MODIFIED GOLOMB CODING ALGORITHM FOR ASYMMETRIC TWO-SIDED GEOMETRIC DISTRIBUTION DATA

Jian-Jiun Ding, Wei-Yi Wei, and Guan-Chen Pan

Graduate Institute of Communication Engineering, National Taiwan University, Taipei, Taiwan
djj@cc.ee.ntu.edu.tw, r97942024@ntu.edu.tw, r99942126@ntu.edu.tw

ABSTRACT

Golomb coding is useful for encoding geometrically distributed data and plays an important role in advanced compression techniques, such as JPEG-LS and H.264. In this paper, a new Golomb coding method for asymmetric two-sided geometrically distributed data is proposed. An asymmetrical model means that the value of x can be positive or negative, but the ratio $P(x = -n)/P(x = -n-1)$ is unequal to $P(x = n)/P(x = n+1)$. The asymmetric model is more suitable for modeling the practical case because many data have higher probability to be positive (or negative) than to be negative (or positive) in nature. Two simulation examples are given: One is to encode the object boundaries for binary image compression and the other one is to encode the DC differences in JPEG. Both simulation results show that the proposed asymmetric two-sided Golomb coding algorithm outperforms other methods and has higher ability for data compression.

***Index Terms*—** Image compression, data compression, Huffman code, Golomb code, JPEG

1. INTRODUCTION

The Golomb code [1][2] is suitable for encoding the data whose probability has a geometrical distribution. It has been adopted by some advanced compression algorithms, such as JPEG-LS and H.264 [3]. Unlike the Huffman code, the Golomb code does not require a coding table. Therefore, if one uses the Golomb code instead of the Huffman code to encode a geometrically distributed data, such as the AC coefficients and the zero run length in JPEG, higher compression ratio can be achieved.

In this paper, the Golomb code is modified to encode the data that has an ***asymmetric*** *two-sided geometric distribution* (ATSGD). That is, the value of a data can be positive or negative and the probability distribution of the data has the form of

$$P_S[s] = \begin{cases} Ap_1^s & ,\text{for } s \ge 0 \\ Ap_2^{|s|} & ,\text{for } s < 0 \end{cases}. \quad (1)$$

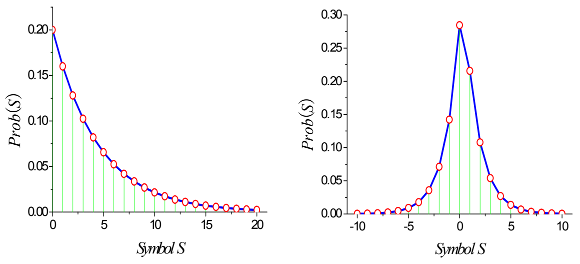
$$\text{where } A = \frac{(1-p_1)(1-p_2)}{1-p_1p_2} \quad \text{and } \boldsymbol{p_1 \neq p_2}. \quad (2)$$

In (1) and (2), $p_1 \neq p_2$ means that the probability that the data has a positive value is unequal to the probability that the data has a negative value. Moreover, the ratio of $P_S[n]/P_S[n+1]$ is also unequal to $P_S[-n]/P_S[-n-1]$. In our daily life, there are many conditions that the data has an ATSGD. For example, the probability that the population of a country increases is higher than the probability that the population decreases. The probability that the weight of a child increases is higher than the probability that the weight decreases.

In image compression, there are also many cases where the data has an ATSGD. When encoding the boundaries of objects, the curve is more possible to be convex than to be concave. Since the convex case has higher probability, it is proper to model the height of each segment of the curve by an ATSGD (illustrated in Section 4.1). Moreover, in the DC differential coding process of JPEG, if the previous block has lower DC value, then the difference between the DC value of the current block and that of the previous block has higher probability to be positive (illustrated in Section 4.2). For the examples described as the above, using the proposed modified Golomb code will achieve higher compression ratio than using the Huffman code and the original Golomb code. (Note that, although the Huffman code is optimal in theory, the **coding table** is required.)

