Universal True Lossless Combinatorial Data Compression

RODOLFO A. FIORINI
Department of Electronics, Information and Bioengineering (DEIB)
Politecnico di Milano University
32, Piazza Leonardo da Vinci, 20133 MI
ITALY
rodolfo.fiorini@polimi.it http://rodolfo.fiorini.faculty.polimi.it

Abstract: - Contemporary approaches to data compression vary in time delay or impact on application performance as well as in the amount of compression and loss of data. The very best modern lossless data compression algorithms use standard approaches and are unable to match spacetime high end requirements for mission critical application, with full information conservation (a few pixels may vary by com/decom processing). Advanced instrumentation, dealing with nanoscale technology at the current edge of human scientific enquiry, like X-Ray CT, generates an enormous quantity of data from single experiment. In previous papers, we have already shown that traditional ***Q*** Arithmetic can be regarded as a highly sophisticated open logic, powerful and flexible bidirectional formal language of languages, according to “Computational Information Conservation Theory” (CICT). This new awareness can offer competitive approach to guide more convenient, spatiotemporal lossless compression algorithm development and application. To test practical implementation performance and effectiveness on biomedical imaging, this new technique has been benchmarked by normalized spatiotemporal key performance index, and compared to well-known, standard lossless compression techniques.

Key-Words: - Lossless Compression, Arithmetic Geometry, Computed Tomography, X-rays, Biomedical Imaging, Compressed Sensing, Biomedical Cybernetics, Biomedical Engineering, Public Health, Healthcare.

1 Introduction

Contemporary approaches to data compression vary in time delay or impact on application performance as well as in the amount of compression and loss of data. Two approaches that focus on data loss are lossless (no data loss) and lossy (some data loss for higher compression ratio). Lossless compression is a class of data compression algorithms that allows the original data to be perfectly reconstructed from the compressed data. By contrast, lossy compression permits reconstruction only of an approximation of the original data, though this usually improves compression rates (and therefore reduces file sizes). Lossless compression is used in cases where it is important that the original and the decompressed data be identical, or where deviations from the original data could be deleterious. Typical examples are executable programs, text documents, and source code.

Advanced instrumentation, dealing with nanoscale technology at the current edge of human scientific enquiry, like X-Ray CT, generates an enormous quantity of data from single experiment [1]. Even in MicroCT or Discrete Tomography (DT) by electron microscopy, 2-D projection images are acquired from various angles, by tilting the sample, generating new challenges associated with the problem of formation, acquisition, compression, transmission, and analysis of an enormous quantity of data [2], [3]. During this time of exploding data growth, disk manufacturers have begun running into the physical limitations of current storage technology (e.g., disk platter bit density, data transfer, etc.) and to seek new technologies to rely on [4].

Different from natural images, advanced technology imaging and medical images generally have two special issues that should be noted in compression. First, they are sensitive to compression errors. Large distortion arising from lossy compression may invalidate their interpretation and diagnostic values. Second, especially the monochromatic images usually have extended dynamic range. Each pixel typically contains 16 or 12 bits per channel, compared to the 8-bit-depth pixels of common natural images. Nevertheless, both lossless and lossy compression schemes have been proposed to compression image application. Lossy compression (irreversible, i.e.

information dissipation) schemes can achieve much higher Compression Ratio (CR) than lossless ones, by allowing some distortion in reconstruction. However, for mission critical application, the lossy distortion level must have to be supervised by qualified experts to avoid possible legal and diagnostic problems, especially for medical application [5],[6].

Lossless compression (reversible, i.e. information conservation) schemes provide reconstructed images identical to the original ones, but suffer from relative lower compression ratio, typically between 1:2 and 1:4. Furthermore, a single compression ratio (CR) cannot serve as a guideline for the compression of medical images, as compression artifacts vary considerably with image content, scanning technique, and compression algorithm [7]. Traditional examples are executable programs, text documents, and source code.

1.1 Basic Techniques

Lossless compression is possible because most real-world data exhibits statistical redundancy. Most lossless compression programs do two things in sequence: the first step generates a statistical model for the input data, and the second step uses this model to map input data to bit sequences in such a way that "probable" (e.g. frequently encountered) data will produce shorter output than "improbable" data. The second compression stage consists of replacing commonly used symbols with shorter representations and less commonly used symbols with longer representations. The best modern lossless compressors use probabilistic models, such as "prediction by partial matching." In a further refinement of the direct use of probabilistic modelling, statistical estimates can be coupled to an algorithm called "arithmetic coding."

Many of the lossless compression techniques used for text also work reasonably well for indexed images, but there are other techniques that do not work for typical text that are useful for some images (particularly simple bitmaps), and other techniques that take advantage of the specific characteristics of images (such as the common phenomenon of contiguous 2-D areas of similar tones, and the fact that color images usually have a preponderance of a limited range of colors out of those representable in the color space). Therefore, current lossless compression methods can be grouped into two large development areas which address: a) Text or b) Signal & Image.

1.1.1 A-Text

1 Run Length Encoding (RLE)
2 EntropyCoding
3 Compressed Sensing
4 Grammar-Based

(A1-B1) Run Length Encoding (RLE)

RLE replaces data by a (length, value) pair, where “value” is the repeated value and “length” is the number of repetitions. It is not useful with files that do not have many repetitions as it could greatly increase the file size [8]. In the case of image, it may have areas of color that do not change over several pixels; instead of coding "red pixel, red pixel, ..." the data may be encoded as "279 red pixels". This is a basic example of RLE. There are many schemes to reduce file size by eliminating redundancy. This technique is especially successful in compressing bi-level images since the occurrence of a long run of a value is rare in ordinary gray-scale images. A solution to this is to decompose the gray-scale image into bit planes and compress every bit-plane separately. RLE method can undergo different variation implementatios [9].

A2 Entropy Coding

Entropy represents the minimum size of dataset necessary to convey a particular amount of information. Huffman, LZ (Lempel-Ziv), and arithmetic coding are the commonly used entropy coding schemes. Huffman coding (HC) [10] creates an unprefixed tree of non-overlapping intervals, where the length of each sequence is inversely proportional to the probability of that symbol needing to be encoded. The more likely a symbol has to be encoded, the shorter its bit-sequence will be. In other words, HC utilizes a variable length code in which short code words are assigned to more common values or symbols in the data, and longer code words are assigned to less frequently occurring values. Modified Huffman coding [11] and adaptive Huffman coding [12] are two examples among many variations of Huffman’s technique.

