices acquiring MR, CT, and CR images. Provided that the 16-bit JPEG-LS mapping table is supported by the decompression software, using the off-line histogram packing and the MT method of histogram encoding we may significantly improve compression ratios while maintaining compatibility with current standards. We note that the annex K of the 2nd part of the JPEG2000 standard is not included in the the DICOM standard, as opposed to JPEG2000 core coding system.

4 Conclusions

In this paper we overview histogram packing methods and report effects of packing histograms of high bit depth images on compression ratios obtained by lossless image compression algorithms. Experiments were performed for a diverse set of test images, including medical MR, CR, and CT images as well as unprocessed and processed (gamma and contrast adjustment) natural 16-bit images. We focused on the off-line packing method. The off-line packing requires the information, describing how to expand the histogram after decompressing an image, to be encoded along with the compressed image–along with the compressed image we have to encode the original histogram. The size of the encoded histogram, for some high bit depth images, is not negligible. One of the histogram encoding methods (RLE+LZ77) obtains the shortest encoded histogram length for nearly all tested images and in practice is sufficiently good for encoding histograms of wide range of images. A simpler method (MT) may be useful for medical images. For these images, its use results in improvements of the compression ratio little worse compared to RLE+LZ77, but decoding of images with packed histograms encoded using the MT method is already supported by JPEG-LS (included in DICOM) and JPEG2000 (part 2) standards. The effects of packing histograms on the compression ratios of JPEG2000 and JPEG-LS are, for both tested algorithms, very similar–histogram packing improves significantly lossless compression ratios for high bit depth sparse histogram images. The ratio improvement due to histogram packing may approach or exceed a factor of two, as in case of CT and MR medical images, respectively. Though effects of histogram packing are known for over a dozen years, the technique is not considered a routine step of reversible image compression algorithms. The results presented in this paper indicate, that at least in case of MR and CT modalities, histogram packing should be exploited in medical image databases as well as in medical picture archiving and communication systems in general as it is both highly advantageous and easy to apply.

Acknowledgments. This work was supported by BK-266/RAu2/2014 grant from the Institute of Informatics, Silesian University of Technology and by POIG.02.03.01-24-099/13 grant: GeCONiI–Upper Silesian Center for Computational Science and Engineering.

