Freeman Chain Code with Digits of Unequal Cost

Sohag Kabir
Department of Computer Science
University of Hull
Hull, UK
Email: s.kabir@hull.ac.uk

Tanzima Azad
Department of CSE
Bangladesh University of Engineering and Technology
Dhaka, Bangladesh
Email: azad.tanzima@dbbl.com.bd

A S M Ashraful Alam
Department of Computer Science
University of Otago
Dunedin, New Zealand
Email: aalam@cs.otago.ac.nz

Abstract—**Chain codes are the most size-efficient lossless compression methods for representing rasterised binary objects and contours. Satisfactory compression ratio, low processing cost and low storage requirements of the decoder make chain code technique interesting for storage and transmission of predefined graphical objects in embedded environments. Each element in the chain is encoded to show the relative angle difference between two adjacent pixels along the boundary of an object. The cost of binary bits representing the codes are considered to be equal. Yet, more efficient encoding is possible by considering and applying technique that treats the binary bits differently considering its requirement of storage space, energy consumption, speed of execution and etc. This paper considers cost of binary digits as unequal and proposes a new representation of the eight-direction Freeman chain code based on a variation of Huffman coding technique, which considers cost of bits as unequal. The evaluation and comparison of the cost efficiency between classical Freeman chain code and the new representation of the chain code is provided. Our experiments yield that the proposed representation of Freeman Chain code reduces overall storage/transmission cost of encoded objects considerably with compared to classical Freeman chain code.**

***Keywords*—Chain Codes, Freeman Chain Code, Image Processing, Huffman Code, Coding Theory**

I. Introduction

The efficiency of encoding technique is very important in representing, recognising, storing, analysing, and transmitting the shape of the objects. Chain codes are widely used in image processing and pattern recognition for representing different rasterised shapes like lines, curves, or region boundaries. The first method for representing the boundaries of digital curves using chain code was introduced by Freeman in 1961 [1]. Many authors have reported different interesting applications using chain codes in computer graphics [2]–[4], image processing [5]–[7], and pattern recognition [8]–[10]. Each element in the chain is encoded to show the relative angle difference between two adjacent pixels along the boundary of digital objects. Freeman chain code is a fixed-length encoding scheme. In the eight-direction version of the Freeman chain code, the directions are represented as numbers from 0 to 7 and the numbers are represented using 3 binary digits as 000, 001, 010, 011, 100, 101, 110, 111 respectively. Therefore the length of each codeword is 3 and the total length of the chain is 3 times the number of total movement in different directions. Though the angle differences between two contiguous elements in a chain are not equally probable, i.e., frequency of movement in different directions are not same, the classical Freeman chain code does not take the frequency of movement in different directions into account while assigning codes. But encoding characters in predefined fixed length code, does not attain an optimum performance, as every character consumes equal number of bits.

There are variable-length encoding schemes like Huffman code [11] that considers the probabilities at which different discrete values are likely to occur and assign variable-length code to symbols based on their occurrence frequencies, and thus represent a message with fewer number of bits. Classical Freeman chain code has been modified by using Huffman code by utilising the compression advantage of the Huffman code in [12]. In both classical and modified Freeman chain code, the cost of bits are considered to be equal. But in real life application, cost of letters are not always equal. For example, Morse Code encodes alphabets as standardized sequences of short and long signals called “dots” and “dashes”. The duration of a dash is three times the duration of a dot. Therefore, if the letter costs are unequal then length and cost of codewords are different. Using compression techniques without considering unequal letter costs may help to reduce the number of bits to represent a message but may not be able to reduce the cost of the message. A number of researchers have proposed data compression techniques that consider unequal letter costs like [13]–[16]. In this paper, we introduces a novel representation of the classical Freeman chain code that uses ‘unequal letter cost’ technique while the basic structure of the Freeman chain code remains unaltered. Our experiments suggest that the technique contributes towards improved performance and it reduces the overall requirements of resources considerably. Our technique reduces cost by 4.57% in the worst case scenario; in the best case it reduces cost by 22.15%, and on an average by 11.4%. A comparison among the cost of Classical Freeman chain code, Freeman chain code using Huffman code technique and the proposed representation using Huffman code with unequal letter costs is evaluated and the result is presented.

