Statistical Modeling of Huffman Tables Coding

S. Battiato[1], C. Bosco[1], A. Bruna[2], G. Di Blasi[1], and G. Gallo[1]

[1] D.M.I. – University of Catania - Viale A. Doria 6,
95125, Catania, Italy
{battiato, bosco, gdiblasi, gallo}@dmi.unict.it
[2] STMicroelectronics - AST Catania Lab
arcangelo.bruna@st.com

Abstract. An innovative algorithm for automatic generation of Huffman coding tables for semantic classes of digital images is presented. Collecting statistics over a large dataset of corresponding images, we generated Huffman tables for three images classes: landscape, portrait and document. Comparisons between the new tables and the JPEG standard coding tables, using also different quality settings, have shown the effectiveness of the proposed strategy in terms of final bit size (e.g. compression ratio).

1 Introduction

The wide diffusion of imaging consumer devices (Digital Still Cameras, Imaging Phones, etc.) coupled with the increased overall performances capabilities, makes necessary an increasing reduction in the bitstream size of compressed digital images. The final step of the JPEG baseline algorithm ([5]) is a lossless compression obtained by an entropy encoding. The normalized DCT coefficients, properly grouped according to some simple heuristics techniques, are encoded by classical Huffman coding ([3],[11]) making use of a set of coding tables. Usually, the standard coding tables are used, as suggested in ([5], [8]) and included in the final bit-stream for every image to be compressed. This approach presents two main disadvantages:

1. the JPEG Huffman encoder writes all codes of the corresponding tables in the final bitstream even if only some of them were used to encode the associated events of the particular input image. An overhead (mainly for high compression rate) is clearly present because the header of the JPEG file will contain complete tables where unused codes are stored.
2. besides the JPEG standard encoder doesn't make use of any statistic about the distribution of the events of the current image.

To overcome these problems the JPEG encoder can be modified so that it can compute the frequencies of the events in each image to encode as described in [5]. The implementation of these algorithms allows obtaining an Huffman optimizer that, as a black box, takes as input the frequencies collected for a single image and generates optimal coding tables for it.

As shown in Figure 1, the optimizer requires a pre-processing phase in which the statistics of the current image are collected. It's not always possible implementing

F. Roli and S. Vitulano (Eds.): ICIAP 2005, LNCS 3617, pp. 711 – 718, 2005.

such computation in embedded systems for low-cost imaging devices, where limited resources are available. On the other hand, static Huffman coding can be properly managed collecting statistics for a class of data having relatively stable characteristics. This approach is also used in [7] where experiments with a previous phase of collection of statistics for source programs in four programming languages are successfully applied.

In this work we present an algorithm to generate Huffman coding tables for classes of images conceived using the Huffman optimizer. Applying the new generated tables instead of the standard ones, it's possible to obtain a further reduction in the final size of the bitstream without loss of quality.

New tables for three classes of images (landscape, portrait and document) were generated taking into account statistical similarities, measured in the "event space". For these particular classes psychovisual and statistical optimization of DCT quantization tables was presented in [1]. In [6] a technique to achieve an improved version of the JPEG encoder modifying the Huffman algorithm is presented; nevertheless using multiple tables to have better compression generates a bitstream not fully compliant with the standard version of the JPEG. Similar drawbacks can be found in ([2], [10]). Our proposed approach allows obtaining a standard compliant bitstream.
The paper is structured as follows: the next section describes the new proposed algorithm; section III reports the experimental results while in the next section a few words about statistical modeling of frequency distribution are reported. Finally, a brief conclusive section pointing to future evolutions is also included.

Fig. 1. The Huffman Optimizer

2 Algorithm Description

After the quantization phase, the JPEG encoder manages the 8x8 DCT coefficients in two different ways: the quantized DC coefficient is encoded as the difference from the DC term of the previous block in the encoding order (differential encoding) while the quantized AC coefficients are ordered into a "zig-zag" pattern to be better managed by entropy coding. In the baseline sequential codec the final step is the Huffman coding. The JPEG standard specification ([5]) suggests some Huffman tables providing good results on average, taking into account typical compression performances on different kinds of images. The encoder can use these tables or can optimize them for a given image with an initial statistics-gathering phase ([8], [11]). The Huffman optimizer is an automatic generator of dynamic optimal Huffman tables for each single image to encode by JPEG compression algorithm. If the statistics, taken as input by the optimizer block, refer to a dataset of images rather than to a single image, the optimizer generates new Huffman coding tables for the global input dataset (as shown in Figure 2).

