A Lossless Coding Scheme for Images Using Cross-Point Regions for Modeling

Tin T. Dang, *Member*, *IEEE*, Thanh D. Vu, *Member*, *IEEE*, Aaron Becker, *Member, IEEE*

***Abstract*—This paper presents CRIC (Cross-Point Regions for Lossless Image Compression), a scheme for losslessly encoding and decoding images, especially medical images, by optimizing on the probability of cross points that neighbor points of grey levels 2^n. The base of this statement is the effect of Gray coding on cross points. At first, the effect of Gray codes is determined on an adjacent data set because images characteristically contain data that does not change much in a specific area; then this effect is generalized for real data without losing generality for their statistical properties. This is especially true for medical images that have many regions with the same grey levels. The Gray code transformation makes the bit states of cross points change from the original data bits, so first the probabilities of data bits on specific bit planes in cross point regions and then the entropies of the messages are changed. These probabilities are estimated and compared with the probabilities of the original data bits. This change of probability has important effects on the encoding and decoding processes in lossless medical image compression.**

***Index Terms*—cross point, entire cross point region, ideal cross point region, probability of bits**

I. INTRODUCTION

THIS paper is based on the ideas of papers [1], [2] that presented the definition of cross points and cross point regions, the propositions on bit states of cross points and the consequences from the entropy of obtained data on cross point regions after Gray code transformation. It presents the effect of Gray coding on the process of calculating the probability of bits in cross point regions.

Cross points are neighbor points around the points of grey levels 2^n. The points of grey levels 2^n may or may not exist in the real data, but cross points are easily specified by [3]. The original data points whose values are less than 2^n have bit states much different from those of the data points greater than or equal to 2^n [4], [5]. The change of bit states for Gray code transformation has been studied by many authors [6]-[8], and the number of bits and the distribution of these bits of Gray codes in the regions of cross points are mentioned in [1], [2]. The real data are arbitrary, so the change of bit states is systematically developed in the form of probability of data bits in cross-point regions before and after Gray coding.

This paper has six sections. After this introduction, Section II briefly mentions the definition of cross-point regions and the propositions about characteristics of Gray codes of cross points, a part of which is presented in [1], [2]. The change of the probability of bits in cross point regions due to Gray coding is presented in Section III. Section IV introduces the scheme CRIC (Cross-point Regions for Lossless Image Compression). In Section V, some of our results obtained from using the above theory are presented. Section VI contains the conclusion and the scope for future research.

II. GRAY CODING AND ITS EFFECT ON CROSS POINTS

A. Effect of Gray coding on data bits in cross point regions on the bit plane $n-1$

DEFINITION 2.1. *For a set of integers from* 0 *to* (2^N-1)*, where* N *is a positive integer representing bit length, the region of cross points* 2^n ***A(n)****, with* n *from* 1 *to* $(N-1)$*, is a set of* 2^n *data points along the direction of encoding around the only value of* 2^n*, consisting of* 2^{n-1} *forward data points* $\boldsymbol{A_l(n)}$ *and* 2^{n-1} *backward data points* $\boldsymbol{A_g(n)}$ *(including the point of* 2^n *if it exists). The value* 2^n *is called the boundary of these two parts.*

If the region of cross points $\boldsymbol{A}$ contains 2^n points with contiguous values, the least significant bits always represent the difference of these values. For example, when $n = 3$ the region $\boldsymbol{A(3)}$ consists of the values 4, 5, 6, 7, 8, 9, 10, 11; the value $8 = 2^3$ ($= 2^n$) is called the boundary of two small regions $\boldsymbol{A_l(3)} = \{4, 5, 6, 7\}$ and $\boldsymbol{A_g(3)} = \{8, 9, 10, 11\}$. In this case the boundary 2^n exists explicitly.

In real data, where grey values of neighbor points often differ slightly from each other, this difference can be seen in the least significant bits while the more significant bits are often similar to each other in a certain region. This is completely like the string of adjacent data. Therefore, without losing generality for statistical properties, the following statements are presented for contiguous data, and their results are used to encode and decode real data, such as medical images.

Manuscript received December 24, 2006.

