Order Adaptive Golomb Rice Coding for High Variability Sources

Adriana Vasilache
Nokia Technologies, Tampere, Finland
Email: adriana.vasilache@nokia.com

***Abstract*—This paper presents a new perspective on the adaptive Golomb Rice codes that is especially suitable for sources having a highly variable distribution in time. Instead of adapting the Golomb Rice parameter, the encoder adapts the order of the symbols based on a count of occurrences measure. The proposed order adaptive Golomb Rice method is compared against different versions of adaptive arithmetic encoder at the encoding of real audio data stereo parameters. The proposed method shows very fast adaptability in the presence of rapidly changing data with respect to the initial data statistics.**

I. Introduction

The encoding of integers is very present in the current media encoders when it comes to efficiently transmitting the codevector indexes [1], [2]. The entropy encoding is mostly done by Huffman encoding, arithmetic encoding or Golomb Rice encoding. The efficiency of the arithmetic coding, combined with the use of contexts and an adaptive mechanism [3] is rarely matched by the Huffman encoder or the Golomb Rice encoder. However, the latter two compensate with a lower complexity of encoding and decoding and, depending on the data statistics, their efficiency may be enough. In addition, the use of Golomb Rice codes enables the encoding of a source with unknown number of symbols and have even lower encoding complexity and storage requirements.

In this paper we study a practical case, the encoding of stereo level parameters, where the statistics of the data is highly variable and the adaptation mechanism provided by the arithmetic coding is not versatile enough. The use of multiple contexts could increase the encoding efficiency, but at the same time it would increase the encoding complexity and the table storage requirements. As alternative we propose a novel approach of adaptive Golomb Rice coding which adapts very fast to the data statistics changes and has virtually no storage requirements.

After this introduction we will present the existing approach of using adaptivity for Golomb Rice encoding followed by the description of several source variability measures. The fourth section will describe the proposed method with its different variants while the fifth section will present numerical results on real audio stereo pameter data.

II. Generic adaptive Golomb Rice coding

The Golomb Rice coding is a well known method for variable rate encoding of integers. The first version was proposed by Golomb [4] and applied to encoding of run-lengths. Its advantage was that it had no lookup table and it could be used for sources whose number of different symbols is not known. It is parameterized by an integer m and can directly convert any positive integer symbol into a codeword. For the special case when m is a power of two, $m = 2^k$ for a positive integer k, the coding is known by Golomb Rice coding [5]. Even though the Golomb Rice coding might be suboptimal with respect to coding efficiency, it has been used in image compression algorithms in JPEG-LS [6] due to its very low encoding and decoding complexity.

The Golomb code is optimal for geometrically distributed sources:

$$Pr(X = k) = (1-p)^k p, \text{for } k = 0, 1, 2, 3, ... \qquad (1)$$

Gallager and van Voorhis [7] have derived the optimal parameter k for a Golomb code for the geometrically distributed source from Equation 1, the integer that satisfies $p^k + p^{k+1} \le 1 < p^k + p^{k-1}$.

When the data statistics changes in time, obviously the optimal Golomb Rice parameter changes as well. There have been thorough studies on how to select and adapt the optimal parameter in [8], [9]. Also strategies for adapting the Golomb Rice parameter for correlated sources have been presented in [10]. The use of contexts has been proposed for Golomb codes optimization in [11]. In [12] the authors have proposed to use the LPC spectral envelope for the adaptation of the Golomb Rice parameter at each frequency bin for a frequency domain audio encoder.

III. Source statistics variability

When the number of symbols of a source is larger than the number of values to sequentially encode, it is difficult to assess the statistics of the current data. In other words, dimension of the vector of one realization of the data is low, the data statistics of each vector is almost unknown to the encoder, or it may be quite distant to the a priori data distribution. We consider as measures characterizing the variability in data statistics the Kullback-Leibler divergence and the Bhattacharyya distance, and as variability of a realization the quartile coefficient of dispersion.