The paper is organized as follows: Section 2 presents the background study of the Freeman chain code, unequal letter cost encoding and the Huffman code. The representation of Freeman chain code using classical Huffman code is shown in Section 3. In Section 4, the Freeman chain code is represented using Huffman code with unequal letter cost. Results and discussion are presented in Section 5. Finally, concluding remarks are presented in Section 6.

978-1-4799-6399-7/14/$31.00

II. BACKGROUND STUDY

A. Freeman Chain Code

Freeman chain code is the most widely used chain code technique because of the simplicity. This coding technique encodes an object as a sequence of movements through the neighbouring boundary pixels based on eight-connectivity. The direction of movement in the chain is encoded by using the numbering scheme $\{i|i=0,1,2,\ldots,7\}$ that is denoted as a counter-clockwise angle of $45^\circ \times i$ with respect to the positive $x-$axis, as shown in Fig.1. This scheme is also known as Freeman eight-directional chain code (FEDCC). In Fig.2(b), the eight-directional chain is shown for the shape given in Fig.2(a).The four-directional version of chain code (FFDCC) is also used where direction of movement is encoded using a numbering scheme $\{i|i=0,1,2,3\}$ denoting an angle of $90^\circ \times i$ counter-clockwise with regards to the positive $x-$axis. In classical Freeman chain code, the chances of movement towards different directions are not equiprobable. The probability of angle differences between two contiguous elements in a chain code is shown in Table I as shown in [12].

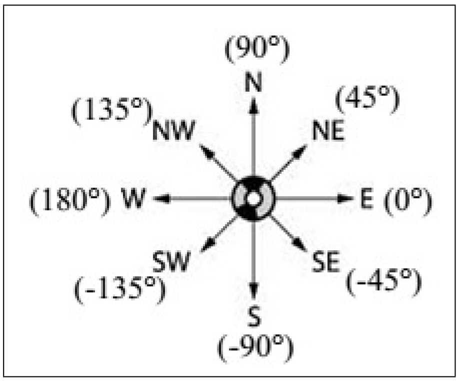
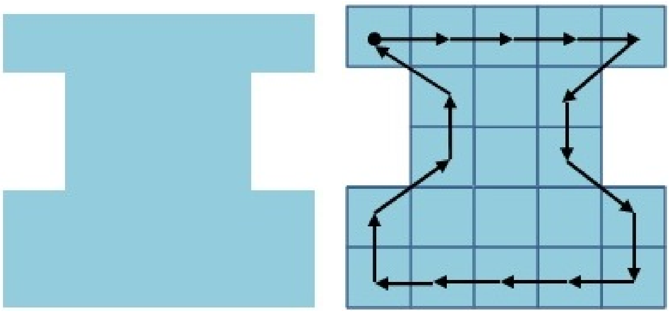


Fig. 1. Eight-Direction Freeman Chain Code



(a) Example Image (b) Chain representation of the image

Fig. 2. Example of Freeman Chain Code

TABLE I. PROBABILITIES OF ANGLE DIFFERENCES BETWEEN CONTIGUOUS SEGMENTS

Angle difference	0°	$\pm 45^\circ$	$\pm 90^\circ$	$\pm 135^\circ$	180°
Probability	0.453	0.488	0.044	0.012	0.003

Angle differences between neighbouring pixels vary from object to object. It is seen from Table I that the probability of the next movement is the highest (0.488) at 45° or -45° and is the lowest (0.003) at 180° from present position. The observation suggests that movement at 45° or -45° is the most frequent and at 180° is the least frequent. But classical Freeman chain code does not consider the changes in frequency of movement during assigning code to the elements of a chain. Classical Freeman chain code assigns three bits of predefined fixed length code to each of the elements in the chain. By considering the frequency of movement a new representation of Freeman chain code in proposed in [12]. However, this is not the only modified representation of Freeman chain code.Bribiesca represented shapes using a derivative of Freeman chain code in 1991 [17]. He also proposed another representation of chain code known as *Vertex Chain Code* in 1999 [18]. Different modifications of Freeman chain codes are deliberately described in [19]–[22] .

