IMPROVEMENT OF THE LOSSLESS COMPRESSION OF IMAGES WITH QUASI-SPARSE HISTOGRAMS

Armando J. Pinho[*] *António J. R. Neves*
Dept. of Electronics and Telecommunications / IEETA
University of Aveiro, 3810–193 Aveiro, Portugal

ABSTRACT

Off-line histogram packing is a known method which is capable of producing improvements if applied prior to the lossless compression of images having sparse histograms. However, this technique becomes useless if the image possesses a quasi-sparse histogram, even if it differs from the strictly sparse case only by a minimal margin. In this paper, we present a technique that is able to overcome this drawback and we present results showing its effectiveness.

1 INTRODUCTION

Off-line histogram packing is a known method capable of producing improvements if applied prior to the lossless compression of images having sparse histograms. Basically, off-line histogram packing is produced through the construction of an one-to-one order-preserving mapping from the image intensity values to a contiguous subset of the integers. For constructing this mapping we need to know, *a priori*, which intensity values are present in the image. If that is unknown, which is the most often case, we need to perform a pass through the image in order to find out those intensity values. In the latter case, the complete encoding process, i.e., histogram packing and lossless image coding, cannot be performed on-line. On-line strategies have been recently proposed, some of them designed specifically for a particular encoding algorithm [1, 2, 3], some others designed to act as a pre-processing stage detached from the particular encoding algorithm that is used [4].

Generally, the off-line histogram packing method is very effective in images characterized by having sparse histograms. This can be verified using the examples given in Tables 1 and 2, where compression ratios obtained before off-line histogram packing ("Normal" column) and after off-line histogram packing ("Off-line packing" column) are displayed. As can be observed, the compression improvement after off-line histogram packing is very significant for images having sparse histograms, i.e., the images in the first group and some of the images in the third group. For images not belonging to this class, such as those in the second group, a slight decrease in compression ratio may be verified, which is due to the small overhead introduced by the mapping table. This table is required for later recovery of the original intensity values. The number of different intensity values that was found in each image is also shown in the "Intensities" column. However, it is worth noting that this number should be used only as an indication of potential histogram sparseness. In fact, an image using only a small, but contiguous, sub-set of the available intensities cannot be further improved by this method, because its histogram is already packed.

2 QUASI-SPARSE HISTOGRAMS

One of the major drawbacks of off-line histogram packing is that if even most of the intensity values appear only once or just a few number of times in the image, they will be considered by the off-line histogram packing technique as having equal importance as those that occur most frequently. In other words, images having quasi-sparse histograms cannot benefit from this method.

Figure 2 shows a 8 bits per pixel, 2,048×2,048 image ("aerial2") which uses all the 256 intensity values. Therefore, as can be seen in Tables 1 and 2, off-line histogram packing does not produce any compression improvement in this image. However, the analysis of the histogram of this image reveals, for example, that 60 of the intensity values occur less than 164 times, i.e., less than 1% of the mean number of occurrences, which is 16,384.

In this paper we present a novel technique that aims overcoming the lack of appropriateness of off-line histogram packing regarding the handling of images with quasi-sparse histograms. The proposed method is also off-line, but works using a reduced set of symbols. We will refer to it as the method of "packing with a reduced symbol-set".

3 PACKING WITH A REDUCED SYMBOL-SET

Let us denote by $\mathcal{I}$ the set of all different intensity values used by a given image, and by $S \in \mathbb{N}$ some pre-defined value. During the processing of sample x^t,[1] which generates a transformed sample y^t at time instant t, we assume

---

[*] This work was supported in part by the Fundação para a Ciência e a Tecnologia (FCT). Corresponding author: Armando J. Pinho, E-mail: `ap@det.ua.pt`, URL: `www.ieeta.pt/~ap`.

[1] We assume that the image samples have been transformed into a sample sequence, x^t, using some image scanning strategy. In this work we used a raster-scanning approach.