The Kullback-Leibler (KL) divergence from p_1 to p_2, when p_1 and p_2 are discrete probabilities is defined as [13]:

$$D_{KL}(p_1||p_2) = \sum_i p_1(i) \log_2 \frac{p_1(i)}{p_2(i)}. \qquad (2)$$

ISBN 978-0-9928626-7-1 © EURASIP 2017

Since the logarithms are taken in basis 2, the KL divergence is measured in bits.

For probability distributions p_1 and p_2 over the same domain, the Bhattacharyya distance is defined as [14]:

$$D_B(p_1,p_2) = -\ln\Big(\sum_x \sqrt{p_1(x)p_2(x)}\Big). \quad (3)$$

The quartile coefficient of dispersion is defined as [15]

$$C = \frac{Q_3 - Q_1}{Q_3 + Q_1} \quad (4)$$

where Q_1 and Q_3 are the first an the third quartiles for the data set.

IV. Order Adaptive Golomb Rice encoding

The data to be encoded consists of M dimensional vectors of positive integers. We are interested mostly in the case when M is relatively small compared to the number of possible symbols N, making hard to really decide on the data statistics at one encoding. The vector components can have thus N distinct values, from 0 to $N-1$.

Given the a priori distribution of the data an initial mapping of the symbols is built. The mapping of the symbols to be encoded is chosen such that the most probable symbol is assigned the value 0, the following most probable is assigned the value 1 and so on. In addition to the mapping, the encoder also stores a vector of counts for each distinct value. The vector of counts can be seen as an a priori histogram of the data. However the value of the counts vector can be also nonintegers. For each input symbol the encoding using the order adaptive Golomb Rice (OAGR) coder goes as follows:

- The input symbol x is assigned an ordered symbol x_o according to the a priori defined mapping.
- The ordered symbol x_o is selected to be encoded for the current input data.
- The vector of counts is updated
- The mapping is updated, to reflect the new order in the counts vector.
- Take next symbol.

The order of the Golomb-Rice encoder can be 0 or 1, but it has the same value for the entire frame.

In order to illustrate the process suppose there are $N = 9$ symbols, from 0 to 8. The vector of counts, which is initialized starting from an experimental histogram for the symbols $[0, 1, 2, 3, 4, 5, 6, 7, 8]$ could look like:

$$h = [1, 3, 5, 7, 9, 8, 6, 4, 2]$$

with the corresponding mapping

$$m = [8, 6, 4, 2, 0, 1, 3, 5, 7].$$

If the sequence of symbols to be encoded is

$$v = [5, 6, 4, 2, 4, 7, 4]$$

then for the first symbol, the symbol $m(5) = 1$ is encoded with GR code. The numbering starts from 0. The experimental counts vector h is updated such that

$$h(5) = 0.9 \cdot h(5) + 1$$

and

$$h(i) = 0.9 \cdot h(i), i \neq 5.$$

The vector of counts becomes then $h = [0.9, 2.7, 4.5, 6.3, 8.1, 8.2, 5.4, 3.6, 1.8]$ and the corresponding updated mapping is $m = [8, 6, 4, 2, 1, 0, 3, 5, 7]$. Note the changes in positions 4 and 5.

From the above description it becomes clear that the relative order between the counts vector components influences on the encoding efficiency. Therefore it is of interest to study different options to update the counts. The illustrative example is just a particular case. Two methods are considered. The first one interprets the counts as histogram data and the update is done by addition. We dubb this approach as additive approach. The second one, after appropriate normalization, interprets the counts as probability function and updates the corresponding probability vector through multiplication. More details follow in the subsequent subsections.

A. Additive model

The counts vector is denoted by $h = \{h_j\}, j = 0 : N-1$. After the encoding of a data vector component v_i the counts vector is updated by

$$h_j = h_j + f(j), j = 0 : N-1.$$

When the function $f(j) = f_l(j)$ is linear we have a linear additive model:

$$f_l(j) = \begin{cases} \frac{\sigma j}{v_i} & j \leq v_i \\ \frac{\sigma(N-1)}{N-1-v_i} - \frac{v_i \sigma}{N-1-v_i} & j > v_i \end{cases} \quad (5)$$

The parameter to be optimized is σ.