LZ (Lempel-Ziv) LZ coding replaces repeated substrings in the input data with references to earlier instances of the strings. It had two major implementations: LZ77 and LZ78. They are the two algorithms published in papers by Abraham Lempel and Jacob Ziv in 1977 [13] and 1978 [14]. Several compression methods, among which LZW (Lempel-Ziv-Welch) [15] is one of the most well known methods, have been developed based on these ideas. Variations of LZ coding are used in the Unix utilities Compress and Gzip. To this category belongs also "Arithmetic coding" [16],[17] that represents a message as some finite intervals between 0 and 1 on the real number line. Basically, it divides the intervals between 0 and 1 into a number of smaller intervals corresponding to the probabilities of the message's symbols. Then the first input symbol selects an interval, which is further divided into smaller intervals. The next input symbol selects one of these intervals, and the procedure is repeated. As a result, the selected interval narrows with every symbol, and in the end, any number inside the final interval can be used to represent the message. That is to say, each bit in the output code refines the precision of the value of the input code in the interval.

A3-B3 Compressed Sensing
Compressed Sensing (CS), also known as compressed sampling technique for compressible and/or sparse signals that project a high-dimensional data in low-dimensional space using random measurement matrix [18]. CS theory enables to address the limitations of Nyquist sampling [19] theory and can perform data acquisition and compression simultaneously. This strength of the CS theory enables to detect anomalies in big data streams. The usual achievable compression ratio is from 1:3 to 1:5.

A4 Grammar Based
Grammar-based compression was proposed in 2000 [20]. The basic task of grammar-based codes is constructing a context-free grammar deriving a single string. The class of grammar-based coding are gaining popularity because they can compress highly repetitive input extremely effectively, for instance, a biological data collection of the same or closely related species, a huge versioned document collection, internet archival, etc. For instance, Sequitur (or Nevill-Manning algorithm) is a recursive algorithm developed by Craig Nevill-Manning and Ian H. Witten in 1996 [21],[22] that infers a hierarchical structure (context-free grammar) from a sequence of discrete symbols. The algorithm operates in linear space and time. Although it pre-dates grammar-based compression, LZ78 and LZW (but not LZ77) can be interpreted as a kind of grammar [23].

1.1.2 B-Signal & Image
1 Run Length Encoding (RLE)
2 Predictive Codimg
3. Compressed Sensing
4 Multiresolution Coding

B1 See A1.

B2 Predictive Coding
Linear predictive coding (LPC) is a tool used mostly in audio signal processing and speech processing for representing the spectral envelope of a digital signal of speech in compressed form, using the information of a linear predictive model [24]. It is one of the most powerful speech analysis techniques, and one of the most useful methods for encoding good quality speech at a low bit rate and provides extremely accurate estimates of speech parameters. For images, it predicts the value of each pixel by using the values of its neighboring pixels. Therefore, every pixel is encoded with a prediction error rather than its original value. Typically, the errors are much smaller compared with the original value so that fewer bits are required to store them. As an example, DPCM (differential pulse code modulation) is a predictive coding underlying the base for lossless JPEG compression [25]. A variation of the lossless predictive coding is the adaptive prediction that splits the image into blocks and computes the prediction coefficients independently for each block to achieve high prediction performance. It can also be combined with other methods to get a hybrid coding algorithm with higher performance [26].

B3 See A3.

B4 Multiresolution Coding
An example of multiresolution coding scheme based on sub-samplings is represented by HINT (hierarchical interpolation) [27]. It starts with a low-resolution version of the original image, and interpolates the pixel values to successively generate higher resolutions. The errors between the interpolation values and the real values are stored, along with the initial low-resolution image. Compression is achieved since both the low-resolution image and the error values can be stored with fewer bits than the original image.

1.2 Advanced Requirements

Some image compression schemes have been adopted by the DICOM standard [28], among which JPEG 2000 [29]. It is based on a reversible integer wavelet transform and it is considered to have the best performance and a few good properties. JPEG 2000 provides both lossy and lossless compressions for images with bit depth no more than 16 bit per channel, and allows progressive transmission with SNR (Signal to Noise Ratio) and spatial scalability. In particular, it supports ROI (Region Of Interest) coding, so that image regions with diagnostic importance can have much higher quality than the other parts.

The codestream obtained after compression of an image with JPEG 2000 is downscalable in nature, meaning that it can be decoded in a number of ways; for instance, by truncating the codestream at any point, one may obtain a representation of the image at a lower resolution, or signal-to-noise ratio. However, as a consequence of this flexibility, JPEG 2000 requires encoders/decoders that are complex and computationally demanding. JPEG 2000 has been published as an ISO standard, ISO/IEC 15444. According to Wikipedia, as of 2013, JPEG 2000 is not widely supported in web browsers, and hence is not generally used on the Internet [30]. In general, traditional lossless compression techniques can be mapped to three main reference areas according to their compression underlining principle: a) Entropy encoding; b) Dictionary; c) Others. The very best modern lossless compressors use probabilistic models, such as prediction by partial matching.

2 Contemporary Limitations

As a matter of fact, we can say that almost all contemporary data compression techniques are still based on binary code uncertainty probabilistic evaluation, by treating messages to be encoded as a sequence of independent and identically distributed random variables, according to the probabilistic approach of the father of probabilistic communication theory [31]. Their major points of weakness for contemporary data compression techniques to be used for high demanding lossless applications are:

1) context and data type compression effectiveness and efficiency strong dependence for algorithm optimization;
2) centralized data size compression/decompression speed and space limitations;
3) fixed bit depth for image com/decom processing or just downgrading.

Third point is especially inadequate for discrete tomography and advanced biomedical imaging applications where high data reliability is required, at different bit depth output representation scenarios (for instance, a network with many different output devices with different output bit depth each).

Our main interest is most focused on convenient zero-knowledge universal lossless comp/decomp algorithm for advanced applications as discrete tomography, computed tomography and medical images with true "Arbitrary Bit Depth" (ABD) resolution [5],[6]. No contemporary lossless comp/decomp algorithm is able to satisfy this kind of specification yet. Furthermore it would be a plus to use traditional BD (Bit Depth) settings of images, more efficiently, so that only a small part of the coded bit-stream is needed at the decoder to lossless regenerate and display the original image, with recommended BD parameters for specific need.