Tin T. Dang is with the Ho Chi Minh City University of Technology, 268 Lythuongkiet St., Dist. 10, Ho Chi Minh City, Vietnam (84-8-8650424; e-mail: dttin@hcmut.edu.vn).

Thanh D. Vu is with the Ho Chi Minh City University of Technology, 268 Lythuongkiet St., Dist. 10, Ho Chi Minh City, Vietnam (e-mail: vdthanh@hcmut.edu.vn).

Aaron Becker is with the Department of Electrical and Computer Engineering, 1406 W. Green St., Urbana IL 61801, University of Illinois at Urbana-Champaign (e-mail:abecker5@uiuc.edu).

These cross-point regions ***A(n)*** are called the *entire cross-point regions* (ECRs) to distinguish them from the *ideal cross point regions* (ICRs) in Definition 2.2.

PROPOSITION 2.1. *For a data string of adjacent values from* 0 *to* (2^N-1)*, where N is a positive integer representing bit length, let n be the exponent of the boundary value* 2^n *in the region of cross point **A(n)**, where n is from* 1 *to* ($N-1$)*. Bits of Gray codes in the backward region* $\boldsymbol{A_g(n)}$ *on the bit plane* $n-1$ *are* 1*. This region* $\boldsymbol{A_g(n)}$ *contains* 2^{n-1} *bits* 1*; therefore, the region **A(n)** on bit plane* $n-1$ *contains* 2^n *bits* 1.

Proof of Proposition 2.1.

According to Definition 2.1, with a value of n in the interval 1 to ($N-1$), 2^{n-1} data points in the region $\boldsymbol{A_l(n)}$, from $2^{n-1},\ldots,\ 2^n-1$ are expanded under the form of a polynomial of radix 2 as the following:

$$0.2^{N-1}+\ldots+0.2^n+1.2^{n-1}+x.2^{n-1}+\ldots+x.2^0. \quad (1)$$

The 2^{n-1} data points in the region $\boldsymbol{A_g(n)}$, from $2^n,\ldots,\ 2^n+2^{n-1}-1$ are expanded by the following polynomial:

$$0.2^{N-1}+\ldots+0.2^{n+1}+1.2^n+0.2^{n-1}+x.2^{n-2}+\ldots+x.2^0, \quad (2)$$

where x is bit 1 or 0.

After Gray code transformation, (1) and (2) become (3) and (4), respectively:

$$0.2^{N-1}+\ldots+0.2^n+1.2^{n-1}+x.2^{n-2}+\ldots+x.2^0, \quad (3)$$

$$0.2^{N-1}+\ldots+0.2^{n+1}+1.2^n+1.2^{n-1}+x.2^{n-2}+\ldots+x.2^0. \quad (4)$$

Equation (4) shows that the region $\boldsymbol{A_g(n)}$ on bit plane $n-1$ contains 2^{n-1} bits 1. From (3), we can see a similarity to the region $\boldsymbol{A_l(n)}$. By combining (3) and (4), the region ***A(n)*** on the bit plane $n-1$ contains 2^n bits 1.

For example, when $n=3$, ***A(3)*** = {4, 5, 6, 7, 8, 9, 10, 11}, $\boldsymbol{A_l(3)}$ = {4, 5, 6, 7} and $\boldsymbol{A_g(3)}$ = {8, 9, 10, 11}. In the region ***A(3)*** the decimal values from 4 to 11 have bits 1 on the bit plane 2 (= 3 − 1) in the form of Gray codes.

According to this proposition and the characteristic of real images, after Gray coding the probability of bit 1 is larger than the probability of bit 0 in ***A(n)*** and also larger than the probability of bit 1 being outside ***A(n)*** on the same bit plane.

DEFINITION 2.2. *Let the positive integer N be the bit length of data points. The region of cross points* $\boldsymbol{A_o(n)}$*, with n from 1 to (N – 1), is a set of data points whose grey values are from* (2^n-2^{n-1}) *to* ($2^n+2^{n-1}-1$)*. The point of grey value* 2^n *(if it exists) is called the center point of the cross point region, and the grey value* 2^n *is called the central value. These regions are the ICRs.*

By Definition 2.2, data points in ICRs have grey values that satisfy the rule

$$V_A(n) = \{\, 2^n-2^{n-1},\ \ldots,\ 2^n+2^{n-1}-1\}, \quad (5)$$

and the ECRs contain the ICRs in real data. We will see that in this scheme for lossless image compression, the position of ICRs determine ECRs by identifying center points 2^n with boundary points 2^n.