This paper is organized as follows. The geometric distribution and Golomb coding are reviewed in Section 2. In Section 3, the proposed modified Golomb code for encoding an ***asymmetric*** *two-sided geometrically distributed data* is presented. In Section 4, two examples that use the proposed modified Golomb code to encode the object boundaries for binary image compression and encode the DC differences in the JPEG process are shown. In Section 5, we make a conclusion.

2. GEOMETRIC DISTRIBUTION AND GOLOMB CODING



(a) One-sided geometric distribution

(b) Two-sided geometric distribution

Figure 1 – One-sided and the two-sided geometric distributions.

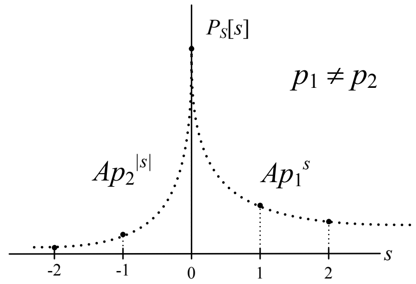


Figure 2 – The proposed modified Golomb code is suitable for encoding the data whose probability has an asymmetric two-sided geometric distribution (asymmetric TSGD).

2.1. Geometric Distribution

The geometrical distribution can be classified into two types. One is the *one-sided geometric distribution* (OSGD), and the other one is the *two-sided geometric distribution* (TSGD). The illustration of the OSGD and the TSGD are shown in Fig. 1. The data with the OSGD is modeled as follows:

$$P_S[s] = (1-p)p^s \quad \text{where } s \geq 0. \tag{3}$$

and p is the common ratio ($p \in (0, 1)$). By contrast, the data with the TSGD can be modeled by (1). In (1), if $p_1 = p_2$, then the data is said to have a symmetric TSGD. Otherwise, it has an asymmetric TSGD (ATSGD), see Fig. 2.

2.2. Golomb Coding

Golomb coding [3][4][5][6][7][8] is a lossless data compression method and has been adopted by JPEG-LS and H.264 [3]. The Golomb code is fully equivalent to the Huffman code if the probability of the data has a OSGD. The Golomb code has a tunable parameter m, which is known as *Golomb parameter*. In the OSGD case, the optimal Golomb parameter m can be estimated by:

$$m(p) = \left\lceil -\frac{\log(1+p)}{\log(p)} \right\rceil \tag{4}$$

where p is the common ratio defined as in (3) and $\lceil\ \rceil$ means rounding toward infinity (i.e., the ceiling operation). If the data is not geometrically distributed, one can approximate its probability by a geometric distribution function and use the Golomb code to encode it. Although in this condition the optimal coding results may not achieve, since the Golomb code does not require a coding table, which is needed by the Huffman, using Golomb code always achieve a lower bit rate than using the Huffman code in practice.

In the symmetric TSGD case, one can use an extra bit to represent the sign and then apply the method the same as that of the OSGD case to encode the data. In the ATSGD case, we suggest that it is proper to apply the proposed method in the next subsection to encode the data.

3. PROPOSED MODIFIED GOLOMB CODES FOR ASYMMETRIC TWO-SIDED GEOMETRICALLY DISTRIBUTED DATA

In image compression, sometimes the absolute value of a data has a geometric distribution but the probability that the data has a positive value is unequal to the probability of the negative value case. In this case, one can use (1) to model the probability distribution of the data, which is also depicted in Fig. 2.