Figure 1: Image with a quasi-sparse histogram: "aerial2".

that a previously constructed subset of $\mathcal{I}$, $\mathcal{I}_n^t$, is available:

$$\mathcal{I}_n^t = \{I_0, I_1, \ldots, I_{n-1}\}, \quad n \leq S.$$

Moreover, we assume, without loss of generality, that $I_i < I_j, \forall_{i<j}$, and that the following one-to-one, order-preserving mapping in $\mathbb{N}_0$ is also available:

$$h^t = (I_0 \mapsto 0, I_1 \mapsto 1, \ldots, I_{n-1} \mapsto n-1).$$

Sample x^t is processed as follows. If $x^t \in \mathcal{I}_n^t$, then $y^t = h^t(x^t)$. However, if $x^t \notin \mathcal{I}_n^t$, then $y^t = n$, which is the first element of $\mathbb{N}_0$ that does not belong to $h^t(\mathcal{I}_n^t)$. In this case, the intensity value $\tilde{I} = x^t$ is stored into a file, which is used for later recovery of the original image intensity values. We call this file the "recovery file".

The occurrence of an intensity not belonging to $\mathcal{I}_n^t$ also implies the rearrangement of the mapping, which depends on whether $n = S$ or not. If $n < S$, and assuming that $I_i < \tilde{I} < I_{i+1}$, then the new mapping is:

$$h^{t+1} = (\ldots, I_i \mapsto i, \tilde{I} \mapsto i{+}1, I_{i+1} \mapsto i{+}2, \ldots, I_{n-1} \mapsto n).$$

As can be seen, $\tilde{I}$ is inserted in the mapping in such a way that the one-to-one, order-preserving property is maintained. On the other hand, if $n = S$, in addition to the inclusion of $\tilde{I}$ in the mapping, as described above, it is also required the deletion of one of the members of $\mathcal{I}_n^t$ (i.e., the cardinality of the set is kept equal to S). Also in this case, the mapping has to be rearranged in order to obey to the one-to-one, order-preserving property.

Decoding is performed using a similar strategy as encoding. When decoding sample y^t, if $y^t < S$, then $x^t = (h^t)^{-1}(y^t)$. [2] Otherwise, an intensity value, $\tilde{I}$, is fetched from the recovery file and $x^t = \tilde{I}$. The mapping is always reorganized following the same procedures as those performed by the encoder.

Until now we left some important issues open, namely, (1) how the intensity values are stored in the recovery file; (2) how do we choose which intensity value is deleted from the mapping when a new value has to be inserted and $\mathcal{I}_n^t$ is not allowed to increase further (i.e., $n = S$); (3) how do we find the optimum value of S.

Concerning the first question, i.e., how the values are stored in the recovery file, for simplicity we store these values directly, without any kind of compression. This means that, for 8 bits per pixel images, each intensity value is stored in one byte. Some approaches using popular entropy coding methods have been tried, although they are not addressed here. Nevertheless, all compression results presented in this paper include the overhead introduced by the size of the recovery file.

The second question, i.e., how intensity values are substituted in $\mathcal{I}_n^t$ when $n = S$, lead us to several approaches. Among them, we just point out two of the most obvious and simple: one of them calls for the removal of the oldest unused intensity; the other relies on the removal of the fewest used intensity. Both methods have been tested, although only results using the former one have been included here, since it provided globally better results on the images used for testing.

Finally, the question regarding finding the optimum value of S. This image-dependent and also encoding-method-dependent parameter plays a crucial role in the preprocessing technique that we describe in this paper. In fact, depending on the value of S, we may obtain substantial compression improvements or, if incorrectly chosen, we may end up with a degradation in the compression ratio. Examples of some curves representing the compressed size of the images as a function of S can be observed in Fig. 2. We are currently investigating efficient ways for determining or estimating the optimum value of S. The results presented in this paper rely on values of S found by exhaustive search, which, although appropriate for demonstrating the effectiveness of the algorithm, may not be desirable for some practical applications. Therefore, this is still a problem that needs further study.

4 EXPERIMENTAL RESULTS

Tables 1 and 2 present compression results using the two most recent ISO/IEC standards and ITU recommendations for compressing continuous-tone images: JPEG-LS [5, 6] and JPEG-2000 [7, 8]. For JPEG-LS encoding, we used the implementation provided by the Signal Processing & Multimedia Group at the University of British Columbia (SPMG / JPEG-LS V.2.2 codec, `ftp://spmg.ece.ubc.ca/pub/jpeg-ls/ver-2.2/`). For JPEG-2000 encoding, we used a codec implemented by the JJ2000 group (JPEG-2000 V.3.2.2 codec, `http://jj2000.epfl.ch/`).