PROPOSITION 2.2. *Let n be the exponent of the central value* 2^n *in the ICR* $\boldsymbol{A_o(n)}$*, where n is from* 1 *to* ($N-1$) *and N is a positive integer representing bit length. Bits of Gray codes in the ICR* $\boldsymbol{A_o(n)}$ *on the bit plane* $n-1$ *are bits* 1.

Proof of Proposition 2.2.

According to Definition 2.2, and Proposition 2.1, with a value of n in the interval 1 to ($N-1$), grey values of data points in the region $\boldsymbol{A_o(n)}$ are from (2^n-2^{n-1}) to ($2^n+2^{n-1}-1$), so they may be expanded under the form of polynomials of radix 2 (1) and/or (2).

After Gray code transformation, (1) and (2) become (3) and (4) respectively. Both (3) and (4) always give bits 1 on the bit plane $n-1$. This is very good for compressing real data because the probability of bit 1 in ICRs is 1, and the probability of bit 0 in ICRs is 0. This fact plays an important role in optimizing the probability of a data bit in ECRs.

B. *Effect of Gray coding on data bits in cross point regions on the bit plane* $n-2$

DEFINITION 2.3. *For a set of integers from* 0 *to* (2^N-1)*, where N is a positive integer for bit length, the region of cross points* 2^n ***R(n)****, with n from* 2 *to* ($N-1$)*, is a set of* 2^{n-1} *data points along the direction of encoding around the only value of* 2^n*, consisting of* 2^{n-2} *forward data points and* 2^{n-2} *backward data points (including the point of* 2^n *if it exists) of the point of* 2^n*. The value* 2^n *is called the boundary of these two parts.*

According to Definition 2.3 the first point of 2^n (if it exists) is also the only center of the region ***R(n)***. This region is a union of two parts: the first set $\boldsymbol{R_l(n)}$ containing 2^{n-2} forward data points and the second set $\boldsymbol{R_g(n)}$ containing 2^{n-2} backward data points including even the point 2^n.

The region of cross points ***R*** contains 2^{n-1} points of contiguous values, where the least significant bits represent the difference of these values. For example, when $n=3$ the region ***R(3)*** consists of the values 6, 7, 8, 9; the value $8=2^3$ ($=2^n$) is called the boundary of two small regions $\boldsymbol{R_l(3)}$ = {6, 7} and $\boldsymbol{R_g(3)}$ = {8, 9}.

These ECRs are also applied to real data, in which grey values of data points differ slightly from each other at the least significant bits while the most significant bits are often similar to each other.

PROPOSITION 2.3. *For a data string of adjacent values from* 0 *to* (2^N-1)*, where N is a positive integer representing bit length, let n be the exponent of the boundary value* 2^n *in the region of cross point **R(n)**, where n is from* 2 *to* ($N-1$)*. Bits of Gray codes in the forward region* $\boldsymbol{R_l(n)}$ *on bit plane n* -2 *are* 0*. This region* $\boldsymbol{R_l(n)}$ *contains* 2^{n-2} *bit* 0*; therefore, the region **R(n)** on bit plane* $n-2$ *contains* 2^{n-1} *bits* 0.

Proof of Proposition 2.3.

According to Definition 2.2, with a value of n in the interval 2 to ($N-1$), 2^{n-2} data points in the region $\boldsymbol{R_l(n)}$, from $2^n-2^{n-2},\ldots,\ 2^n-1$ are expanded in the form of a polynomial of radix 2:

$$0.2^{N-1}+\ldots+0.2^n+1.2^{n-1}+1.2^{n-2}+x.2^{n-3}+\ldots+x.2^1+x.2^0. \quad (6)$$

The 2^{n-2} data points in the region $\boldsymbol{R_g(n)}$, from $2^n, \ldots, 2^n + 2^{n-2} - 1$ are expanded by the following polynomial:

$$0.2^{N-1} + \ldots + 0.2^{n+1} + 1.2^n + 0.2^{n-1} + 0.2^{n-2} + x.2^{n-3} + \ldots + x.2^0, \quad (7)$$

where x is bit 1 or 0.