To encode the data S whose probability has the form as in Fig. 2, a mapping function can be utilized to map S into a new data $\tilde{S}$ such that the value of $\tilde{S}$ is non-negative and nearly has a geometric probability distribution

First, assume that the probability parameters p_1 and p_2 in (1) have the following relationship

$$p_2 = p_1^k \tag{5}$$

where k is called the *asymmetric parameter*. Then, from (1), one can verify that

$$P_S(s) < P_S(-t) \tag{6}$$

$$\text{where } s > 0, \quad t = 1, 2, \ldots, \lceil s/k \rceil - 1,$$

Moreover,

$$P_S(-t) \leq P_S(s) \tag{7}$$

$$\text{where } t \geq 0, \quad s = 0, 1, \ldots, \lfloor k|t| \rfloor,$$

and $\lfloor\ \rfloor$ means rounding toward zero (i.e., the flooring operation). Therefore, from (6) and (7), one can apply the following mapping function to generate $\tilde{S}$ from S:

$$\tilde{s} = \begin{cases} s + \left\lceil \dfrac{s}{k} \right\rceil - 1 & \text{, for } s > 0 \\ |s| + \lfloor k|s| \rfloor & \text{, for } s \leq 0 \end{cases} \tag{8}$$

where k is defined as in (5), i.e., $k = \log p_2 / \log p_1$. Eq. (8) is a one-to-one mapping operation. It comes from sorting the magnitude of $P_S(s)$ according to (6) and (7) and one can verify that the probability that $\tilde{S} = s_1$ is always no less than the probability that $\tilde{S} = s_2$ if $s_1 < s_2$.

Moreover, if $n = s + \lceil s/k \rceil - 1$ where $s > 0$, then

$$s + s/k - 1 \le n < s + s/k\,, \tag{9}$$

$$s > \frac{k}{k+1} n \quad \text{and} \quad s \le \frac{k}{k+1}(n+1)\,,$$

$$Ap_1^{\frac{k}{k+1}(n+1)} \le P_{\tilde{S}}(n) = Ap_1^{s} < Ap_1^{\frac{k}{k+1}n}\,. \tag{10}$$

Similarly, if $n = |s| + \lfloor k|s| \rfloor$ where $s \le 0$, then

$$|s| + k|s| - 1 < n \le |s| + k|s|\,, \tag{11}$$

$$|s| \ge n/(k+1) \quad \text{and} \quad |s| < (n+1)/(k+1)\,,$$

$$Ap_1^{\frac{k}{k+1}(n+1)} = Ap_2^{\frac{n+1}{k+1}} < P_{\tilde{S}}(n) = Ap_2^{|s|} \le Ap_2^{\frac{n}{k+1}} = Ap_1^{\frac{k}{k+1}n}\,. \tag{12}$$

Therefore, from (10) and (12),

$$p_3^{n+1} \le \frac{P_{\tilde{S}}(n)}{P_{\tilde{S}}(0)} \le p_3^{n}\,, \tag{13}$$

$$\text{where } p_3 = p_1^{\frac{k}{k+1}}\,. \tag{14}$$

That is, although $\tilde{S}$ is not geometrically distributed, one can use the one-sided geometric distribution with the common ratio p_3 to approximate the probability distribution of $\tilde{S}$. It means that $\tilde{S}$ can be encoded using a Golomb code in the one-sided case [1][2], but in (4) p is replaced with p_3:

$$m = \left\lceil -\frac{\log(1+p_3)}{\log(p_3)} \right\rceil. \tag{15}$$

Therefore, one can use the procedure summarized as in Table I to encode the data whose probability distribution has the ATSGD form as in Fig. 2. We call it the modified Golomb code in the ATSGD case.

Table I. Proposed modified Golomb coding algorithm for the data with asymmetric two-sided geometric distribution (ATSGD).