B. Unequal bit cost encoding

An important function of a digital communications system is to represent the digitised signal with as fewer bits as possible without losing information. However, in several applications involving image processing and sound representation, certain degree of information loss may be acceptable as it does not invalidate the purpose of those applications. Data must be encoded to meet the purposes like: unambiguous retrieval of information, efficient storage, efficient transmission and etc. Efficiency can be measured in terms of incurred cost, required storage space, consumed power, time spent and likewise. Let a message consist of sequences of characters taken from an alphabet Σ, where $\alpha_1, \alpha_2, \alpha_3 \ldots, \alpha_r$ are the elements that represent the letters in the source Σ. The length of α_i represents its cost or transmission time, i.e., $c_i = length(\alpha_i)$. A codeword w_i is a string of characters in Σ, i.e., $w_i \in \Sigma^+$. If a codeword $w_i = \alpha_{i1}, \alpha_{i2}, \ldots, \alpha_{in}$, then the length or cost of the codeword is the sum of the lengths of its constituent letters:

$$\text{cost}(w_i) = \sum_{j=1}^{n} c_{ij} \tag{1}$$

A scheme of prefix code assigns codes to letters in Σ to form codeword w_i such that none of them is a prefix to another. For example, the codes $\{1, 01, 001, 0001\}$ and $\{000, 001, 011, 111\}$ are prefix-free, whereas the code $\{1, 01, 100\}$ is not, because 1 is a prefix in 100. Prefix-free property is very desirable to be able to uniquely decipher a code. The problem of constructing an optimal prefix code, when all the letters of the alphabet are of equal cost and the probabilities of the words are arbitrarily specified, was solved by Huffman [11]. But the letters in the alphabet may require unequal cost. For example, in Morse code a dash$(-)$ is three times longer than a dot$(\cdot)$ in duration [23]. Morse Code is an example of a variable-length encoding scheme. In Morse code, frequently used letters like E and T have shorter length codes than seldom-used letters like Q and Z , which have longer codes. Using Morse Code, we can treat each dot and dash mark as the equivalent of one binary bit each. The more general case where the costs of the letters as well as the probabilities of the words are arbitrarily specified was treated by Karp [24]. A number of other researchers have focused on uniform sources and developed algorithm for the unequal letter costs encoding [13]–[15], [25]–[27]. Let $p_1, p_2, \ldots, p_n$ be the probabilities with which the source symbols occur in a message and the codewords representing the source symbols are $w_1, w_2, \ldots, w_n$ then the cost of the code W is

$$C(W) = \sum_{i=1}^{n} cost(w_i).p_i \tag{2}$$

The aim of producing an optimal code problem with unequal letter cost is to find a codeword W that consists of n prefix code letters each with minimum cost c_i that produces the overall minimum cost $C(W)$, given that costs $0 < c_1 \leq c_2 \leq c_2 \ldots \leq c_n$, and probabilities $p_1 \geq p_2 \geq \ldots \geq p_n > 0$.

C. Huffman Codes

Technique of producing Huffman codes is an optimal encoding rule that minimises the number of bits needed to represent source data is proposed by D.A. Huffman [11]. It is an entropy encoding algorithm used for lossless data compression. It is an efficient data compression scheme that takes into account the probabilities at which different quantisation levels are likely to occur and results into fewer data bits on the average. For any given set of levels and associated probabilities, there is an optimal encoding rule that minimises the number of bits needed to represent the source. Encoding characters in predefined fixed length code, does not attain an optimum performance, because every character consumes equal number of bits. Huffman code tackles this by generating variable length codes, given a probability usage frequency for a set of symbols. Applications of Huffman code are pervasive throughout computer science. Huffman code can be used effectively where there is a need for a compact code to represent a long series of a relatively small number of distinct bytes. The algorithm to completely perform Huffman encoding and decoding has too many implementation details to describe here, but everything that is required is explained in detail by Amsterdam in [28]. There are many other variants of Huffman codes that compress source data to reduce data size and/or transmission cost. For example, Mannan and Kaykobad introduced block technique in Huffman coding which overcomes the limitation of reading whole message prior to encoding [29]. In classical Huffman coding scheme, the letter costs are considered as equal. The unequal letter cost versions of Huffman codes scheme are proposed in [16], [30]–[32]. In the unequal letter cost version of the classical Huffman code, letters of the alphabet are considered as unequal. The idea of this method is to assign the most frequent symbol the minimum cost and the least frequent symbol the maximum cost code, whereas classical Huffman code assigns most frequent symbol the minimum length and the least frequent symbol the maximum length code. It is worth noting that while considering letter or bit cost as unequal, the length and cost of a codeword is not the same. For example, if we consider cost of the binary bit 1 as three times that of the cost of binary bit 0 (analogous to dash ($-$) and dot ($\cdot$) in Morse code) then codeword 011 has length 3 but cost 7 (considering cost of '0' as 1 and '1' as 3 unit).