3 True Lossless Compression

Human biological transducers, by which we acquire information on the outer world interacting with it, are intrinsically discrete. This means that our perception of continuous shapes is just an illusion created by our mind. From this ground we can infer that an illusion of continuity can be achieved by discrete optimized support, without even noticing any difference, maintaining thus a maximum representation coverage property.

In general, optimization problems can be divided into two large categories depending on whether the variables are continuous or discrete. In solving a classical linear optimization problem (continuous), one can exploit the fact that, due to the convexity of the feasible region, any locally optimal solution is a global optimum.

An optimization problem with discrete variables is known as a combinatorial optimization problem. In a combinatorial optimization problem, we are looking for an object such as an integer, permutation or graph from a finite (or possibly countable infinite) set [32]. In many such problems, exhaustive search is not feasible. It operates on the domain of those optimization problems, in which the set of feasible solutions is discrete or can be reduced to discrete, and in which the goal is to find the best solution. In finding global optima for integer optimization problems, one is required to prove that a particular solution dominates all others by arguments, other than the calculus-based approaches of convex optimization.

While in linear optimization problems, the feasible region is convex, the feasible regions of classical integer optimization problems consists of either a discrete set of points or, in the case of general MILP (Mixed Integer Optimization Problem), a set of disjoint convex polyhedra [33]. Some common problems involving combinatorial optimization are the Traveling Salesman Problem (TSP) and the minimum spanning tree problem. TSP is NP-hard problem in combinatorial optimization, notoriously, and P versus NP problem is a major unsolved problem in computer science [34]. Nevertheless, when a discrete optimized solution is obtained, the discrete approach reveals to be, in this sense, highly convenient because it strongly decreases the computational cost and the complexity of the system for representation modelling. According to computational information conservation theory (*CICT)* [35],[36], in Arithmetic the structure of closure spaces (across an Outer-Inner Universe boundary) is self-defined by Natural Numbers Reciprocal Space (RS) representation [37],[38]. By this way, Natural Number can be thought as both structured object and symbol at the same time.

3.1 Linear Arithmetic Closure

As a simple example, let us consider a generic fraction N/D, where N and D $\in$ ***Z***, and D = $\bar{0}7.\bar{0}$, L an integer counter, N the dividend, Q the quotients and R the remainders of the long hand division. In traditional arithmetic long division algorithm (the one you learn to divide at elementary school), usual dominant result (quotient, Q) is important, and minority components (remainders, R) are always discarded. We can write the L, N, Q and R sequences as from Table 1.

Table 1. L, **N**, **Q**, R SEQUENCES.

L	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15…
N	10	20	30	40	50	60	70	80	90	100	110	120	130	140	150…
Q	1	2	4	5	7	8	10	11	12	14	15	17	18	20	21…
R	3	6	2	5	1	4	0	3	6	2	5	1	4	0	3…

for L = 1, 2, 3, …

In this specific case, we see that the remainder **R** sequence repeats itself after 7 remainders with R(L) given by:

R(1+7n) = R1 = 3, R(2+7n) = R2 = 6, R(3+7n) = R3 = 2, R(4+7n) = R4 = 5, R(5+7n) = R5 = 1, R(6+7n) = R6 = 4, R(7+7n) = R7 = 0 for n = 0,1,2,3,…

Accordingly, we can rearrange the quotients with respect to their cyclic remainder value respectively, obtaining the **R1**, **R2**, **R3**, **R4**, **R5**, **R6**, **R7** sequences as from Table 2.

Table 2. R1, R2, R3, R4, R5. R6, R7 SEQUENCES.

R1	Q1,	Q8,	Q15,	Q22,	Q29,	Q36,	Q43,	Q50,…
R2	Q2,	Q9,	Q16,	Q23,	Q30,	Q37,	Q44,	Q51,…
R3	Q3,	Q10,	Q17,	Q24,	Q31,	Q38,	Q45,	Q52,…
R4	Q4,	Q11,	Q18,	Q25,	Q32,	Q39,	Q46,	Q53,…
R5	Q5,	Q12,	Q19,	Q26,	Q33,	Q40,	Q47,	Q54,…
R6	Q6,	Q13,	Q20,	Q27,	Q34,	Q41,	Q48,	Q55,…
R7	Q7,	Q14,	Q21,	Q28,	Q35,	Q42,	Q49,	Q56,…

In a more compact modular format, we can write:
R(L) = 3 L mod(07) and
D Q(L) = N – R(L), where N = 10 L.
We can interpret the remainders R(L) as the linear (unfolded) arithmetic closure to N with respect to D Q(L).

3.2 Exponential Rational Closure

For sake of simplicity, at elementary level, let us consider fraction 1/D, where D in ***Z***, or Egyptian fraction, with no loss of generality for common fraction (common fraction is given by Egyptian fraction multiplied by N $\in$ ***Z***, where N is the Numerator) is considered a simple integer division. In traditional rational representation, rational proper quotient is represented by infinite repetition of a basic digit cycle, called "reptend" (the repeating decimal part) [39]. The first repetition of basic digit cycle corresponds to the first full scale interval where number information can be conserved; *CICT* calls it "Representation Fundamental Domain" (RFD) [40]. In general, D, the denominator of the considered OSR (Outer Symbolic Representation) is given by a finite decimal word of length W_D digits. From IOR (Inner OpeRational Representation) X, the related RFD_L can be obtained, by a word length of L_X digits. Elementary number theory considerations give us the usual worst case word length L_X for RFD_L, with no loss of information, by:

$$L_X = D - 1 \qquad (1)$$

digits, if and only if 10 is a primitive root modulo D. Otherwise Lx is a factor of (D-1). If the period of the repeating decimal of 1/D for prime D is equal (D−1) then the repeating decimal part is called "cyclic number" and D can be referred as "primitive number" or "solid number" (SN) in *CICT*, or "full reptend prime" elsewhere [41]. Thus a SN is necessarily prime. It is a sufficient qualification, only. Conversely a prime number may not be a SN. So, the usual worst case word length L_X for X, given by eq.(1), can get RFD_L with no loss of information, related to D, just in case D is SN. With no loss of generality, let us consider, the first Integer to show SN property manifestly, that is number "$\overline{0}7.\overline{0}$". In this case D = 7, so that worst case length analysis gives L_X = D–1 = 6 digits.