Step 1: Estimate the probability parameters p_1 and p_2

Step 2: Determine the asymmetric parameter k by

$$k = \frac{\log p_2}{\log p_1}.$$

Step 3: Map the original two sided data s into $\tilde{s}$:

$$\tilde{s} = \begin{cases} s + \left\lceil \dfrac{s}{k} \right\rceil - 1 & \text{, for } s > 0 \\ |s| + \lfloor k|s| \rfloor & \text{, for } s \le 0 \end{cases}.$$

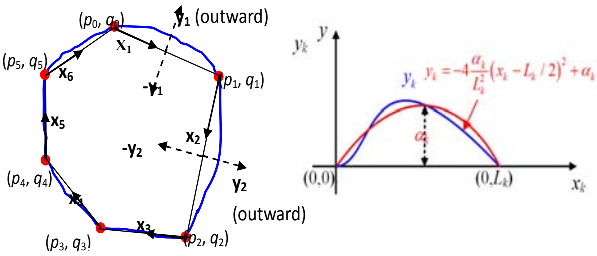
Step 4: Compute the probability parameter p_3 for the one sided data $\tilde{s}$

$$p_3 = p_1^{\frac{k}{1+k}}$$

Step 5: Determine the Golomb parameter m for $\tilde{s}$ by

$$m = \left\lceil -\frac{\log(1+p_3)}{\log(p_3)} \right\rceil.$$

Step 6: Encode $\tilde{S}$ by the Golomb code with the parameter m.



(a) (b)

Figure 3 – (a) Most parts of an object are convex. (b) Using a 2nd order polynomial to approximate a curve.

Note that, as the original Golomb code, the proposed modified Golomb code for the ATSGD case also requires no coding table. One only has to record the value of m and k. In [9][10], an alternative way to encode the data with the ATSGD is proposed. However, instead of (1), they modeled the probability by $P_S[n] = Ap^{|s-d|}$, i.e., using the "off-centered" model. From the simulations in Section 4, we can see that although their method has good performance for encoding the data with the ATSGD, if the proposed modified Golomb code is applied, even higher compression ratio can be achieved.

4. SIMULATION RESULTS

In nature, there are many conditions that the data has an asymmetric TSGD and it is proper to use the proposed modified Golomb code to encode and compress it. Here, two examples are given, which verifies that the proposed modified Golomb code can achieve better performance than other methods for encoding the data with an asymmetric TSGD.

4.1. Example 1: Encoding the Boundaries of Objects in Binary Image Compression

To encode a binary image, one only has to record the boundaries of objects and indicate which regions have the value of 1. Thus, using the least number of data to encode the object boundaries is an important issue for binary image compression. In our simulations, the figures were acquired from the database provided by Achanta et al. [11].

In Fig. 3(a), it can be observed that, because most parts of an object are convex, after connecting the corners by straight lines, the differences between the true boundary curve and the straight lines are always positive (the outward direction is positive and the inward direction is negative).

To encode the boundaries of an object, an efficient way is to record the coordinates of corners by differential coding and use the 2nd order polynomial in (16) to approximate the curve between the corners:

(a) (b)

Figure 4 – Two binary images that are used in the simulations of object boundary encoding in Table II.

Table II. The numbers of bits (including those required for the coding table) used for encoding the boundaries of the objects in Fig. 4 when using the proposed modified Golomb codes and other methods.

	Huffman	Original Golomb Code + Sign Bit	Off-Centered Golomb Code	**Proposed** Modified Golomb Code for ATSGD
Fig. 4(a)	241 bits	152 bits	143 bits	129 bits
Fig. 4(b)	292 bits	185 bits	185 bits	169 bits

Table III. The average numbers of bits (including those required for the coding table) used for encoding the boundaries of **1000 objects** in the database provided by [11] when using the proposed modified Golomb codes and other methods.

Huffman	Original Golomb Code + Sign Bit	Off-Centered Golomb Code	**Proposed** Modified Golomb Code for ATSGD
203.941 bits	127.378 bits	126.231 bits	114.807 bits

$$y_k = -\frac{4\alpha_k}{L_k^2}\left(x_k - \frac{L_k}{2}\right)^2 + \alpha_k . \qquad (16)$$

The illustration of (16) is shown in Fig. 3(b). If the corners are ordered in the clockwise direction, then y_k always has the direction outward the object, as in Fig. 3(a). Since in (16) L_k can be determined by the coordinate of the corners. To record the 2nd order polynomial in (16), one only has to record the value of α_k. The value of $|\alpha_k|$ is near to have a geometric distribution and suitable for being encoded by the Golomb code. However, it is shown in Fig. 3(a) that most parts of an object are convex. Therefore, the value of α_k in (16) has higher probability to be positive than to be negative. Thus, it is more proper to use the proposed modified Golomb code for the ATSGD case instead of the original Golomb code to encode the value of α_k and improve the compression efficiency.