For example, Table II shows 6 different symbols, their frequencies and the codewords generated for those symbols using both classical Huffman code and Huffman code with unequal letter costs. The trees generated by classical Huffman code and Huffman code with unequal letter costs are shown in Fig.3 and 4 respectively.

TABLE II. CODEWORD GENERATED BY VARIANTS OF HUFFMAN CODE

Symbol	Frequency	Codeword generated by Classical Huffman	Codeword generated by Huffman code with Unequal letter cost
A	5	010	0001
B	9	110	001
C	7	011	11
D	12	00	10
E	11	111	0000
F	15	10	01

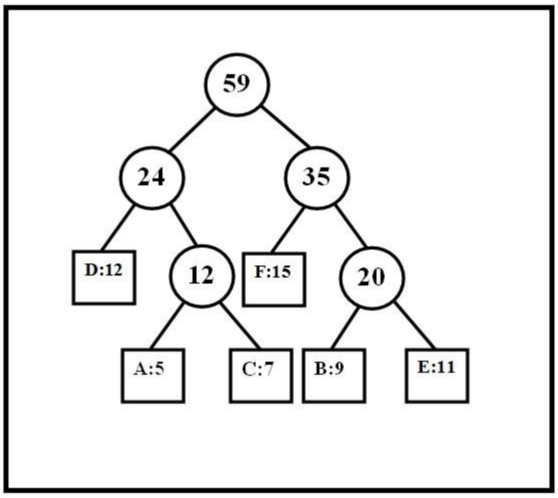


Fig. 3. Tree generated by Classical Huffman code

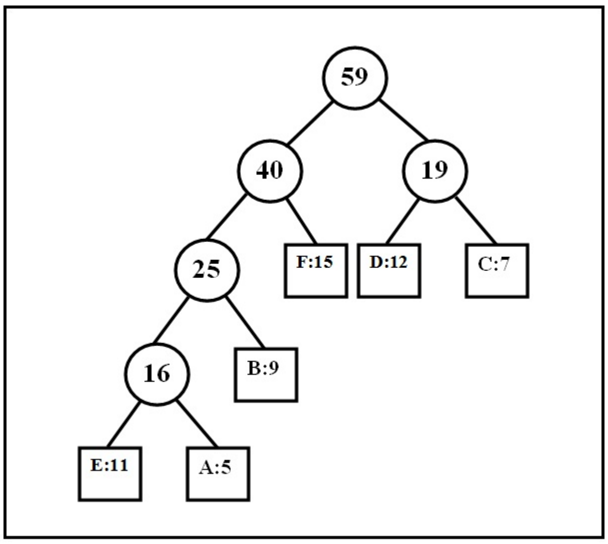


Fig. 4. Tree generated by Huffman code with unequal letter cost

III. Representation of Freeman Chain code using Huffman Code

Huffman code technique is an optimal encoding scheme that achieves the minimum amount of redundancy possible in a fixed set of variable-length codes. Its application reduces the average code length used to represent the symbols of the source alphabet. As Freeman chain code is a fixed-length coding scheme and does not take the frequencies of movements in different directions into account therefore application of this scheme results in more number of bits to encode an object. Huffman coding scheme could be used to help considering frequencies of movements in different directions in Freeman chain code, and thus reducing total number of bits. The process of using Huffman code to compress Freeman chain code is shown in Fig.5. At first the chain code of an image is generated using the Freeman chain code scheme then eight different directions and number of movements in those directions are provided to Huffman coding algorithm as symbols and frequencies of those symbols respectively. Huffman coding algorithm assigns variable length code to represent each direction by considering their frequencies in the chain. Table III shows the frequencies of angle changes in the chain representing the butterfly image of Fig.6 and also shows the fixed-length code assigned by classical Freeman chain code and the variable-length code assigned by the Huffman Code. To encode the chain of the butterfly image, the Freeman chain code takes 2751 bits, on the other hand Huffman code takes 2684 bits. At the same time cost (length) of the Freeman chain code is 5705 unit and cost of the Freeman chain code using Huffman code is 5634 unit for the butterfly image by considering the cost of binary bit ‘1’ as 3 unit and cost of bit ‘0’ as 1 unit (analogous to ‘−’ and ‘·’ of Morse code).