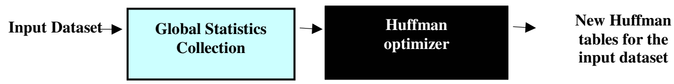

Fig. 2. A block description of the steps performed by the algorithm for automatic generation of Huffman coding tables for classes of images

To collect reliable data, the input dataset should contain a large number of images belonging to the same semantic category: we claim that images of the same class have similar distribution of events. The concept of semantic category has been also successfully used in ([1], [7]).

We considered three classes of images as input for our algorithm: landscape, portrait and document. The main steps of the proposed technique can be summarized as follows:

1. The algorithm takes as input a dataset of images belonging to the same semantic category;
2. Each single image of the considered dataset is processed collecting its statistics;
3. The statistics of the currently processed image are added to the global frequencies (i.e. the statistics corresponding to already processed images);
4. After the computation of statistics for the whole dataset, the values of global frequencies that are equal to zero are set to one;
5. In the last step the Huffman optimizer takes as input the global statistics and returns the new Huffman coding tables corresponding to the input dataset.

Step IV guarantees the generation of complete coding tables as well as the standard tables are complete. In the worst case it must be possible to associate a code to each generated event. The Huffman optimizer for single image, showed in Figure 1, doesn't create complete tables because all the events in the image are known, hence just such events must have an Huffman code. The described algorithm generates new Huffman tables that can be used to encode images belonging to each considered class (landscape, portrait and document). It is possible to improve compression performances of JPEG encoder by replacing the standard tables with the new generated tables for each specific class. Through this substitution, the previous collection of statistics is not needed because fixed Huffman tables are used to encode every image belonging to the three considered classes.

3 Experimental Results

The proposed method has been validated through an exhaustive experimental phase devoted to prove the effectiveness of the generated Huffman tables for each considered semantic class. Initially, in the retrieving phase, three default datasets were used with 60 images for each class chosen at different resolution and acquired by different consumer devices.

Our algorithm, starting from the corresponding class, provides as output the new coding tables for the landscape, portrait and document classes. The standard tables were simply replaced by the new generated Huffman tables.

The default datasets were considered as our training set (TS). So, first of all, we used the TS to make a comparison between the standard coding tables and the new optimized coding tables. Three different types of coding were made on our TS:

1. optimized coding: the images were coded using the optimal tables generated by the Huffman optimizer just for single image;
2. standard coding: the images were coded using the standard tables;
3. optimized coding for class: the images were coded using the new Huffman tables generated for each class.

As expected, the best coding is the optimized coding: it represents a lower bound in terms of dimension of the output bitstream because this coding uses optimal tables for each image to encode.

More interesting results concern the comparison between the standard coding and new tables for each class: the results obtained by optimized coding for class are slightly better than results obtained with default tables.

Table 1. Comparison between the file-size obtained with the standard tables and the derived tables for specific class

Training	LD	MD	Gain	σ
Landscape	-1.6%	19.3%	**5.3%**	4.5%
Portrait	-3.1%	10.6%	**5.9%**	3.2%
Document	-1.3%	11.9%	**1.8%**	2.1%

3.1 Experiments on New Dataset

In order to validate the preliminary results just described above, we performed a further test on three new datasets. The datasets for each class are composed by: 97 landscape images, 96 portrait images and 89 document images. These images were also chosen randomly and they were different from the previous collected ones. Using the corresponding tables properly learned on the training set, all images have been coded. In the table II are reported the relative results that confirm the previous analysis. Just for completeness, in Figure 3 are reported three different plots, corresponding to the measured bpp (bit-per-pixel), obtained coding each image of the new dataset, with the corresponding optimized table for portrait class. Similar results have been measured for the other two classes under investigation.