The simulations of using the Huffman code, the original Golomb code [1][2], the off-centered Golomb code [9][10], and the proposed modified Golomb code to encode the boundaries of the two objects in Fig. 4 and the 1000 objects provided by [11] were performed and the results were shown in Tables II and III. The simulation results show that our proposed coding algorithm indeed has better performance for coding the object boundaries and hence has higher ability to compress binary images.

Note that, although the Huffman code is optimal in theory, since it requires the "coding table", the number of bits required by the Huffman coding algorithm is higher than those of the proposed algorithm and other Golomb based algorithms.

4.2. Example 2: Using the Proposed Code for DC Term Differential Coding in JPEG

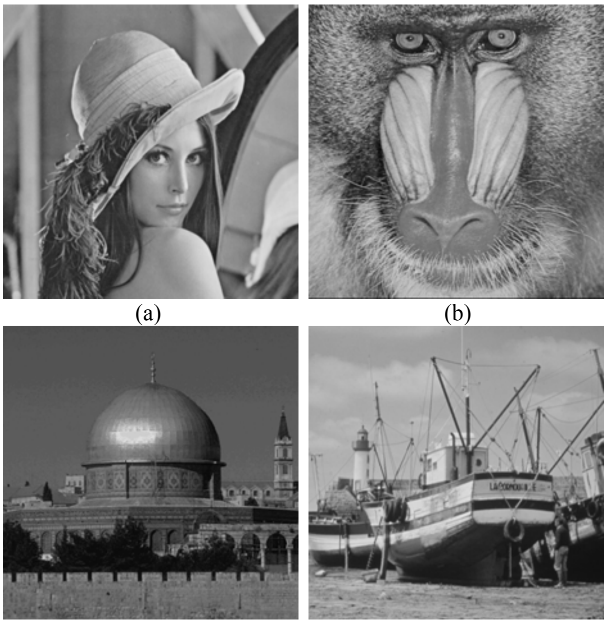
In the JPEG standard, the differential coding method is applied for the DC terms. First, the differences of the DC values of adjacent blocks are calculated. Then, the Huffman code is used for encoding the DC differences. Nonetheless, it can be observed that the DC difference has some characteristics:

(a) If the DC value of the $(m, n)^{th}$ block is denoted by $D[m, n]$, then the value of $|D[m, n] - D[m, n-1]|$ always has a geometric distribution.

(b) If $D[m, n-1]$ is low, then $D[m, n] - D[m, n-1]$ has a higher probability to be positive. If $D[m, n-1]$ is high, then $D[m, n] - D[m, n-1]$ is more probable to be negative.

Due to the above reasons, it is proper to the proposed modified Golomb code in the ATSGD case instead of the Huffman code and the original Golomb code to encode the DC difference in the JPEG procedure. One can use the ATSGD as in Fig. 1 to model the probability distribution of $D[m, n] - D[m, n-1]$. If $D[m, n-1]$ is small, then p_1 is larger than p_2 (i.e., in (5) k should be larger than 1). If $D[m, n-1]$ is large, p_1 is smaller than p_2 and the value of k in (5) should be smaller than 1.

The simulation results of using the Huffman code, the original Golomb code [1][2], the off-centered Golomb code [9][10], and the proposed modified Golomb code for DC term differential coding in the JPEG process were given in Table IV. The four tested images were as in Fig. 5. The simulation results in Table IV showed that our proposed coding algorithm absolutely has better performance for coding the DC differential terms in the JPEG procedure and hence has higher ability for image compression.