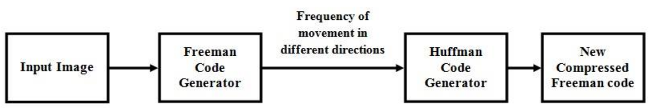


Fig. 5. Process of Representing Freeman chain code using Classical Huffman Code

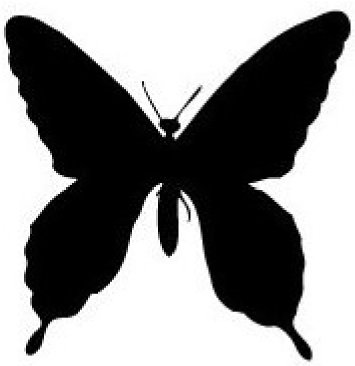


Fig. 6. Example image

IV. Representation of Freeman Chain code with unequal digits cost

In the above section, it is seen that the Huffman code could be used to compress the Freeman code. Classical Freeman

TABLE III. Codewords assigned by Freeman chain code and Freeman chain code using Huffman code

Angle Change	Frequency	Freeman Code	Freeman Code using Huffman
0°	59	000	1001
45°	122	001	110
90°	165	010	111
135°	114	011	011
180°	56	100	1000
-135°	115	101	101
-90°	182	011	00
-45°	104	111	010

code assigns fixed-length code and Huffman code technique assigns variable-length code to represent movements in different directions but none of the techniques take unequal letter cost into account while assigning codes. But in section II-B, we have seen that a considerable amount of research has been performed to address the issue of unequal letter cost problem. Therefore, if the aim of data encoding is to reduce the overall cost instead of number of bits then it required to consider unequal letter/bit cost during the encoding process. As described earlier, unequal letter cost version of Huffman code technique is available in the literature. But representation of the Freeman code with unequal letter cost has not been considered. A new representation of Freeman chain code with unequal bit cost is possible that can be considered analogous to Huffman code with unequal letter cost. The process of using unequal letter cost variant of Huffman code in representing Freeman chain code is shown in Fig.7. The chain code of an image is generated using the Freeman chain code scheme and then the outcome of the Freeman code is treated as input to Huffman code with unequal letter cost algorithm [33]. For simplicity and limited scope the details of the Huffman code with unequal letter cost is not discussed in this paper. The aim of the Huffman code with unequal letter cost scheme is to assign most frequent symbol the least cost code and vice versa. The resultant code of the new representation using Huffman code with unequal letter cost for the butterfly image of Fig.6 is shown in Table IV. A comparison between cost and bits required by Classical Freeman chain code, Freeman chain code using Huffman code, and Freeman chain code using Huffman code with unequal letter to represent the chain of the butterfly image is shown in Fig.8(a) and Fig.8(b) respectively. In terms of cost, it is seen from Fig.8(a) that the new representation of Freeman chain code using Huffman code unequal letter cost incurs the least cost to encode the chain, whereas classical Freeman chain code incurs most cost. On the other hand, the new representation with unequal letter cost takes more bits than other two representations and representation using classical Huffman code takes the fewest number of bits.