To assess the efficiency of the proposed technique we used three sets of images. Two of them[3] were those used by Yoo *et al.* to test the efficiency of EIDAC, an embedded image-domain adaptive compression technique, specialized in the

---

[2] Notation $(h^t)^{-1}$ indicates a mapping that is the inverse of h^t.

[3] From `http://sipi.usc.edu/~younggap/EIDAC`.

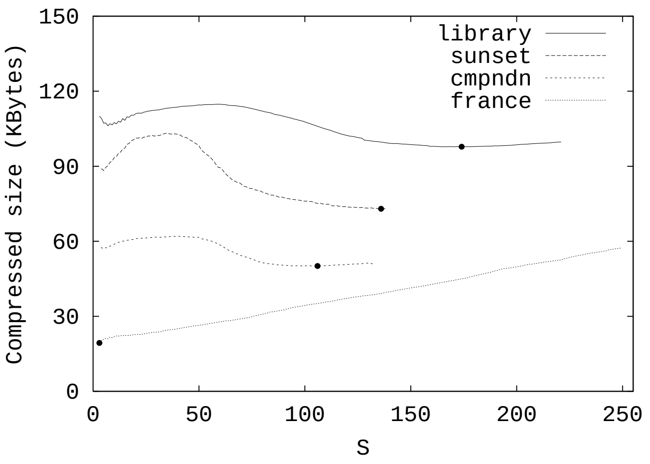

Figure 2: Examples of the dependence of the compressed size of the images with parameter S. These values have been obtained with the JPEG-LS encoder. The dots mark the optimum values.

compression of “simple images” [4] [9]. The first set (corresponding to the first group of images in Tables 1 and 2) is a gray-scale-converted version of a set used by Ausbeck in its PWC coder [10]. [5]

The second set (second group in Tables 1 and 2) comprises several natural images and has the objective of testing the robustness of the method in images that are not “simple” (this set was also used in [9]). The third set (last group of images in Tables 1 and 2) is composed of five images taken from the BragZone archive[6]. This set was recently used to illustrate the poor performance of JPEG-LS and CALIC (context-based, adaptive, lossless image codec) [11] in compressing this type of images [12].

5 CONCLUSIONS

From the observation of the “Total” rows in Tables 1 and 2, we can immediately conclude that the proposed method (off-line packing with a reduced symbol-set) provides globally better results than the normal off-line packing. Looking at individual images, we can notice some dramatic improvements. That is the case of image “france”, which gets a compression improvement of 75% if using the JPEG-2000 codec and 66% if using JPEG-LS. The compression of image “sea_dusk” improves 72% (using JPEG-2000) and 58% (using JPEG-LS). Also, a considerable 19% improvement, attainable by both compression standards, can be noticed in the compression of image “aerial2”. This percentages represent improvements over the normal off-line packing method. Other less dramatic but also important improvements can be observed in other images.

In this paper we proposed a pre-processing technique which is capable of producing compression improvements on images that have histograms that, although not strictly sparse, are quasi-sparse. In this case, the known off-line packing approach is unsuitable. However, by reducing the size of the symbol-set used by the packing procedure, we are able to globally attain better results, being some of them, individually, quite dramatic.

References

[1] R. Ansari and N. Memon. The JPEG lossless compression standards. In A. Bovik, ed., *Handbook of image and video processing*, Academic Press, 2000.

[2] *Information technology - Lossless and near-lossless compression of continuous-tone still images: extensions*. ISO/IEC 14495–2, 2000.

[3] A. J. Pinho. On the impact of histogram sparseness on some lossless image compression techniques. In *Proc. of the 8th IEEE Int. Conf. on Image Processing, ICIP-2001*, pages 442–445, Thessaloniki, Greece, October 2001.

[4] A. J. Pinho. An on-line pre-processing technique for improving the lossless compression of images with sparse histograms. *IEEE Signal Processing Letters*, 2002 (in press).