When the function $f(j) = f_e(j)$ is exponential we have an exponential additive model:

$$f_e(j) = \frac{C}{\sqrt{2\pi\sigma^2}} \exp^{\frac{-(v_i-j)^2}{2\sigma^2}} + \delta \quad (6)$$

The parameters allowing flexibility to the method are C, σ and δ.

B. Multiplicative model

In the multiplicative model the vector h is updated by:

$$h_j = h_j f(j), j = 0 : N-1$$

and the multiplicative factor can be linear or exponential.

V. Results

We have considered as data the stereo parameters of a parametric stereo/binaural encoder. Most precisely the stereo subband level differences are the object of study. The audio data consists of superwideband stereo or binaural speech, music, mixed content. The sampling rate for superwideband data is 16 kHz. The two channels are FFT transformed, aligned and divided in subbands. The subband energies for each channel are compared subband-wise. It means that for each of the b subbands, a subband level parameter is calculated as:

$$l_i = 10\log_{10}\frac{E_r^i}{E_l^i}, i = 0, b-1 \tag{7}$$

where $E_{l/r}^i$ is the energy of the subband i for the left or right channel. A positive value indicates that for the corresponding subband there is more energy on the right channel, while a negative value corresponds to more energy being present in the left channel.

At each 20 ms frame, a vector of $M = 12$ subband levels is calculated. In a stereo parametric encoder, the subband levels are scalarly quantized and the indexes must be encoded. These indexes, with values from 0 to $N-1$, where $N = 31$ is the number of codewords in the scalar quantizer, are the input data for the proposed entropy encoding method in this paper.

In the following we will present the results of the proposed variants of adaptive Golomb Rice encoder at encoding these parameters for a variety of audio signals. The results will also be compared with what an adaptive arithmetic encoder gives. Measures describing the statistics of the data to be encoded will be also calculated and presented.

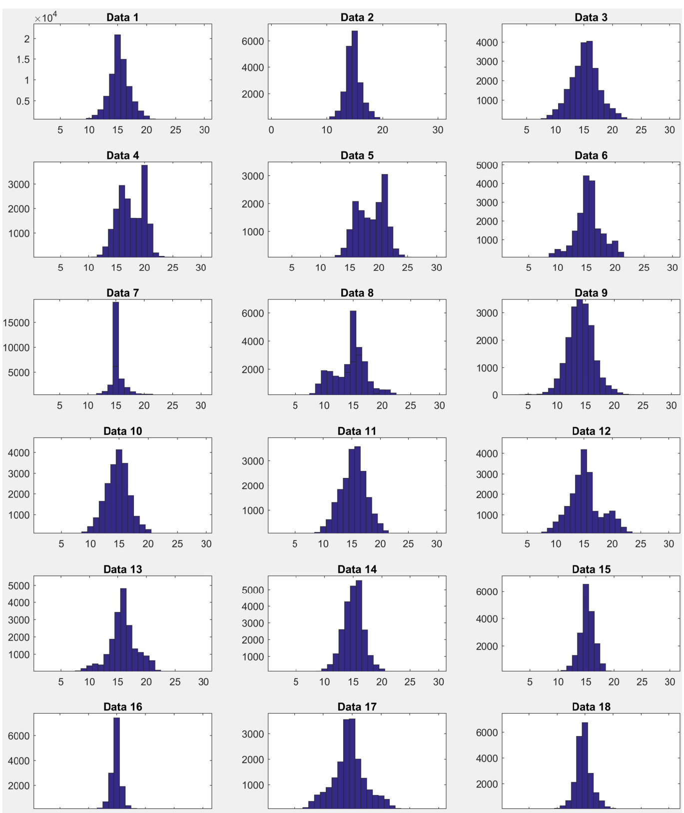

Fig. 1. Experimental histograms of the data set considered.