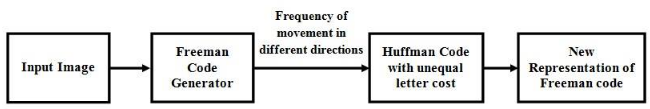
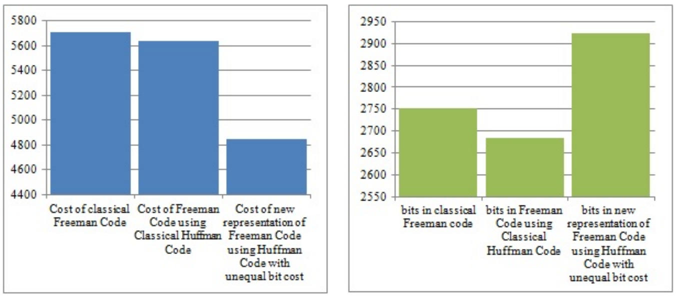


Fig. 7. Process of Representing Freeman chain code using Huffman code with unequal letter cost

TABLE IV. CODEWORD ASSIGNED BY FREEMAN CHAIN CODE AND FREEMAN CHAIN CODE USING HUFFMAN CODE WITH UNEQUAL LETTER COST

Angle Change	Frequency	Freeman Code	New Freeman code
$0°$	59	000	011
$45°$	122	001	010
$90°$	165	010	001
$135°$	114	011	11
$180°$	56	100	101
$-135°$	115	101	100
$-90°$	182	011	0000
$-45°$	104	111	0001



(a) Cost Comparison (b) Bits Comparison

Fig. 8. Comparison among different variants of Freeman chain code

V. RESULTS AND DISCUSSION

Freeman chain code technique does not consider the frequency of angle differences between two contiguous elements in its algorithm and assigns three bits fixed-length codes to represent movements in different directions. On the other hand, Huffman code technique considers frequency of the symbols that are contributing to the message to be encoded. It aims to increase the compression ratio basing on the frequency of incidence and does not consider cost of the bits representing the symbols, i.e. most frequent symbols are given shortest length codes and vice versa. It is possible to compress Freeman chain code using Huffman code as seen in the example of above section. But they do not take the cost of bits into account.The new representation of Freeman chain code takes cost of bits contributing to the code of the symbols into account. As a result, total cost of the compressed data is reduced as most frequent movement in a certain direction contribute relatively less to the cost of compressed data with compared to the both classical Freeman chain code and Freeman chain code using Huffman code.

We have analysed the experimental data such as cost of compressed data and bits required by three different representations of Freeman chain code. It focuses on cost reduction and bit overhead issues of the codes produced by three techniques. We have performed our experiments on 50 different sample images. The results of the experiments are presented in Fig.9 and Fig.10. Fig.9 shows the cost comparison among different representations of Freeman chain code in representing different images. It is seen from this figure that in case of every image sample, the new representation of Freeman chain code with unequal letter cost incurs least cost. Our experiments suggest that the technique reduces cost by 4.57% in the worst case scenario; in the best case it reduces cost by 22.15%, and on an average by 11.4%. We assert that the variation in performance is due to the varied shape of the images which is the deciding factor for number of movements in a particular direction. A mixed trends is seen between the cost of classical Freeman chain code and Freeman chain code using Huffman code. For some images the former performs better and for some other images the later one performs better. But in terms of cost reduction, the performance of the new representation considering unequal letter is the best among all three different representations. With regards to bits requirement to represent chains, the representation of Freeman chain code using Huffman code technique performs better than the other two representations as is evident in Fig.10.