By realizing that the remainder R_1 is the fixed multiplicative ratio of a formal power series, the computation from generator 3^n (mod7) for n = 1, 2, 3,...., till its exponential closure, gives the sequence the "Fundamental Cyclic Remainder Sequence" (FCRS):

$$R_1 = 3, R_2 = 2, R_3 = 6, R_4 = 4, R_5 = 5, R_6 = 1, \qquad (2)$$

from which the "Fundamental Cyclic Quotient Sequence" (FCQS) can be readily regenerated by 7 * R_n (mod10):

$$Q_1 = 1, Q_2 = 4, Q_3 = 2, Q_4 = 8, Q_5 = 5, Q_6 = 7. \qquad (3)$$

So, quotient and remainder information can always be regenerated anew by remainder information only, but not vice-versa [40].

Therefore, 7 is just a SN and its full information content is usually captured by a six-digit word length RFD_6, for Q_6 and the same for R_6, and the full-information content of long division 1/7 would be stored into two decimal coupled words <Q, R> having length M_{QR} = (6 + 6) digits in total, for exact arbitrary precision computation. As a matter of fact, for all rational sequences, the Remainder R_L, at any computation evolutive stage L_X (accuracy), is the fixed multiplicative ratio of a formal power series associated to the optimized decimal representations of their elementary generators [40]. Thus 1/7 associated information content can be lossless compressed down to minimal M_{QR} = (1 + 1) digits in total, in this specific case. As a matter of fact, any word couple <Q_L, R_L> can be thought to be equivalent to and can represent a different real measurement instrument class, defined by RFD word length L_X. It is easy to see that, in general for SN of order 1 (SN_1 for short), the greater D, the lengthier L_X, and higher the achievable compression ratio E_w.

In this simple case E_w = (2*6)/2 = 6:1, for number 7. By this way, finite length coupled words can provide both finite arbitrary computational precision and relative precision exact error computability in a finite amount of time. By remainder knowledge, it is always possible to regenerate exact quotient and new remainder information at any arbitrary accuracy, with full information conservation. Thanks to the above arithmetic properties, the division algorithm can become free from trial and error like in finite segment *p*-adic representation system, but with no usually associated coding burden [41]. According to our humble knowledge, this is the first time that these arithmetic properties allow this kind of numeric awareness for arithmetic computational system.

All the other members of the 1/7 RFD "Word Family Group" can be derived immediately, by cyclic permutation only, obtaining our result of Fig. 1. So, the final SN_1 Family Group overall information compression ratio is given by E_f = (2 * 6 *6)/2 = 36:1. It does not take a large leap in imagination to suspect that the next larger prime number *p* to show SN property might be a good candidate for exhibiting cyclic-number property on an even larger scale. Unfortunately there does not seem to be any simple rule dictating which prime number *p* will have SN_1 property and which will not, so that one just has to check each prime out by long division to see. It turns out that the next higher prime number with the desired SN_1 property is *p* = D = 17 where T_p = 16 digits. A search for still larger prime numbers with the same cyclic properties reveals that they are not at all rare. In fact no less than seven more prime numbers smaller than 100 generate cyclic numbers of order one or SN_1. They are: 19, 23, 29, 47, 59, 61 and 97.

Till now we discussed a peculiar number property focusing our attention, for simplicity of presentation, on SN_1 only, but it does not imply that a number has to be a SN_1 in order to generate intriguing cycles. For instance, Family Group 1/13 contains the smallest cyclic-numbers of order two or SN_2. What about more second order cyclics? The basic ones are also generated by a subset of prime fractions 1/*p*. The other prime numbers smaller than 100 which generate second order cyclics are 31, 43, 67, 71, 83 and 89. In the same manner one can now go on to define, and find, cyclic numbers of order three, four, five, and higher and higher, according to his own computational needs. The smallest prime number producing the Family Group of third order or SN_3 cyclic-number is 103. In this case, T_p = 34 digit string (so we have 3* 34 = 102, instead of the unique 102 digit string a first order cyclic would have).

1/7	0.	$Q_1=1$	$Q_2=4$	$Q_3=2$	$Q_4=8$	$Q_5=5$	$Q_6=7$
		$R_1=3$	$R_2=2$	$R_3=6$	$R_4=4$	$R_5=5$	$R_6=1$
2/7	0.	$Q_1=2$	$Q_2=8$	$Q_3=5$	$Q_4=7$	$Q_5=1$	$Q_6=4$
		$R_1=6$	$R_2=4$	$R_3=5$	$R_4=1$	$R_5=3$	$R_6=2$
3/7	0.	$Q_1=4$	$Q_2=2$	$Q_3=8$	$Q_4=5$	$Q_5=7$	$Q_6=1$
		$R_1=2$	$R_2=6$	$R_3=4$	$R_4=5$	$R_5=1$	$R_6=3$
4/7	0.	$Q_1=5$	$Q_2=7$	$Q_3=1$	$Q_4=4$	$Q_5=2$	$Q_6=8$
		$R_1=5$	$R_2=1$	$R_3=3$	$R_4=2$	$R_5=6$	$R_6=4$
5/7	0.	$Q_1=7$	$Q_2=1$	$Q_3=4$	$Q_4=2$	$Q_5=8$	$Q_6=5$
		$R_1=1$	$R_2=3$	$R_3=2$	$R_4=6$	$R_5=4$	$R_6=5$
6/7	0.	$Q_1=8$	$Q_2=5$	$Q_3=7$	$Q_4=1$	$Q_5=4$	$Q_6=2$
		$R_1=4$	$R_2=5$	$R_3=1$	$R_4=3$	$R_5=2$	$R_6=6$
7/7	0.	$Q_1=9$	$Q_2=9$	$Q_3=9$	$Q_4=9$	$Q_5=9$	$Q_6=9$
		$R_1=7$	$R_2=7$	$R_3=7$	$R_4=7$	$R_5=7$	$R_6=7$

Fig. 1. Q_n and R_n values for each component of SN_1 Word Family Group 1/7 [40].