After Gray code transformation, (6) and (7) become

$$0.2^{N-1} + \ldots + 0.2^n + 1.2^{n-1} + 0.2^{n-2} + x.2^{n-3} + \ldots + x.2^0, \quad (8)$$

$$0.2^{N-1} + \ldots + 0.2^{n+1} + 1.2^n + 1.2^{n-1} + 0.2^{n-2} + \ldots + x.2^0. \quad (9)$$

Equation (8) shows that the region $\boldsymbol{R_l(n)}$ on the bit plane $n - 2$ contains 2^{n-2} bits 0. From (9), we can see a similarity to the region $\boldsymbol{R_g(n)}$. By combining (8) and (9), the region $\boldsymbol{R(n)}$ on the bit plane $n - 2$ contains 2^{n-1} bits 0.

For example, when $n = 3$, $\boldsymbol{R(3)} = \{6, 7, 8, 9\}$, $\boldsymbol{R_l(3)} = \{6, 7\}$ and $\boldsymbol{R_g(3)} = \{8, 9\}$. In the region $\boldsymbol{R(3)}$ the decimal values from 6 to 9 have bits 0 on the bit plane 1 (= 3 − 2) when represented by Gray codes.

If applying this proposition to real data, we can certify that after Gray coding the probability of bit 0 is larger than the probability of bit 0 in $\boldsymbol{R(n)}$ and also larger than the probability of bit 0 being outside $\boldsymbol{R(n)}$ on the same bit plane. This statement is convenient for the process of optimizing the probability of data bits in the scheme of lossless image compression.

DEFINITION 2.4. *Let the positive integer N be the bit length of data points and the region of cross points* $\boldsymbol{R_o(n)}$, *with n from 2 to* $(N - 1)$, *be a set of data points whose grey values are from* $(2^n - 2^{n-2})$ *to* $(2^n + 2^{n-2} - 1)$. *The point of grey value* 2^n *(if it exists) is called the center point of the cross-point region, and the grey value* 2^n *is called the central value. These regions are the ICRs.*

According to this definition, grey values of data points in ICRs are in the following set:

$$V_R(n) = \{ 2^n - 2^{n-2}, \ldots, 2^n + 2^{n-2} - 1\}. \quad (10)$$

The entire cross point regions (ECRs) also contain the ideal cross point regions (ICRs) in real data. The position of ICRs will give us ECRs for changing the probability of bits in them.

PROPOSITION 2.4. *Let n be the exponent of the central value* 2^n *in the ideal cross point region* $\boldsymbol{R_o(n)}$, *where n is from* 2 *to* $(N - 1)$, *and N is a positive integer of bit length. Bits of Gray codes in the ideal cross point region* $\boldsymbol{R_o(n)}$ *on the bit plane* $n - 2$ *are bits* 0.

Proof of Proposition 2.4.

Based on Definition 2.4 and Proposition 2.3, with a value of n in the interval 2 to $(N - 1)$, grey values of data points in the region $\boldsymbol{R_o(n)}$ are from $(2^n - 2^{n-2})$ to $(2^n + 2^{n-2} - 1)$, so they may be expanded under the form of polynomials of radix 2 (6) and/or (7).

After Gray coding, (6) and (7) become (8) and (9), respectively. Both (8) and (9) always give bits 0 in ICRs $\boldsymbol{R_o(n)}$ on the bit plane $n - 2$. This is good for optimizing the probability of data bits of real data, because the probability of bit 1 in $\boldsymbol{R_o(n)}$ is always 0, and the probability of bit 0 in $\boldsymbol{R_o(n)}$ is always 1.