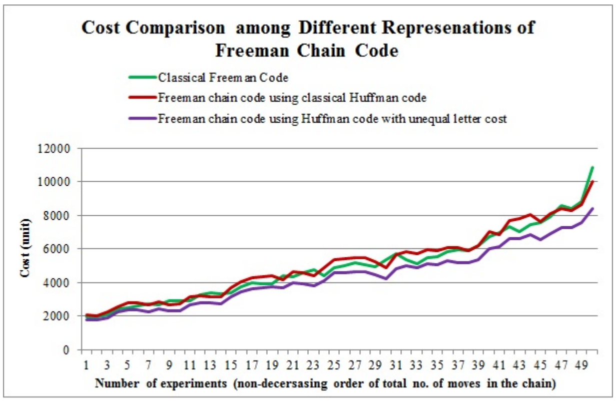


Fig. 9. Cost comparison among different representations of Freeman chain code

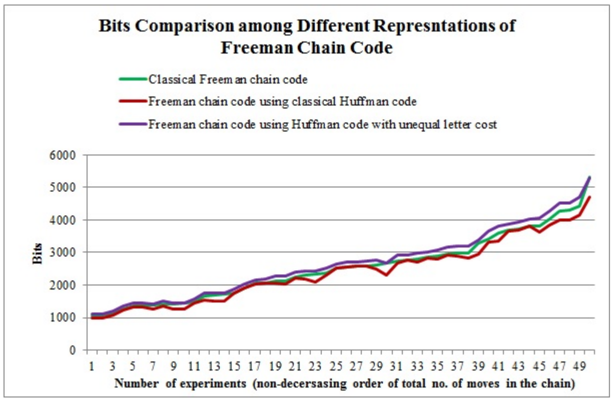


Fig. 10. Bits comparison among different representations of Freeman chain code

VI. CONCLUSION

The encoding efficiency to represent shapes of objects is very important in digital object recognition and analysis. Freeman chain code, a fixed-length encoding scheme, is an efficient lossless compression methods for representing binary objects and contours. Encoding symbols using predefined fixed-length Freeman chain code does not attain an optimum performance in terms of compression, because every symbols consumes equal number of bits. Representing Freeman chain code using Huffman code helps to attain an optimum compression performance because Huffman code assigns variable-length code to symbols based on their occurrence frequency. All these methods consider letter costs as equal. Equal letter cost algorithms consider the length of codeword of the elements in an Σ only and it does not consider the cost of bits creating the codeword. As a result, even though it might achieve a high compression ratio, it incurs a high cost as well. It assigns the lowest length code to the most frequent symbol. As the cost of bit is not considered, it is not unlikely that the cost would be much higher. On the other hand, the new representations of Freeman chain code using Huffman code with unequal letter costs considers the cost of bits contributing towards the cost of the codeword. Accordingly, it assigns least costly codeword to the most frequent symbol. Consequently, the overall cost is reduced as the most frequent symbols contribute more to the uncompressed data. Therefore, users can make a tradeoff between cost and compression ratio by choosing a suitable representations of Freeman chain code. In one hand, representation of Freeman chain code using classical Huffman code would attain good compression performance, on the other hand, Freeman code using unequal letter cost would produce cost efficient representation.

REFERENCES

[1] H. Freeman, "On the encoding of arbitrary geometric configurations," *Electronic Computers, IRE Transactions on*, vol. EC-10, no. 2, pp. 260–268, June 1961.

[2] T. Globanik and B. Žalik, "An efficient raster font compression for embedded systems," *Pattern Recognition*, vol. 43, no. 12, pp. 4137 – 4147, 2010.

[3] T.-L. Weng, S. Lin, W. Chang, and Y.-N. Sun, "Voxel-based texture mapping for medical data," *Computerized medical imaging and graphics*, vol. 26, no. 6, pp. 445–452, 2002.

[4] P. Nunes, F. Marqués, F. Pereira, and A. Gasull, "A contour-based approach to binary shape coding using a multiple grid chain code," *Signal Processing: Image Communication*, vol. 15, no. 78, pp. 585 – 599, 2000.

[5] M. Ren, J. Yang, and H. Sun, "Tracing boundary contours in a binary image," *Image and Vision Computing*, vol. 20, no. 2, pp. 125 – 131, 2002.

[6] K. I. Ho, T.-S. Chen, and C.-Y. Cheug, "An efficient face detection method using skin-color discovering and chain code," *Machine Graphics & Vision International Journal*, vol. 11, no. 2/3, pp. 241–256, 2002.

[7] F. Arrebola and F. Sandoval, "Corner detection and curve segmentation by multiresolution chain-code linking," *Pattern Recognition*, vol. 38, no. 10, pp. 1596 – 1614, 2005.

[8] S. Kaygin and M. Bulut, "A new one-pass algorithm to detect region boundaries," *Pattern Recognition Letters*, vol. 22, no. 10, pp. 1169 – 1178, 2001.

[9] H. Sun, J. Yang, and M. Ren, "A fast watershed algorithm based on chain code and its application in image segmentation," *Pattern Recognition Letters*, vol. 26, no. 9, pp. 1266 – 1274, 2005.

[10] I. K. G. D. Putra and M. A. Sentosa, "Hand geometry verification based on chain code and dynamic time warping," *International Journal of Computer Applications*, vol. 38, no. 12, pp. 17–22, February 2012.

[11] D. A. Huffman, "A method for the construction of minimum-redundancy codes," in *Proceedings of the Institute of Radio Engineers*, vol. 40, no. 9, September 1952, pp. 1098–1101.