References

1. Ausbeck Jr., P.J.: The piecewice-constant image model. Proc. of the IEEE 88(11), 1779–1789 (2000)
2. Bernas, T., Starosolski, R., Robinson, J.P., Rajwa, B.: Application of detector precision characteristics and histogram packing for compression of biological fluorescence micrographs. Computer Methods and Programs in Biomedicine 108(2), 511–523 (2012), http://dx.doi.org/10.1016/j.cmpb.2011.03.012
3. Christopoulos, C., Skodras, A., Ebrahimi, T.: The JPEG2000 still image coding system an overview. IEEE Trans. on Consumer Electronics 46(4), 1103–1127 (2000)
4. Ferreira, P.J., Pinho, A.J.: Why does histogram packing improve lossless compression rates? IEEE Signal Processing Letters 9(8), 259–261 (2002)
5. INCITS/ISO/IEC, ITU-T: Information technology – JPEG 2000 image coding system: Core coding system, INCITS/ISO/IEC International Standard 15444-1 and ITU-T Recommendation T.800 (2004)
6. INCITS/ISO/IEC, ITU-T: Information technology – JPEG 2000 image coding system: Extensions, INCITS/ISO/IEC International Standard 15444-2 and ITU-T Recommendation T.801 (2008)
7. INCITS/ISO/IEC, ITU-T: Information technology – Lossless and near-lossless compression of continuous-tone still images – Baseline, INCITS/ISO/IEC International Standard 14495-1 and ITU-T Recommendation T.87 (2006)
8. INCITS/ISO/IEC, ITU-T: Information technology – Lossless and near-lossless compression of continuous-tone still images: Extensions, INCITS/ISO/IEC International Standard 14495-2 and ITU-T Recommendation T.870 (2003)
9. Iwahashi, M., Kobayashi, H., Kiya, H.: Lossy compression of sparse histogram image. In: Proc. ICASSP 2012, pp. 1361–1364 (2012), http://dx.doi.org/10.1109/ICASSP.2012.6288143
10. Nasr-Esfahani, E., Samavi, S., Karimi, N., Shiran, S.: Near lossless image compression by local packing of histogram. In: Proc. ICASSP 2008, pp. 1197–1200 (2008), http://dx.doi.org/10.1109/ICASSP.2008.4517830
11. NEMA: Digital imaging and communications in medicine (DICOM) part 5: Data structures and encoding, National Electrical Manufacturers Association Standard DICOM PS3.5 2014c (2014)
12. Pinho, A.J.: On the impact of histogram sparseness on some lossless image compression techniques. In: Proc. ICIP 2001, vol. II, pp. 442–445 (2001)
13. Pinho, A.J.: A comparison of methods for improving the lossless compression of images with sparse histograms. In: Proc. ICIP 2002, vol. 2, pp. 673–676 (2002)
14. Pinho, A.J.: An online preprocessing technique for improving the lossless compression of images with sparse histograms. IEEE Signal Processing Letters 9(1), 5–7 (2002)
15. Pinho, A.J.: Preprocessing techniques for improving the lossless compression of images with quasi-sparse and locally sparse histograms. In: Proc. ICME 2002, vol. 1, pp. 633–636 (2002)
16. Starosolski, R.: Compressing images of sparse histograms. In: Kowalczyk, A., Fercher, A.F., Tuchin, V.V. (eds.) Proc. SPIE: Medical Imaging, vol. 5959, pp. 209–217 CID:595912. The International Society for Optical Engineering, Bellingham (2005), http://dx.doi.org/10.1117/12.624489

17. Starosolski, R.: Performance evaluation of lossless medical and natural continuous tone image compression algorithms. In: Kowalczyk, A., Fercher, A.F., Tuchin, V.V. (eds.) Proc. SPIE: Medical Imaging, vol. 5959, pp. 116–127 CID:59590L. The International Society for Optical Engineering, Bellingham (2005), http://dx.doi.org/10.1117/12.624136
18. Starosolski, R.: Compressing high bit depth images of sparse histograms. In: Simos, T.E., Psihoyios, G. (eds.) International Electronic Conference on Computer Science, AIP Conference Proceedings, vol. 1060, pp. 269–272. American Institute of Physics, USA (2008), http://dx.doi.org/10.1063/1.3037069
19. Starosolski, R.: New simple and efficient color space transformations for lossless image compression. Journal of Visual Communication and Image Representation 25(5), 1056–1063 (2014), http://dx.doi.org/10.1016/j.jvcir.2014.03.003
20. W3C, ISO/IEC: Information technology – Computer graphics and image processing – Portable Network Graphics (PNG): Functional specification, World Wide Web Consortium Recommendation and ISO/IEC International Standard 15948 (2004)
21. Weinberger, M.J., Seroussi, G., Sapiro, G.: The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS. IEEE Trans. Image Proc. 9(8), 1309–1324 (2000)
22. Wu, X., Memon, N.: Context-based, adaptive, lossless image codec. IEEE Trans. on Communications 45(4), 437–444 (1997)
23. Yoo, Y., Kwon, Y.G., Ortega, A.: Embedded image-domain compression of simple images. In: Proc. of the 32nd Asilomar Conf. on Signals, Systems, and Computers, vol. 2, pp. 1256–1260 (1998)
24. Yoo, Y., Kwon, Y.G., Ortega, A.: Embedded image domain compression using context models. In: Proc. ICIP 1999, pp. 477–481 (1999)
25. Ziv, J., Lempel, A.: A universal algorithm for sequential data compression. IEEE Trans. on Information Theory 32(3), 337–343 (1997)