[5] *Information technology - Lossless and near-lossless compression of continuous-tone still images*. ISO/IEC 14495–1 and ITU Recommendation T.87, 1999.

[6] M. J. Weinberger, G. Seroussi, and G. Sapiro. The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS. *IEEE Trans. on Image Processing*, 9(8):1309–1324, August 2000.

[7] *Information technology - JPEG 2000 image coding system*. ISO/IEC International Standard 15444–1, ITU-T Recommendation T.800, 2000.

[8] M. W. Marcellin, M. J. Gormish, A. Bilgin, and M. P. Boliek. An overview of JPEG-2000. In *Proc. of the Data Compression Conf., DCC-2000*, pages 523–541, Snowbird, Utah, March 2000.

[9] Y. Yoo, Y. G. Kwon, and A. Ortega. Embedded image-domain compression using context models. In *Proc. of the 6th IEEE Int. Conf. on Image Processing, ICIP-99*, pages 477–481, Kobe, Japan, October 1999.

[10] P. J. Ausbeck Jr. The piecewice-constant image model. *Proceedings of the IEEE*, 88(11):1779–1789, November 2000.

[11] X. Wu and N. Memon. Context-based, adaptive, lossless image coding. *IEEE Trans. on Communications*, 45(4):437–444, April 1997.

[12] B. Carpentieri, M. J. Weinberger, and G. Seroussi. Lossless compression of continuous-tone images. *Proceedings of the IEEE*, 88(11):1797–1809, November 2000.

---

[4] The expression “simple images” is used by some authors to designate images with sparse histograms.

[5] The piecewise-constant image model (PWC) coder has been designed for compressing palette images, although it can also be used for compressing “simple” (gray-level) images. We considered only images with at least four different intensities, due to a restriction in the JPEG-LS coder.

[6] From `http://links.uwaterloo.ca/BragZone`.

Image	Intensities	Normal		Off-line packing			Packing with a reduced symbol-set			
		Size	bps	Size	bps	%	Size	bps	%	S
benjerry	48	14,076	4.027	9,664	2.765	31.3	6,718	1.922	52.3	4
books	7	43,859	6.164	15,318	2.152	65.1	12,570	1.766	71.3	5
cmpndd	133	114,362	2.326	98,767	2.009	13.6	77,298	1.572	32.4	129
cmpndn	132	107,596	2.189	92,594	1.883	13.9	66,069	1.344	38.6	3
gate	69	32,916	4.323	24,316	3.193	26.1	24,147	3.171	26.6	64
music	8	8,457	5.491	3,180	2.064	62.4	2,944	1.911	65.2	3
netscape	27	30,769	4.022	17,887	2.338	41.9	17,586	2.298	42.8	26
sea_dusk	43	8,214	0.417	5,894	0.299	28.2	1,667	0.084	79.7	3
sunset	138	119,031	3.099	106,013	2.760	10.9	103,896	2.705	12.7	125
winaw	10	84,913	2.307	33,757	0.917	60.2	25,059	0.681	70.5	5
yahoo	156	13,782	4.062	13,001	3.832	5.7	8,610	2.537	37.5	3
Total	—	**577,975**	—	**420,391**	—	**27.3**	**346,564**	—	**40.0**	—
aerial2	256	2,852,576	5.440	2,852,834	5.441	0.0	2,315,638	4.416	18.8	119
bike	220	2,967,668	4.528	2,722,648	4.154	8.3	2,726,023	4.159	8.1	220
bikeh	220	159,116	6.078	150,191	5.737	5.6	150,798	5.760	5.2	220
cafe	220	3,507,488	5.352	3,410,314	5.203	2.8	3,410,937	5.204	2.8	220
goldhill	220	158,411	4.834	158,631	4.841	-0.1	159,801	4.876	-0.9	218
lena	215	141,348	4.313	141,566	4.320	-0.2	142,431	4.346	-0.8	212
woman	220	223,463	5.455	215,073	5.250	3.8	215,423	5.259	3.6	220
Total	—	**10,010,070**	—	**9,651,257**	—	**3.6**	**9,121,051**	—	**8.9**	—
france	249	84,051	2.017	83,999	2.016	0.1	20,859	0.500	75.2	3
frog	102	241,791	6.254	204,789	5.297	15.3	180,905	4.679	25.2	55
library	221	116,215	5.692	114,259	5.596	1.7	112,430	5.506	3.3	193
mountain	110	257,216	6.698	208,477	5.429	18.9	205,696	5.356	20.0	97
washsat	35	145,105	4.428	73,201	2.233	49.6	73,269	2.235	49.5	35
Total	—	**844,378**	—	**684,725**	—	**18.9**	**593,159**	—	**29.8**	—