III. PROBABILITY OF DATA BITS IN CROSS POINT REGIONS

A. *Probability of data bits*

Define $P^A(b_{ij}^l \mid f_k(i, j, l))$ as the conditional probability of bit 1 ($b_{ij}^l = 1$) and bit 0 ($b_{ij}^l = 0$) in row i, column j on the bit plane l being encoded, in accordance with Proposition 2.1, where l is the significant number of the bit plane and is equal to n–1. These probabilities present the kth-order estimate [9] of the image, where the function $f_k(i, j, l)$ is a set of previous neighbor bits:

$$f_k(i, j, l) = \{b_{uv}^m \mid (m > l, \forall u, \forall v) \cup (m = l, u < i, \forall v) \cup (m = l, u = i, v < j)\}. \quad (11)$$

The subscript k in $f_k(i, j, l)$ presents the kth-order estimate, so the number of bits b_{uv}^m in the set $f_k(i, j, l)$ is $k - 1$; their order is defined beforehand.

When estimating bits in a cross-point region corresponding to a specific boundary value 2^n, we can change their probabilities from the original probabilities to decrease the entropy of data in cross-point region [1], and this allows a better compression ratio. The original probabilities are also the probabilities of bits outside cross-point regions due to Proposition 2.1 [1] and the opinions above. The rule for optimizing the probabilities of bits in cross point regions $\boldsymbol{A}$ is as follows. Define $P_{CR}^A(b_{ij}^l \mid f_k(i, j, l))$ as the probabilities of bit 1 ($b_{ij}^l = 1$) and bit 0 ($b_{ij}^l = 0$) in a cross point region $\boldsymbol{A}$. The term α is a real value in the range of [0, 1). The probability of data bits in a specific cross point region can be changed by the following rule without affecting the coding and decoding processes [10], [11]:

$$P_{CR}^A(b_{ij}^l = 1 \mid f_k(i, j, l)) = P^A(b_{ij}^l = 1 \mid f_k(i, j, l)) + \alpha . P^A(b_{ij}^l = 0 \mid f_k(i, j, l)), \quad (12)$$

$$P_{CR}^A(b_{ij}^l = 0 \mid f_k(i, j, l)) = P^A(b_{ij}^l = 0 \mid f_k(i, j, l)) - \alpha . P^A(b_{ij}^l = 0 \mid f_k(i, j, l)). \quad (13)$$

The value α can be obtained by averaging the Hamming distances of data bits in cross point regions before and after Gray coding.

In equations (12) and (13), the characteristic probability distribution of data bits in cross-point regions does not change, because we always have $P^{A}_{CR}(1|f_k(i,j,l)) + P^{A}_{CR}(0|f_k(i,j,l)) = P^{A}(1|f_k(i,j,l)) + P^{A}(0|f_k(i,j,l)) = 1$. Furthermore, this change of probability improves the entropy of data bits in cross point regions [1], [2], and perfectly implements the algorithm [10], [11] that uses a first in, first out (FIFO) queue to get the codeword. The original probabilities need to be computed by the context model of the kth-order estimate [9], which means they are the conditional probabilities of bits being encoded and neighbor bits. The neighbor bits were chosen by condition (11).

Similarly, with Definition 2.3 and Proposition 2.3, we can see that the probability of bit 0 is greater than the probability of bit 1 in ***R(n)*** and is also greater than the probability of bit 0 being outside ***R(n)*** on the same bit plane. Consequently, when coding bits in ***R(n)***, we change their probabilities to obtain a better compression ratio.

We now consider cross points in a cross-point region ***R(n)*** on the bit plane $n - 2$, defining $P^{R}(b^{l}_{ij} \mid f_{k'}(i,j,l))$ as the conditional probability of bit 1 ($b^{l}_{ij} = 1$) and bit 0 ($b^{l}_{ij} = 0$) in row i, column j on the bit plane l being encoded, in accordance with Proposition 2.3. Here l is also the significant number of the bit plane, and is equal to $n - 2$. These probabilities present the k'th-order estimate of the image. The function $f_{k'}(i,j,l)$ is also a set of previous neighbor bits and is determined by

$$
\begin{aligned}
f_{k'}(i,j,l) = \{ b^{m}_{uv} \mid \; & (m > l,\ \forall u,\ \forall v) \cup \\
& (m = l,\ u < i,\ \forall v) \cup \\
& (m = l,\ u = i,\ v < j) \}.
\end{aligned} \tag{14}
$$

The subscript k' in $f_{k'}(i,j,l)$ presents the k'th-order estimate, so the number of bits b^{m}_{uv} in the set $f_{k'}(i,j,l)$ is $k' - 1$ and their order is defined beforehand. In general, k' may be either equal to or different from k. This is defined beforehand in the algorithm. The rule for optimizing the probabilities of bits in cross point regions ***R*** is the following.