Table 2. Comparison between standard coding and coding with tables for each class over a large dataset of images not belonging to the training set

Dataset	LD	MD	Gain	σ
Landscape	-0.2%	14.2%	**5.8%**	3.8%
Portrait	-4.3%	13.8%	**5.1%**	4.5%
Document	0.02%	7.2%	**4.1%**	1.7%

Table 3. Percentages of improvement for the landscape class with variable QF

QF	Gain	QF	Gain
10	**18.7%**	40	**7.5%**
20	**13%**	60	**4.3%**
30	**9.7%**	70	**2.2%**

3.2 Experiments Using Different Quality Factory

Experiments devoted to understand the behavior of the new Huffman tables for each class when the Quality Factor (QF) assumes different values are here described. The QF is a JPEG parameter which allows choosing the preferred ratio quality-compression, through a direct modification (usually by a multiplicative factor) of the quantization tables. First, a small set of images of the new dataset belonging to the each class, was randomly chosen, just to evaluate in the range [10, 80] the bpp values obtained by coding with optimized tables for the class. As shown in Figure 4, by using specific tables for landscape class is better than using standard tables. Similar results have been measured for the other two classes under investigation. To further validate these results, the

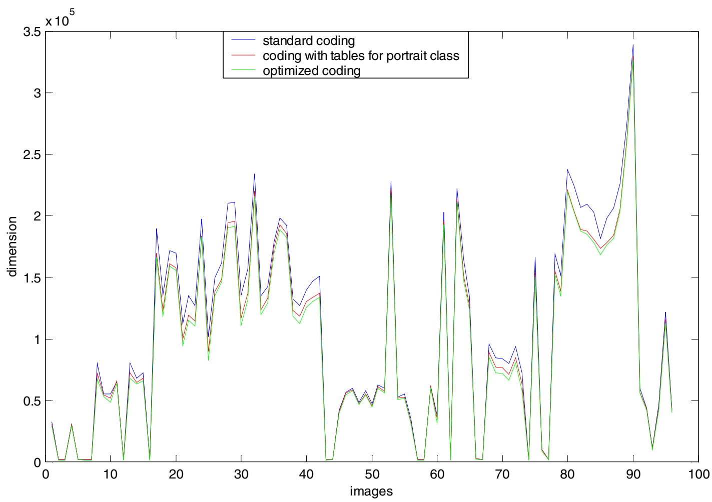

Fig. 3. Three types of coding applied to the new datasets.for portrait images. In all graphics the standard coding is represented by a blue line, the optimized coding for class is represented by a red line while the optimized coding is represented by a green line. In the Y axis of each graphic the dimension is reported in bytes.

three types of coding tables (standard, optimized and optimized for class) were repeated on the overall new dataset (97 landscape images) considering increasing values of the QF. Table III reports the percentages of improvement found for some values of the QF referred to the overall landscape class. Such experiments seem to confirm how the overall statistics of each class, have been really captured and effectively reported on the Huffman tables. Also the best performances have been noted at lower bit-rate suggesting how there is a sort of regularity among the events generated and coded into the tables.

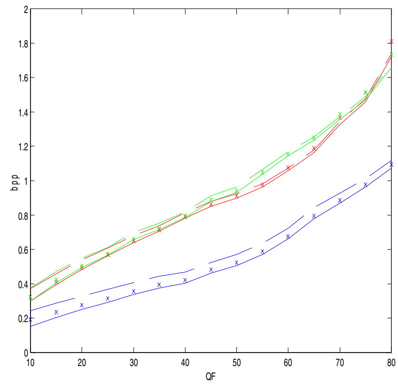

Fig. 4. Three types of coding of three landscape images with variable QF in the range [10, 80]. For each image: the optimized coding is represented by a continuous line, the standard coding is represented by an outlined line and the coding with tables for the landscape class is represented by the symbol X. In the Y axis the bpp (bit per pixel) values are reported while the X axis reports the different values assumed by the QF.