Table 1: Comparison of the compression results (in bits/sample) obtained with JPEG-2000 applied directly to the images ("Normal"), with JPEG-2000 applied after off-line packing ("Off-line packing") and with JPEG-2000 applied after using the method proposed in this paper ("Packing with a reduced symbol-set"). Percentages represent compression gains in relation to the "Normal" values.

Image	Intensities	Normal		Off-line packing			Packing with a reduced symbol-set			
		Size	bps	Size	bps	%	Size	bps	%	S
benjerry	48	6,707	1.919	4,881	1.396	27.2	4,784	1.368	28.7	26
books	7	39,859	5.601	13,396	1.882	66.4	11,657	1.638	70.8	5
cmpndd	133	71,469	1.454	62,431	1.270	12.6	61,733	1.255	13.6	108
cmpndn	132	58,639	1.193	51,619	1.050	12.0	51,298	1.043	12.5	106
gate	69	27,656	3.632	20,718	2.721	25.1	21,692	2.848	21.6	64
music	8	4,534	2.943	1,747	1.134	61.5	1,894	1.229	58.2	8
netscape	27	21,249	2.777	13,191	1.724	37.9	12,957	1.693	39.0	26
sea_dusk	43	4,061	0.206	3,479	0.176	14.3	1,446	0.073	64.4	3
sunset	138	83,552	2.175	75,412	1.963	9.7	74,761	1.946	10.5	136
winaw	10	48,189	1.309	20,102	0.546	58.3	19,286	0.524	60.0	7
yahoo	156	8,822	2.600	8,401	2.476	4.8	7,382	2.175	16.3	3
Total	—	**374,737**	—	**275,377**	—	**26.5**	**268,890**	—	**28.2**	—
aerial2	256	2,772,925	5.288	2,773,183	5.289	0.0	2,250,493	4.292	18.8	117
bike	220	2,854,695	4.355	2,600,571	3.968	8.9	2,604,249	3.973	8.8	220
bikeh	220	144,412	5.516	134,535	5.139	6.8	136,167	5.201	5.7	220
cafe	220	3,336,249	5.090	3,238,178	4.941	2.9	3,238,527	4.941	2.9	220
goldhill	220	154,391	4.711	154,637	4.719	-0.2	155,609	4.748	-0.8	204
lena	215	139,106	4.245	139,340	4.252	-0.2	140,146	4.276	-0.7	209
woman	220	220,671	5.387	212,511	5.188	3.7	212,998	5.200	3.5	220
Total	—	**9,622,449**	—	**9,252,955**	—	**3.8**	**8,738,189**	—	**9.2**	—
france	249	58,792	1.411	58,602	1.406	0.3	19,812	0.475	66.3	3
frog	102	233,831	6.048	200,089	5.175	14.4	176,525	4.566	24.5	11
library	221	104,140	5.100	102,217	5.006	1.8	100,139	4.904	3.8	174
mountain	110	246,604	6.421	201,417	5.245	18.3	198,728	5.175	19.4	95
washsat	35	135,309	4.129	65,734	2.006	51.4	65,767	2.007	51.4	35
Total	—	**778,676**	—	**628,059**	—	**19.3**	**560,971**	—	**28.0**	—

Table 2: Comparison of the compression results (in bits/sample) obtained with JPEG-LS applied directly to the images ("Normal"), with JPEG-LS applied after off-line packing ("Off-line packing") and with JPEG-LS applied after using the method proposed in this paper ("Packing with a reduced symbol-set"). Percentages represent compression gains in relation to the "Normal" values.