Define $P^{R}_{CR}(b^{l}_{ij} | f_{k'}(i,j,l))$ as the probabilities of bit 1 ($b^{l}_{ij} = 1$) and bit 0 ($b^{l}_{ij} = 0$) in a cross-point region ***R***. The term β is a real value in the range of [0, 1). The probability of data bits in a specific cross point region can be changed by the following rule without making any effects on the coding and decoding process [10], [11]:

$$
\begin{aligned}
P^{R}_{CR}(b^{l}_{ij} = 1 \mid f_{k'}(i,j,l)) = {} & P^{R}(b^{l}_{ij} = 1 \mid f_{k'}(i,j,l)) \\
& - \beta . P^{R}(b^{l}_{ij} = 1 \mid f_{k'}(i,j,l)),
\end{aligned} \tag{15}
$$

$$
\begin{aligned}
P^{R}_{CR}(b^{l}_{ij} = 0 \mid f_{k'}(i,j,l)) = {} & P^{R}(b^{l}_{ij} = 0 \mid f_{k'}(i,j,l)) \\
& + \beta . P^{R}(b^{l}_{ij} = 1 \mid f_{k'}(i,j,l)).
\end{aligned} \tag{16}
$$

The value β can also be obtained by estimating the Hamming distances of data bits in cross point regions before and after Gray coding.

In Equations (15) and (16), the characteristic probability distribution of data bits in cross point regions does not change because we always have $P^{R}_{CR}(1 | f_{k'}(i,j,l)) + P^{R}_{CR}(0 | f_{k'}(i,j,l)) = P^{R}(1 | f_{k'}(i,j,l)) + P^{R}(0 | f_{k'}(i,j,l)) = 1$. Furthermore, this change of probability also improves the entropy of data bits in cross point regions [1], [2], and perfectly appropriates to the Jones' algorithm [10], [11] with some supplements. The original probabilities need to be computed by the context model of the k'th-order estimate.

B. Hamming distance in calculating factors α and β

The values α and β describe the change of the probabilities before Gray coding $P^{A}(.)$, $P^{R}(.)$ and after that $P^{A}_{CR}(.)$, $P^{R}_{CR}(.)$ in (12), (13) and (15), (16). This change describes the state of bits being changed in cross-point regions, so the values of α and β have to be connected to this event. Within a specific cross-point region on the bit plane being encoded, the change between bits before and after Gray coding is the Hamming distance [9].

According to Definition 2.2 we call the ICR i of type ***A*** on the bit plane l CR(i, l). Let h_i be the Hamming distance between bits before and after Gray coding in CR(i, l), and let w_i be the width of that cross-point region. By definition, $h_i \in [0, w_i]$. The value h_i is the quantity of bit states that are changed. From Proposition 2.2 these bits are always in the same states, i.e. bits 1. This increases the probability of bits 1, $P^{A}_{CR}(1|\ldots)$, and decreases the probability of bits 0, $P^{A}_{CR}(0|\ldots)$ in the ECR that includes the region CR(i, l). A larger value of h_i corresponds with a larger probability of bits 1 in that ECR. Therefore, the values α_i are a standardization of the values h_i.