Given the significance of the data, on average it is expected that the indexes are symmetric with respect to the middle index and there are more central valued indexes. This is also experimentally confirmed by the histograms in Figure 1. There is a general tendency to have more indexes around the centre. However, by comparing the histograms, it can be even visually observed that they are different from one data set to another. The audio content type for each data set is described in Table I. Each data set contains 5 or 6 audio samples of approximately 8 seconds. The position of the sound sources/speakers differs as it can also be inferred from the histrograms. The capture microphone positioning also varies across the samples. The multiple speaker samples have the speakers at different physical position.

TABLE I
DATA (D) DESCRIPTION

D	Description	D	Description
1	Binaural music	10	Two overlapping speakers, music
2	Multiple speakers male	11	Four ovelapping speakers two by two
3	Clean speach	12	Four speakers, no overlap
4	Male Speech, kitchen noise	13	Female speech, kitchen noise
5	Female speech, cafeteria noise	14	Four speakers overlapping two by two, music
6	Female speech, kitchen noise	15	Female speech, car noise
7	Male and female speaker, no overlap	16	Male speech
8	Mixed speech and music, car noise	17	Three male speakers in big hall
9	Two speakers over music	18	Four male speakers

A. Coding efficiency

We compare three variants of adaptive arithmetic coding against three variants of the proposed Golomb Rice method. To preserve the frame errors resilience, no intra-frame correlation is taken into account in any of the methods.

Adaptive arithmetic coding has been used to encode stereo level indexes and the results are presented in Table II. Three variants of arithmetic coding have been tested using different probability initialization: AC1 - uniform initialization, AC2 - off-line collected probability; AC3 - artificial vector of counts used in the OAGR. In all cases the adaptation for the arithmetic coder is done like for the OAGR case, within the frame only.

KL is the average Kullback-Leibler divergence between the data for each frame and off-line collected probabilities. DC is the average quartile coefficient of dispersion over all frames. As reference, also the zero order entropy H on each set is given.

Table III presents the average number of bits used for encoding the same data sets with order adaptive Golomb Rice. There are three variants of the proposed method that are tested. The first one, GR1, uses an exponential multiplicative update function and a constant vector of counts initializer. The parameter values are $\sigma = 2.5$ and $\delta = 0.5$. All the values of counts are initialized to the value 1, at the beginning of each frame. The second method GR2, is exponential multiplicative update with artificial counts vector. The parameter values are $\sigma = 2.5$ and $\delta = 0.5$. The GR3 method uses linear additive update function with artificial counts vector and the last one GR4 uses linear multiplicative update function with artificial counts vector. The parameter values are $\sigma = 0.04$ and $\sigma = 0.1$.

TABLE II
NUMBER OF BITS USED BY ADAPTIVE AC OF STEREO LEVEL INDEXES.

D	AC1	AC2	AC3	KL	DC	H
1	53.72	**37.88**	50.20	4.57	0.10	36.3
2	51.56	**33.07**	49.62	6.21	0.07	30.8
3	54.25	43.23	50.79	5.62	0.10	40.2
4	52.31	54.24	51.63	7.11	0.07	37.6
5	53.29	65.86	53.09	8.64	0.08	39.5
6	54.76	41.90	50.72	4.87	0.12	39.0
7	53.84	*39.37*	50.41	4.50	0.11	37.0
8	54.57	49.87	51.60	6.82	0.13	40.9
9	55.09	42.78	50.92	6.25	0.12	38.5
10	54.59	40.69	50.61	5.31	0.11	38.3
11	54.04	42.32	50.65	5.29	0.10	39.3
12	53.53	48.64	51.44	7.96	0.10	43.5
13	54.66	43.30	50.78	4.33	0.12	38.7
14	53.73	**37.39**	50.19	4.52	0.10	35.7
15	52.21	**32.72**	49.59	4.32	0.08	30.8
16	48.23	**27.78**	48.73	6.71	0.06	23.2
17	54.60	47.36	51.40	7.10	0.12	42.7
18	51.52	**33.02**	49.61	6.24	0.07	30.7
Average	53.36	42.30	50.67	5.90	0.10	36.82

TABLE III
NUMBER OF BITS USED BY OAGR CODING OF STEREO LEVEL INDEXES.