The seven smallest primes generating cycles of order four through ten are respectively 53, 11, 79, 211, 41, 73, and 281, but you must reach prime number 353 to get the first prime generating cycle of order 11 or SN_{11}. All the prime numbers less than 100 have now been covered with the exception of p = 37, with T_p = 3 digits. Therefore 37 is a cyclic number of order 12 or SN_{12}. Multiplication composition of prime numbers can generate an entire universe of new number-cycles. So any modular group can be better represented by generators and relations. One of the earliest presentations of a group by generators and relations was given by the Irish mathematician William Rowan Hamilton in 1856, in his Icosian Calculus, a presentation of the icosahedral group [42], [43]. The first systematic study was given by German mathematician Walther Franz Anton von Dyck [44], student of German mathematician Christian Felix Klein, in the early 1880s, laying the foundations for combinatorial group theory [45]. Every group has a presentation, and in fact many different presentations; a presentation is often the most compact way of describing the structure of the group. In abstract algebra, the "fundamental theorem of cyclic groups" states that every subgroup of a cyclic group *G* is cyclic. Moreover, the order "k" of any subgroup of a cyclic group *G* of order *n* is a divisor of *n*, and for each positive divisor "k" of *n*, the group *G* has exactly one subgroup of order "k". This is just the first step to start an intriguing voyage from the concept of "presentation of a group" to the concept of "representation theory" for combinatorial modular group theory [46]. Traditional rational number system ***Q*** can be regarded as a highly sophisticated open logic, powerful and flexible formal language, with self-defining consistent words and rules, starting from elementary generators and relations [40].

4 Application Example

The rich operative scenario offered by combinatorial modular group theory is full of articulated solutions to information processing problems.

Table 3. PICTURE BENCHMARK DATABASE LIST [27].

N.	NAME	PIXELS	SIZE (MB)	BPP
01	artificial	3072x2048	36.80	24
02	big_building	7216x5412	223.36	24
03	big_tree	6088x4550	158.50	24
04	cathedral	2000x3008	34.42	24
05	fireworks	3136x2352	42.21	24
06	flower_foveon	2268x1512	19.62	24
07	hdr	3072x2048	36.80	24
08	leaves_iso_1600	3008x2000	34.42	24
09	leaves_iso_200	3008x2000	34.42	24
10	nightshot_iso_100	3136x2352	42.29	24
11	nightshot_iso_1600	3136x2352	42.29	24
12	spider_web	4256x2848	69.36	24

For instance, Word Family Group SN_1, discussed in the previous section, shows peculiar, cyclical properties that can be conveniently used to get interesting image lossless comp/decomp algorithm. Please, note from Fig.1 that remainder $R_n = a$ = 1,2, …, p -1, can be thought as pointer to Q_n+1 to get the beginning of a/D quotient cyclical string immediately, with no computation at all, but the initial one for SN_1 Word Family p. Combinatorial optimization is achieved by finding the best $SN_1 = p$ which allow to minimize a desired constraint within a specified interval for each image. Then, a/p will be best in that no other rational in that specified interval will have a smaller numerator or a smaller denominator. To check our idea, we used an Internet public picture database for 24 bpp color images [47] as reported by Table 3 and Fig.2. Twelve original colour pictures with different dynamic range and different overall content were considered. They are lossless compressed from 24 to 16 bpp by JPEG-LS (Lossless JPEG), JPEG 2000, HD Photo (JPEG XR) algorithms and by ours that is called UC (Universal Compression).

4.1 Lossless Compression Details

Lossless JPEG was developed as a late addition to JPEG in 1993, using a completely different technique from the lossy JPEG standard developed previously. It uses a predictive scheme based on the three nearest (causal) neighbors (upper, left, and upper-left), and entropy coding is used on the prediction error.

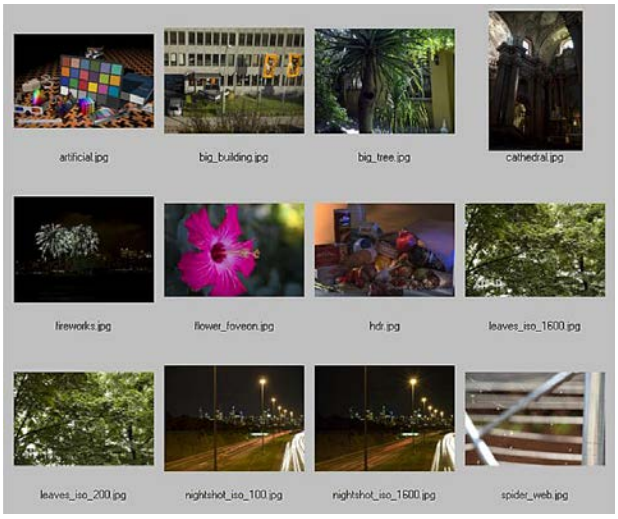

Fig. 2. Pictures in Benchmark Database as from Table 1, for our application example [47].

The standard Independent JPEG Group libraries cannot encode or decode it, but Ken Murchison of Oceana Matrix Ltd. wrote a patch that extends the IJG library to handle Lossless JPEG. It has some popularity in medical imaging, and is used in DNG (Adobe) and some digital cameras to compress raw images, but otherwise was never widely adopted. It might be used as an umbrella term to refer to all lossless compression schemes developed by the Joint Photographic Expert group. They include JPEG 2000 and JPEG-LS.