(a) (b) (c) (d)

Figure 5 – Four 512×512 test images used in the simulations of DC term differential coding in Table IV.

Table IV. Number of bits (including those required for the coding table) required for DC term differential coding in the JPEG procedure when using the Huffman code, the original Golomb code, the off-centered Golomb code, and the proposed modified Golomb code.

	Huffman	Original Golomb Code + Sign Bit	Off-Centered Golomb Code	**Proposed** Modified Golomb Code for ATSGD
Fig. 5(a)	23705 bits	23133 bits	23049 bits	21988 bits
Fig. 5(b)	23673 bits	21574 bits	21625 bits	20639 bits
Fig. 5(c)	16944 bits	17209 bits	15238 bits	13064 bits
Fig. 5(d)	20309 bits	20725 bits	19526 bits	17879 bits

5. CONCLUSION

In this paper, a modified Golomb coding algorithm based on the asymmetric TSGD model is proposed. Unlike the original Golomb code, our proposed code is suitable for dealing with the data whose probability is not symmetric in the positive and the negative cases. The simulation results show that the proposed code based on the asymmetric TSGD model has better performance than the Huffman code, the original Golomb code, and the off-centered Golomb code. It is useful for binary image compression and can be applied for DC term differential coding in the JPEG procedure. Many other image coding problems, such as coding the motion vector in MPEG, are also the potential applications of the proposed modified Golomb code in the ATSGD case.

REFERENCES

[1] S. W. Golomb, "Run Length Encodings," *IEEE Trans. Information Theory*, vol. 12, pp. 399-401, 1966.

[2] K. Sayood, "Golomb Codes," *Introduction to Data Compression*, pp. 61-64, 2000.

[3] I. Richardson, *H.264 and MPEG-4 Video Compression: Video Coding for Next Generation*, John Wiley and Sons, 2003.

[4] R. F. Rice, "Some Practical Universal Noiseless Coding Technique – Part I and Part III," Tech. Rep. JPL-79-22 and JPL-91-3, Jet Propulsion Laboratory, Pasadena, CA, Mar. 1979 and Nov. 1991.

[5] K. M. Cheung, and P. Smyth, "A High-Speed Distortionless Predictive Image Compression Scheme," *Proc. Int'l Symposium on Information Theory and Its Applications*, pp. 467-470, Nov. 1990.

[6] A. Said, "On the Determination of Optimal Parameterized Prefix Codes for Adaptive Entropy Coding," *HP Labs Report*, Palo Alto, CA, Apr. 2006.

[7] S. Xue, Y. Xu, and B. Oelmann, "Hybrid Golomb Codes for a Group of Quantised GG Sources," *IEEE Proceedings-Vision, Image and Signal Processing*, vol. 150, no. 4, pp. 256-260, Aug. 2003.

[8] J. J. Ding, S. C. Pei, W. Y. Wei, H. H. Chen, and T. H. Lee, "Adaptive Golomb code for joint geometrically distributed data and its application in image coding," *APSIPA Annual Summit and Conference*, pp. 302-305, Biopolis, Singapore, Dec. 2010.

[9] N. Merhav, G. Seroussi, and M. J. Weinberger, "Modeling and Low-Complexity Adaptive," *Proc. of the Int'l Conference on Processing*, vol. 2, pp. 353-356, Sept. 1996.

[10] G. Seroussi, and M. J. Weinberger, "On Adaptive Strategies for an Extended Family of Golomb-Type Codes," *Proc. DCC'97*, pp. 131-140, 1997.

[11] R. Achanta, S. Hemami, F.Estrada, and S.Susstruck. "Frequency-Tuned Salient Region Detection." *CVPR*, pp. 1597-1604, 2009.

[12] N. Merhav, G. Seroussi, and M.J. Weinberger, "Optimal Prefix Codes for Sorces with Two-Sided Geometric Distributions," *IEEE Trans. Inf. Theory*, vol. 46, no. 1, pp. 120-135, Jan. 2000.