D	GR1	GR2	GR3	GR4
1	47.41	40.64	39.74	40.18
2	43.28	37.04	35.96	36.00
3	48.27	41.84	**40.39**	42.55
4	45.84	38.90	**38.62**	41.65
5	48.22	**42.16**	42.63	47.40
6	50.11	45.26	**42.55**	45.11
7	50.60	42.91	40.78	42.24
8	49.26	44.13	**41.12**	45.54
9	49.70	43.46	**41.46**	43.77
10	48.92	42.19	*40.80*	42.39
11	47.59	41.02	**40.30**	41.52
12	47.43	41.60	**41.33**	43.28
13	50.00	44.62	**42.13**	45.24
14	47.50	39.75	38.77	40.33
15	44.07	37.49	36.27	36.65
16	39.55	33.66	33.27	31.87
17	50.64	44.98	**43.50**	45.68
18	43.24	36.97	35.90	35.95
Average	47.31	41.06	**39.75**	41.60

The artificial counts vector is generated such that its corresponding mapping gives an almost symmetric distribution with the maximum in the centre. The articifial counts vector is given by: $h = [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 30, 28, 26, 24, 22, 20, 18, 16, 14, 12, 10, 8, 6, 4, 2]$.

The best results for the AC and for OAGR are given in bold. It can be seen that for most of the data sets the OAGR performs better. The cases when the AC is slightly better are the ones where the data histogram is similar to the one presumed by the AC.

Analyzing the average values of the KL divergence and quartile dispersion coefficient, one notices some differences across the data sets. As expected the sets having higher KL divergence from the off-line collected probabilities are best encoded with the OAGR. Also, at a closer look, let's consider the Data set 4, where there is a bit performance difference between AC and OAGR. The KL divergence per frame between the frame data statistics and the off-line collected data statistics, for a sample of 8 seconds from Data set 4 is given in Figure 2. It can be observed that the KL divergence has different tendencies in different sections of the data. The same can be observed also when plotting the Bhattacharyya distance for the same data (Figure 3). From the above considerations we can say that the KL divergence per frame is a more useful indicator on the variability of the source.

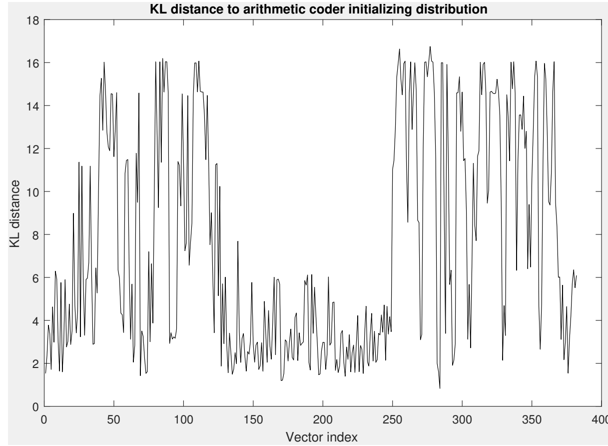

Fig. 2. Kullback-Leibler distance of the data probability distribution to the default, initializing probability distribution of the arithmetic encoder. Each vector has 12 components. One sample from Data set 4 is considered.