JPEG-LS is a lossless/near-lossless compression standard for continuous-tone images. Its official designation is ISO-14495-1/ITU-T.87. It is a simple and efficient baseline algorithm which consists of two independent and distinct stages called modeling and encoding. It was developed with the aim of providing a low-complexity lossless and near-lossless image compression standard that could offer better compression efficiency than JPEG. At the time, the Huffman coding-based JPEG lossless standard and other standards were limited in their compression performance. Total decorrelation cannot be achieved by first order entropy of the prediction residuals employed by these inferior standards. JPEG-LS, on the other hand, can obtain good decorrelation. Part 1 of this standard was finalized in 1999. Part 2, released in 2003, introduced extensions such as arithmetic coding. The core of JPEG-LS is based on the LOCO-I algorithm [48] that relies on prediction, residual modeling and context-based coding of the residuals. Most of the low complexity of this technique comes from the assumption that prediction residuals follow a two-sided geometric distribution (also called a discrete Laplace distribution) and from the use of Golomb-like codes, which are known to be approximately optimal for geometric distributions. Besides lossless compression, it also provides a lossy mode ("near-lossless") where the maximum absolute error can be controlled by the encoder. Compression for JPEG-LS is generally faster than JPEG 2000 and much better than the original lossless JPEG standard. Here JPEG-LS is used for historical reason comparison, JPEG 2000 provides lossless compressions for images with bit depth no more than 16 bit per channel, and allows progressive transmission with signal-to-noise ratio (SNR) and spatial scalability. In particular, it supports ROI (Region Of Interest) coding, so that image regions with diagnostic importance can have much higher quality than the other parts. The codestream obtained after compression of an image with JPEG 2000 is downscalable in nature, meaning that it can be decoded in a number of ways; for instance, by truncating the codestream at any point, one may obtain a representation of the image at a lower resolution, or SNR. However, as a consequence of this flexibility, JPEG 2000 requires encoders/decoders that are complex and computationally demanding.

HD Photo slgorithm is based on technology originally developed and patented by Microsoft. It supports deep color images with 48-bit RGB, both lossy and lossless compression, and is the preferred image format for Ecma-388 Open XML Paper Specification documents. HD Photo offers several major key improvements over JPEG, including better compression, lossless compression, tile structured support, more color accuracy, transparency map and metadata support. HD Photo is conceptually very similar to JPEG: the source image is optionally converted to a luma-chroma colorspace, the chroma planes are optionally subsampled, each plane is divided into fixed-size blocks, the blocks are transformed into the frequency domain, and the frequency coefficients are quantized and entropy coded. In July 2007, the Joint Photographic Experts Group and Microsoft announced HD Photo to be under consideration to become a JPEG standard known as JPEG XR. On 16 March 2009, JPEG XR was given final approval as ITU-T Recommendation T.832 and starting in April 2009, it became available from the ITU-T in "pre-published" form. On 19 June 2009, it passed an ISO/IEC Final Draft International Standard (FDIS) ballot, resulting in final approval as International Standard ISO/IEC 29199-2. The ITU-T updated its publication with a corrigendum approved in December 2009, and ISO/IEC issued a new edition with similar corrections on 30 September 2010. In 2010, after completion of the image coding specification, the ITU-T and ISO/IEC also published a motion format specification (ITU-T T.833|ISO/IEC 29199-3), a conformance test set (ITU-T T.834|ISO/IEC 29199-4), and reference software (ITU-T T.835|ISO/IEC 29199-5) for JPEG XR. In 2011, they published a technical report describing the workflow architecture for the use of JPEG XR images in applications (ITU-T T.Sup2|ISO/IEC TR 29199-1). In April 2013, Microsoft released an open source JPEG XR library under the BSD licence. As of August 2014, there were still no cameras that shoot photos in the Jpeg XR (.JXR) format.

UC provides lossless compressions for images with arbitrary bit depth (ABD) per channel, and allows progressive transmission with SNR and spatial scalability, at no extra cost. Specifically, it supports ROI (Region Of Interest) coding, so that image regions with diagnostic importance can have much higher quality than the other parts. The codestream obtained after compression of an image with UC is downscalable and upscalable in nature, meaning that it can be decoded in a number of ways; for instance, by truncating the codestream at any point, one may obtain a representation of the image at a lower resolution, or signal-to-noise ratio. Different from all other algorithms, this time, if you have local computational resources, you can regenerate the original information arbitrarily to higher BD. As a consequence of this flexibility, you do have to pay no extra cost!

4.2 Results and Discussion

Compression results for 24 to 16 bpp lossless compression of the twelve database pictures reported in Table 3, with different content and different extended dynamic each, show that, on average, UC takes about 17% more space than JPEG 2000. Nevertheless UC shows compression speed about 17% faster than JPEG 2000 on average.

To achieve an overall evaluation and a fair comparison between so different algorithms, it involves determining a function that relates the length of an algorithm's input to the number of steps it takes (its time complexity) or the number of storage locations it uses (its space complexity) or both of them. An algorithm is said to be efficient when this function's values are small. Since different inputs of the same length may cause the algorithm to have different behavior, the function describing its performance is usually an upper bound on the actual performance, determined from the worst case inputs to the algorithm. For this reason, we define the patiotemporal, normalized key performance index (KPI) given by compressed bit-per-pixel (bpp) times compression time (s, in second) for each picture, where:

$$KPI = bpp \times s/MAX(bpp \times s) \quad . \qquad (4)$$

Our first raw results, with no further algorithm refinement out of cyclic family computation only, as described in Section 3.2, show that overall UC KPI compare quite well to HD Photo and it is better than traditional JPEG 2000, with no further algorithmic development effort or computational load.

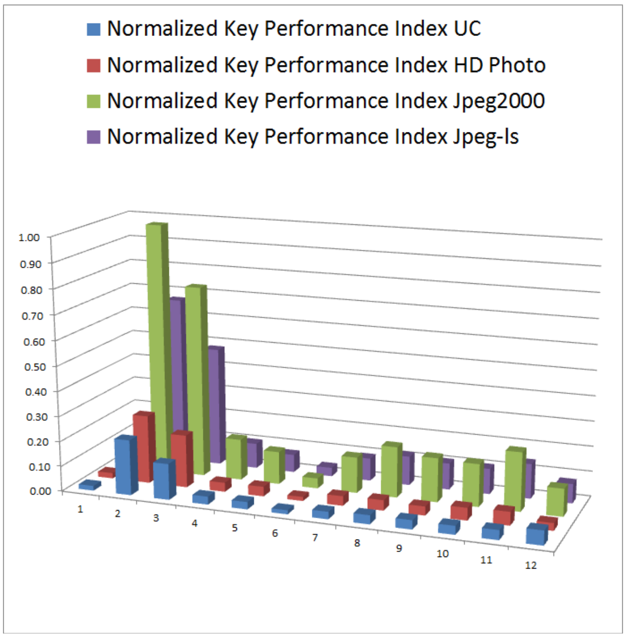

Fig. 3. Normalized KPI 24 to 16 bpp lossless compression result for JPEG-LS, JPEG 2000, HD Photo and UC algorithms (from back to front respectively): ordinate in KPI units, abscissa number ordering refers to picture list as from Table 1.