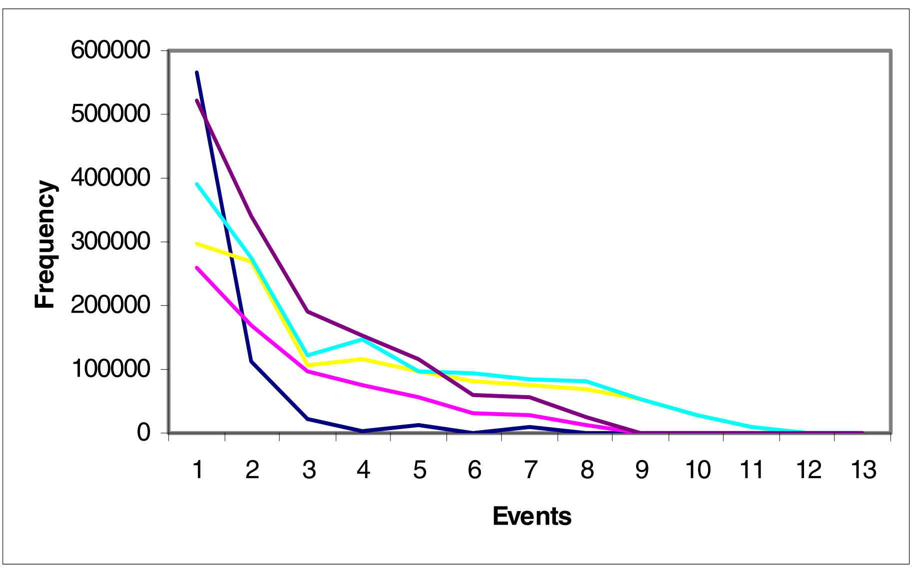

Fig. 5. DC Events frequency distribution for different quality factor values

4 Statistical Modeling

During the experiments performed in the test phase an interesting regularity between quality factor and events frequency has emerged. Such evidence is reported in Figure 5, where the plot corresponding to the DC events frequency distribution for different quality factor values (in the range [0, 100]), just for a single semantic class is showed. We claim that it should be possible fit some mathematical model for these curves capturing the underlying statistical distribution ([4]). In our experiments we have chosen linear spline in order to model the curves with a suitable mathematical function. In this way, just through a homotopyc function between linear spline curves it is possible to adapt the corresponding tables coding for each event frequency distribution for a given QF, without using the statistical classification step described in the previous sections.

Preliminary results seem to confirm our conjecture and suggest us to continue to investigate in this direction; detailed experiments will be reported in the final paper.

5 Conclusions and Future Works

In this paper we presented a new algorithm for automatic generation of Huffman tables based on semantic classes of digital images. By the proposed algorithm we generated new fixed Huffman tables for the following classes: landscape, portrait and document. These new tables allow improving compression performances of JPEG algorithm if compared with standard tables. Future research will be devoted to extend such methodology to Huffman encoding of video compression standard (e.g MPEG-2 [9]).

Also the possibility to derive a reliable model able to fit the distribution of the collected events will be considered.

References

1. Battiato, S., Mancuso, M., Bosco, A., Guarnera, M.: Psychovisual and Statistical Optimization of Quantization Tables for DCT Compression Engines, In IEEE Proceedings of International Conference on Image Analysis and Processing ICIAP (2001) 602-606
2. Bauermann, I., Steinbach, E.: Further lossless compression of Jpeg Images. In Proceedings of PCS 2004 – Picture Coding Symposium, CA - USA, (2004)
3. Bookstein, A., Klein, S.T., Raita, T.: Is Huffman Coding Dead?, Computing, Vol.50 (1993) 279-296
4. Epperson, J.F.: An Introduction to Numerical Methods and Analysis, Wiley, 2001
5. ITU-CCITT Recommendation T.81 Information technology – Digital compression and coding of continuous-tone still images – Requirements and Guidelines (1992) | ISO/IEC 10918-1 (1992)
6. Lakhani, G.: Modified JPEG Huffman Coding. In IEEE Transactions on Image Processing, Vol.12 no.2 (2003) 159-169
7. McIntyre, D. R., Pechura, M.A.: Data Compression Using Static Huffman Code-Decode Tables. Commun. ACM, Vol.28 (1985) 612-616

8. Pennebaker, W.B., Mitchell, J.L.: JPEG - Still Image Compression Standard, Van Nostrand Reinhold, NY, (1993)
9. Pillai, L.: Huffman Coding, Xilinx, XAPP616 (v1.0) (2003)
10. Skretting, K., Husøy, J.H., Aase, S.O.: Improved Huffman Coding Using Recursive Splitting, In Proceedings of Norwegian Signal Processing, NORSIG 1999, Symp., Asker, Norway (1999)
11. Wallace, G.K.: The JPEG still picture compression standard, Commun. ACM, vol.34 (1991) 30-44