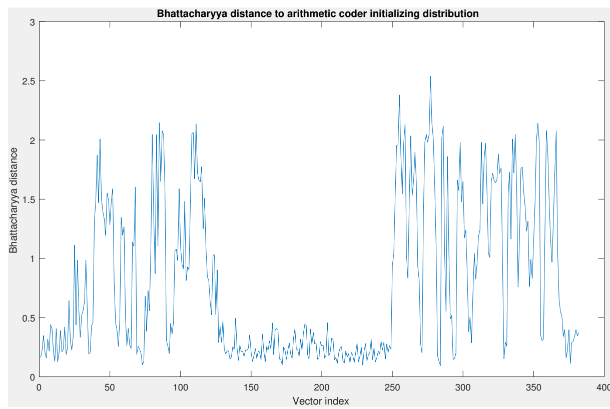

Fig. 3. Bhattacharyya distance of the data probability distribution to the default, initializing probability distribution of the arithmetic encoder. Each vector has 12 components. One sample from Data set 4 is considered.

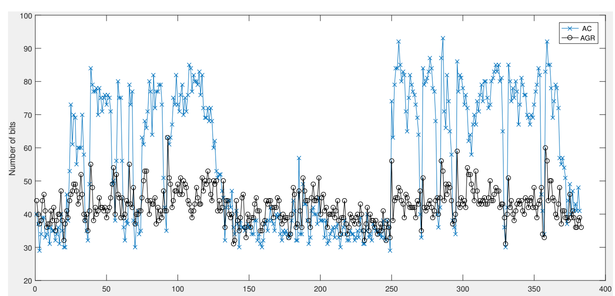

Fig. 4. Number of bits using AC and OAGR for one sample from Data set 4

ISBN 978-0-9928626-7-1 © EURASIP 2017

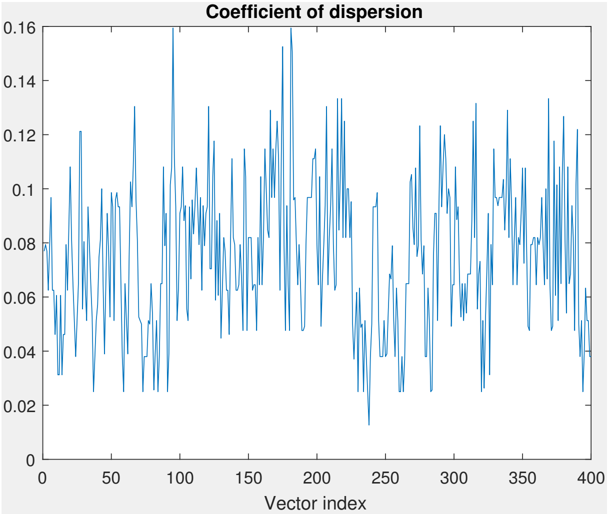

Fig. 5. Coefficient of dispersion for one sample from Data set 4

Correspondingly, when plotting the number of bits per frame used by the arithmetic encoder AC2 and the the OAGR variant GR3 it can be seen that the large difference in the data statistics cannot be coped with by the AC encoder, while the OAGR keeps the bit consumption at much lower levels and very similar to other frames (see Figure 4).

Not the same can be said for the coefficient of dispersion (Figure 5), for which no particular correlation with the source variability affecting the encoding performance can be visible. The variability of the source, or dispersion, within one data realization vector is not affecting the coding performance in the sense considered in this work.

As a counter-example, for the Data set 15, the KL divergence with respect to the one initializing the arithmetic encoder at each frame is presented in Figure 6. The performance of the AC for this data set is better than for the OAGR and this is justified by the closer resemblence of the test data distribution to the one stored in the AC encoder.

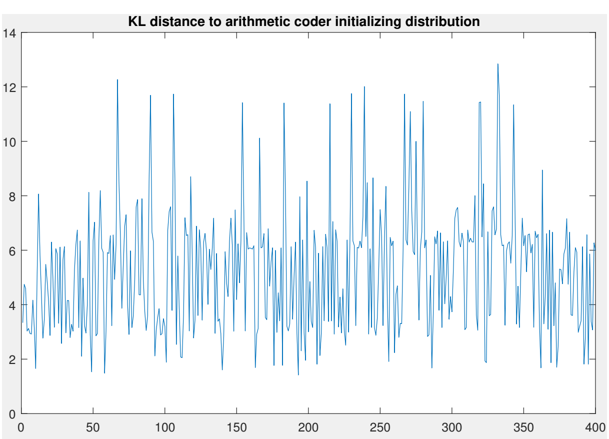

Fig. 6. Kullback-Leibler distance of the data probability distribution to the default, initializing probability distribution of the arithmetic encoder. One audio sample from Data set 15 is considered.