The only image where UC KPI is greater than HD Photo KPI is picture number 12, the “spider_web” picture shown in Fig.4. As you can see from that picture, different from all other picture in our Benchmark Database, there are a lot of linear, finer details with delicate superpositions embedded in it and HD Photo algorithm can take advantage from them quite easily.

UC is a zero-knowledge universal lossless compression algorithm that does not use linear a priori knowledge to achieve better result. Therefore, for general purpose application raw UC on average can perform better than HD Photo, but for pictures full of superposed linear, finer details it cannot compete with HD Photo.

Fig. 4. Spider_web picture as n.12 test picture in our Picture Benchmark Database (see Table 3 for image parameters).

Nevertheless, raw UC algorithm parameters have not yet been quite tuned for overall optimum performance, so there is still room for further improvement. UC exploits arbitrary bit depth (ABD) per channel and full information conservation and regeneration by design. UC method differs from the existing ones in three main features. First, it is not a traditional extended bit depth encoding/decoding and does not use Gray-Golomb coding to support progressive transmission as in VOI (Value Of Interest) approaches [49].

Second, UC scheme can utilize dynamic BD settings for original images to support dynamic BD scalability, which allows pixels to be progressively reconstructed in the order according to their display importance. Pixel reconstruction can be centralized or decentralized, local or global. Third, we can consider end-user display computational resource availability, for end-point information regeneration and network bandwidth minimization. Different from all other algorithms, not only a downscaling, but even an upscaling is possible by UC information conservation and regeneration. For images with extended bit-depth, the pixel values can be displayed after BD mapping to fit the display bit depth. Thus only pixels with interested values can be visually important, under certain predefined BD parameters.

5 Conclusion

We have presented results on an universal zero-knowledge combinatorial lossless compression scheme called UC, for extended dynamic range images, with arbitrary bit depth, based on information conservation and regeneration according to *CICT* by design. UC has an interesting, peculiar and operational property: the larger the image size the higher the achievable compression ratio. In this first operational implementation, its raw overall zero-knowledge lossless compression performance compare quite well to standard ones, but offering true ABD and achieving full image information conservation and regeneration, at no extra cost, without complex and computationally demanding encoders/decoders.

References:

[1] N., Reims, T., Schoen, M., Boehnel, F., Sukowski, M., Firsching, Strategies for efficient scanning and reconstruction methods on very large objects with high-energy x-ray computed tomography, *Proc. SPIE* 9212, Developments in X-Ray Tomography IX, 921209, September 12, 2014; doi: 10.1117/12.

[2] K.J., Batenburg, J., Sijbers, DART: a practical reconstruction algorithm for discrete tomography, *IEEE Transactions on Image Processing*, Vol. 20, No. 9, 2011, pp. 2542-2553.

[3] K.J., Batenburg, S., Bals, J., Sijbers, C., Kuebel, P.A., Midgley, J.C., Hernandez, U., Kaiser, .R., Encina, E.A., Coronado, G., Van Tendeloo, 3D imaging of nanomaterials by discrete tomography, *Ultramicroscopy*, Vol. 109, No. 6, 2009, pp. 730-740.

[4] M.H., Kryder, S.K., Chang, After Hard Drives – What Comes Next?, *IEEE Transactions on Magnetics*, Vol. 45**,** No. 10, October 2009, pp. 3406-3413.

[5] D.A., Koff, H., Shulman, An overview of digital compression of medical images: Can we use lossy image compression in radiology?, *Canadian Association of Radiologists Journal*, Vol. 57, 2006, pp. 211–217.

[6] M.J., Zukoski, T., Boult, T., Iyriboz, A novel approach to medical image compression,

International Journal of Bioinformatic Research and Applications, 2006, pp. 89–103.

[7] T.J., Kim, K.H., Lee, B., Kim, E.J. Chun, V. Baipal, Y.H. Kim, S. Hahn, K.W. Lee, Regional variance of visually lossless threshold in compressed chest CT images: lung versus mediastinum and chest wall, *European Journal of Radiology*, Vol.69, No. 3, 2009, pp. 483–488.

[8] A.H., Robinson, C., Cherry, Results of a prototype television bandwidth compression scheme, *Proceedings of the IEEE*, Vol. 55, No. 3, 1967, pp. 356–364.

[9] Recommendation T.45 (02/00), Run-length colour encoding, International Telecommunication Union, 2000. Available at:<URL= https://www.itu.int/itudoc/itu-t/84601.pdf>.

[10] D., Huffman, A Method for the Construction of Minimum-Redundancy Codes, *Proceedings of the IRE*, Vol. 40, No. 9, 1952, pp. 1098–1101.

[11] Modified Huffman coding, Available at:<URL= https://web.archive.org/web/20020628195336/http://www.netnam.vn/unescocourse/computervision/104.htm>.

[12] J.S., Vitter, Design and Analysis of Dynamic Huffman Codes, *Journal of the ACM*, Vol. 34, No.4, October 1987, pp 825–845.

[13] J., Ziv, A., Lempel, A Universal Algorithm for Sequential Data Compression, *IEEE Transactions on Information Theory*, Vol. 23, No. 3, May 1977, pp. 337–343.

[14] J., Ziv, A., Lempel, Compression of Individual Sequences via Variable-Rate Coding, IEEE *Transactions on Information Theory*, Vol. 24, No. 5, September 1978, pp. 530–536.

[15] T., Welch, A Technique for High-Performance Data Compression, *Computer*, Vol. 17, No. 6, 1984, pp. 8–19.

[16] J.J., Rissanen, Generalized Kraft Inequality and Arithmetic Coding, *IBM Journal of Research and Development*, Vol. 20, No. 3, May 1976, pp. 198–203.

[17] J.J., Rissanen, G.G., Langdon, Jr., Arithmetic coding, *IBM Journal of Research and Development*, Vol. 23, No. 2, March 1979, pp. 149–162

[18] E.J., Candès, J.K., Romberg, T., Tao, Stable signal recovery from incomplete and inaccurate measurements, *Communications on Pure and Applied Mathematics*, Vol. 59, No. 8, 2006, pp. 1207–1223

[19] ., Nyquist, Certain topics in telegraph transmission theory, *Transactions of AIEE*, Vol.47, April 1928, pp. 617–644.