[12] Y. K. Liu and B. Žalik, "An efficient chain code with huffman coding," *Pattern Recognition*, vol. 38, no. 4, pp. 553 – 557, 2005.

[13] D. Altenkamp and K. Mehlhorn, "Codes: Unequal probabilities, unequal letter costs," *Journal of the Association for Computing Machinery*, vol. 27, no. 3, pp. 412–427, July 1980.

[14] Y. Perl, M. R. Garey, and S. Even, "Efficient generation of optimal prefix code: Equiprobable words using unequal cost letters," *Journal of the ACM (JACM)*, vol. 22, no. 2, pp. 202–214, 1975.

[15] E. N. Gilbert, "Coding with digits of unequal costs," *IEEE Transactions on Information Theory*, vol. 41, 1995.

[16] M. J. Golin, C. Kenyon, and N. E. Young, "Huffman coding with unequal letter costs," in *ACM Symposium on Theory of Computing*, May 2002, pp. 785–791.

[17] E. Bribiesca, "A geometric structure for two-dimensional shapes and three-dimensional surfaces," *Pattern Recognition*, vol. 25, no. 5, pp. 483 – 496, 1992.

[18] ——, "A new chain code," *Pattern Recognition*, vol. 32, no. 2, pp. 235 – 251, 1999.

[19] ——, "A chain code for representing 3d curves," *Pattern Recognition*, vol. 33, no. 5, pp. 755 – 765, 2000.

[20] Y. K. Liu, W. Wei, P. J. Wang, and B. Žalik, "Compressed vertex chain codes," *Pattern Recognition*, vol. 40, no. 11, pp. 2908 – 2913, 2007.

[21] S. Priyadarshini and G. Sahoo, "A new lossless chain code compression scheme based on substitution," *International Journal of Signal and Imaging Systems Engineering*, vol. 4, no. 1, pp. 50 – 56, 2011.

[22] Y.-K. Liu, B. Žalik, P. jie Wang, and D. Podgorelec, "Directional difference chain codes with quasi-lossless compression and run-length encoding," *Signal Processing: Image Communication*, vol. 27, no. 9, pp. 973 – 984, 2012.

[23] W. A. Redmond, "International morse code," *Microsoft Encarta 2009 [DVD]*, pp. 275–278, 1964.

[24] R. Karp, "Minimum-redundancy coding for the discrete noiseless channel," *IRE Transactions on Information Theory*, vol. 7, no. 1, pp. 27–38, 1961.

[25] R. M. Krause, "Channels which transmit letters of unequal duration," *Information Control*, vol. 5, pp. 13–24, March 1962.

[26] B. Varn, "Optimal variable length codes -Arbitrary symbol cost and equal code word probability," *Information Control*, no. 19, pp. 289–301, 1971.

[27] N. Cot, "Characterization and design of optimal prefix codes," Ph.D. dissertation, Stanford University, 1977.

[28] J. Amsterdam, "Data compression with huffman coding," *BYTE*, vol. 11, no. 5, pp. 98–108, May 1986.

[29] M. A. Mannan and M. Kaykobad, "Block huffman coding," *Computers and Mathematics with Applications*, 2003.

[30] M. Golin and N. Young, "Prefix codes: Equiprobable words, unequal letter costs," *SIAM JOURNAL ON COMPUTING*, vol. 25, no. 6, pp. 1281–1292, DEC 1996.

[31] P. Bradford, M. Golin, L. Larmore, and W. Rytter, "Optimal prefix-free codes for unequal letter costs: Dynamic programming with the Monge property," *JOURNAL OF ALGORITHMS*, vol. 42, no. 2, pp. 277–303, FEB 2002.

[32] M. J. Golin, C. Mathieu, and N. E. Young, "Huffman Coding with Letter Costs: A Linear-Time Approximation Scheme," *SIAM JOURNAL ON COMPUTING*, vol. 41, no. 3, pp. 684–713, 2012.

[33] S. Kabir, T. Azad, A. S. M. A. Alam, and M. Kaykobad, "Effects of unequal bit costs on classical huffman codes," in *The 17th International Conference on Computer and Information Technology(ICCIT-2014)*, 2014, (accepted).