Regarding the complexity, the general advantage brought by the Golomb Rice encoding is no longer valid, since sorting of the counts vector needs to be performed. However, with respect to storage requirements only an initial table of counts must be stored, having the length equal to the number of symbols. Furthermore, since the vector of counts is artificially generated it can also be generated by the code, so virtually no storage is required.

VI. Conclusion

In the present work we have proposed a new perspective on the adaptivity of Golomb Rice codes. Instead of adapting the Golomb Rice parameter, the order of the symbols to be encoded is adapted on the fly. The approach allows for a greater adaptivity when it comes to sources with high time variability, as well as sources where the length of a realization vector is small compared to the number of possible symbols. Results on stereo parametrs of real audio data, when compared with arithmetic encoding show better overall performance for the proposed method.

References

[1] I. 14496-3:2009, "Information technology – coding of audio-visual objects – part 3: Audio," 2009, https://www.iso.org/standard/53943.html.

[2] H. Huang, H. Shu, and R. Yu, "Lossless audio compression in the new IEEE standard for advanced audio coding," in *Proceedings of ICASSP*, Florence, Italy, 2014, pp. 6934–6938.

[3] D. Marpe, H. Schwarz, and T. Wiegand, "Context-based adaptive binary arithmetic coding in the h.264/avc video compression standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 620–636, July 2003.

[4] S. W. Golomb, "Run-length encodings," *IEEE Transactions on Information Theory*, vol. 3, no. 12, pp. 399–401, 1966.

[5] R. F. Rice, "Some practical universal noiseless coding techniques," March 1979, Jet Propulsion Lab., Pasadena, CA, Tech. Rep. JPL-79-22,.

[6] M. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," *IEEE Transactions on Image Processing*, vol. 9, no. 8, pp. 1309–1324, August 2000.

[7] R. G. Gallager and D. C. van Voorhis, "Optimal source codes for geometrically distributed integer alphabets," *IEEE Transactions on Information Theory*, pp. 228–230, March 1975.

[8] A. Said, "On the determination of optimal parameterized prefix codes for adaptive entropy coding," April 2006, online: http://www.hpl.hp.com/techreports/2006/HPL-2006-74.pdf.

[9] A. Kiely, "Selecting the Golomb parameter in Rice coding," November 2004, online: http://ipnpr.jpl.nasa.gov/progress_report/42-159/159E.pdf.

[10] E. S. Hong and R. E. Ladner, "Optimal adaptation strategies for Golomb codes on correlated sources," in *Proceedings of International Conference on Image Processing 2005*, vol. 1, 2005, pp. 169–172.

[11] J.-J. Ding, H.-H. Chen, and W.-Y. Wei, "Adaptive Golomb code for joint geometrically distributed data and its application in image coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 4, pp. 661–670, April 2013.

[12] R. Sugiura, Y. Kamamoto, N. Harada, H. Kameoka, and T. Moriya, "Golomb-rice coding optimized via LPC for frequency domain audio coder," in *Proceedings of GlobalSIP 2014*, Atlanta, USA, 2014, pp. 1024–1028.

[13] T. M. Cover and J. Thomas, *Elements of information theory*, J. W. . Sons, Ed., 1991.

[14] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using a mean shift," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, South Carolina, USA, 2000, pp. 142–149.

[15] D. G. Bonett, "Confidence interval for a coefficient of quartile variation," *Computational Statistics & Data Analysis*, vol. 50, no. 11, pp. 2953–2957, July 2006.

ISBN 978-0-9928626-7-1 © EURASIP 2017