[20] J.C., Kieffer, E-H., Yang, Grammar-based codes: A new class of universal lossless source codes, *IEEE Trans. Inform. Theory*, Vol. 46, No. 3, 2000, 737–754.

[21] C.G., Nevill-Manning, Inferring sequential structure, Ph.D. thesis, Department of Computer Science, University of Waikato, New Zealand.

[22] C.G., Nevill-Manning, .H., Witten, Identifying Hierarchical Structure in Sequences: A linear-time algorithm, 1997, arXiv:cs/9709102. Available at:<URL= https://arxiv.org/abs/cs/9709102>.

[23] J., Kieffer, E-H., Yang, Structured grammar-based codes for universal lossless data compression, *Communications in Information Systems*, Vol. 2, No.1, 2002, pp.29-52. Available at:<URL= http://www.ims.cuhk.edu.hk/~cis/2002.1/KIEFFER.pdf>.

[24] D., Li, D., O'Shaughnessy, *Speech processing: a dynamic and optimization-oriented approach*, Marcel Dekker, 2003, pp. 41–48.

[25] C., Chapin Cutler, Differential Quantization of Communication Signals, *U.S. patent 2605361*, filed June 29, 1950, issued July 29, 1952. Available at:<URL= http://www.google.com/patents/US2605361>.

[26] H., Al –Mahmood, Z., Al-Rubaye, Lossless Image Compression based on Predictive Coding and Bit Plane Slicing, *International Journal of Computer Applications (0975 – 8887)*, Vol. 93, No 1, May 2014, pp.1-6.

[27] K., Chen, T.V., Ramabadran, An improved hierarchical interpolation (HINT) method for the reversible compression of grayscale images, Proceedings. Data Compression Conference, Snowbird, UT, USA, 1991, pp. 436-439.

[28] The DICOM standard, (2009). Available at:<URL= http://dicom.nema.org/dicom>.

[29] ISO/IEC 15444 (2007). Available at:<URL= https://www.iso.org/standard/41570.html>.

[30] Wikipedia, Jpeg2000 (2017). Available at:<URL= https://en.wikipedia.org/wiki/JPEG_2000>.

[31] C.E., Shannon, Bell Systems Technical Journal, Vol. 27, No.3, July/October 1948, 379.

[32] A., Schrijver, 2017, A Course in Combinatorial Optimization. Available at:<URL=

http://homepages.cwi.nl/~lex/files/dict.pdf>.

[33] K.L., Hoffman, T.K., Ralphs, Integer and Combinatorial Optimization, Encyclopedia of Operations Research and Management Science, 2012. Available at:<URL= http://coral.ie.lehigh.edu/~ted/files/papers/ICO-EORMS11.pdf>.

[34] L. Fortnow, The status of the P versus NP problem, *Communications of the ACM*, Vol. 52, No. 9, 2009, p. 78–86.

[35] R.A. Fiorini, Computational Information Conservation Theory: An Introduction, Proceedings of the 8th International Conference on Applied Mathematics, Simulation, Modelling (ASM '14), in N.E. Mastorakis, M. Demiralp, N. Mukhopadhyay, F. Mainardi, Eds., *Mathematics and Computers in Science and Engineering Series*, No.34, NAUN Conferences, WSEAS Press, November 22-24, 2014, Florence, Italy, pp.385-394.

[36] R.A. Fiorini, GSI & CICT as coupled resource for multi-scale system biology and biomedical engineering modeling, *International Journal of Systems Applications, Engineering and Development*, Vol. 9, pp. 93-97 (2015). Available at:<URL= http://www.naun.org/main/UPress/saed/2015/a305994003.pdf>.

[37] R.A. Fiorini, How Random is Your Tomographic Noise? A Number Theoretic Transform (NTT) Approach, *Fundamenta Informaticae*, Vol. 135, Nos 1-2, 2014, pp.135-170.

[38] R.A. Fiorini, G. Laguteta, in V. Mladenov, Ed., *Proceedings of the 19th International Conference on Systems*, Zakynthos Island, Greece, 2015, pp. 648-655.

[39] E.W. Weisstein, (1999-2012). Available at:<URL= http://mathworld.wolfram.com/FullReptendPrime.html>.

[40] R.A. Fiorini, G. Laguteta, Discrete Tomography Data Footprint Reduction by Information Conservation, *Fundamenta Informaticae*, Vol. 125, Nos 3-4, pp. 261-272.

[41] D.M. Young, R.T. Gregory, *A Survey of Numerical Mathematics, vol.I and II*, Addison Wesley, Reading, Mass., USA, 1973.

[42] W.R. Hamilton, Memorandum respecting a new System of Roots of Unity, Philosophical Magazine, 12, 1856, pp. 446.

[43] B. Chandler, W. Magnus, *The History of Combinatorial Group Theory: A Case Study in the History of Ideas, Studies in the History of Mathematics and Physical Sciences*, Springer, 234, 1st ed., December 1, 1982.

[44] W. von Dyck, Gruppentheoretische Studien, Math. Ann., Vol. 20, No. 1, 1882, p.1-45.

[45] J. Stillwell, *Mathematics and its history*, Springer, 2002.

[46] C.W. Curtis, *Pioneers of Representation Theory: Frobenius, Burnside, Schur, and Brauer, History of Mathematics*, American Mathematical Society,(Providence, RI, USA, 2003.

[47] The New Test Images for Image Compression Benchmark (Jan 2008). Available at:<URL= http://www.imagecompression.info/test_images>.

[48] M. J., Weinberger, G. Seroussi, G. Sapiro, The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS, IEEE Trans. Image Processing, Vol. 9, No. 8, Aug. 2000, pp. 1309-1324. Available at:<URL= http://www.labs.hp.com/research/info_theory/loco/> <URL= http://www.labs.hp.com/research/info_theory/loco/HPL-98-193R1.pdf>.

[49] W. Sun, Y. Lu, W. Feng, L. Shipeng, Level Embedded Medical Image Compression Based on Value of Interest, IEEE, ICIP, 2009, pp. 1749-1752. Available at:<URL= http://toc.proceedings.com/07230webtoc.pdf>.