The value of α_i for optimizing the probability of bits in CR(i, l) is defined by

$$
\alpha_i = \frac{h_i}{w_i}. \tag{17}
$$

The final value of α on the bit plane l being encoded is determined by a rule F that assigns the value $\alpha(l)$ based on values α_i. In a general form it is

$$
\alpha(l) = F_{\alpha}\{\alpha_1, \alpha_2, \ldots, \alpha_i, \ldots, \alpha_{T(l)}\}, \tag{18}
$$

where $T(l)$ is the number of cross point regions of type ***A*** on the bit plane l that are determined by the algorithm of comparison [3]. The rule F_{α} determines the value $\alpha(l)$ and is proposed as

$$
\alpha(l) = \frac{\sum_{i=1}^{T(l)} \alpha_i}{T(l)}. \tag{19}
$$

In accordance with Definitions 2.1 and 2.2, there are N-2 values of α (l), where N is the bit length for representing pixels of images, l is the number of bit plane being processed, and $l = n - 1$, where n is the exponent of the boundary value. The set of $\alpha(l)$s is

$$S_{\alpha}^{l} = \{\ \alpha(l)\ \big|\ 1 \le l \le N-2\}$$

$$= \{\alpha(1),\ \alpha(2),\ \ldots\ \alpha(N-2)\ \}. \quad (20)$$

The set S_{α}^{l} can be called S_{α} for short.

Similarly, let h_j be the Hamming distance between bits before and after Gray coding in the ideal cross point region j of type $\boldsymbol{R}$ on the bit plane m, denoted as CR(j, m) in accordance with Definition 2.4. Let w_j be the width of that cross-point region. For $h_j \in [0, w_j]$, the value h_j is the number of bits whose states were changed in CR(j, m). From Proposition 2.4 these bits are always bits 0; the larger the value h_j, the bigger the probability of bits 0. The values β_j are a standardization of the values h_j. The value of β_j for calculating the probability of bits in CR(j, m) is defined by

$$\beta_j = \frac{h_j}{w_j}. \quad (21)$$

The final value of β on the bit plane m being encoded is determined by a rule F that assigns the value $\beta(l)$ based on the values β_j. In a general form it is

$$\beta(m) = F_{\beta}\ \{\beta_1,\ \beta_2,\ \ldots,\ \beta_j, \ldots,\ \beta_{U(m)}\}, \quad (22)$$

where $U(m)$ is the number of cross point regions of type $\boldsymbol{R}$ on the bit plane m being determined by the algorithm of comparison in [3]. The rule F_{β} for determining the value $\beta(m)$ is proposed as

$$\beta(m) = \frac{\sum_{j=1}^{U(m)} \beta_j}{U(m)}.$$

(23)

In accordance with Definitions 2.3 and 2.4, there are $N - 2$ values of $\beta(m)$, so the set of $\beta(m)$s is

$$S_{\beta}^{m} = \{\ \beta(m)\ \big|\ 0 \le m \le N\text{-}3\}$$

$$= \{\ \beta(0),\ \beta(1),\ \ldots\ \beta(N\text{-}3)\ \}. \quad (24)$$

Each $\alpha(l)$ on the bit plane l corresponds with one $\beta(m)$ on the bit plane m, and in [2] we have $m = l - 1 = n - 2$, where n is the exponent of the boundary value. The set S_{β}^{m} can be called S_{β}^{l-1}, or S_{β} for short.

Therefore, when encoding with a specific boundary value, the equations (12), (13) and (15), (16) use values α (l) and $\beta(m)$, respectively. This means with two sets S_{α}^{l}, S_{β}^{l-1} the pairs are used together as follows:

$$S_{\alpha,\beta} = \{\ \ (\alpha(l),\ \beta(l-1))\ \big|\ 1 \le l \le N-2\ \} \quad (25)$$

$$= \{\ (\alpha(N-2),\ \beta(N-3)),$$

$$\vdots$$

$$(\alpha(2),\ \beta(1)),$$

$$(\alpha(1),\ \beta(0))\ \}.$$

The set $S_{\alpha,\beta}$ is called the entire set of values α, β. Each pair in $S_{\alpha,\beta}$ is automatically detected after cross point regions on each bit plane are determined.

IV. The CRIC Scheme for Lossless Image Compression

Figure 1 presents a broad overview of the CRIC scheme. Each step is numbered according to the sequence of the scheme, so we have 7 steps from 1 to 7. Step 1 (cross-point regions) looks for regions of cross point where we can optimize the probability of data bits. These regions are coarsely found because step 5 (cross-point regions) will identify exactly the regions where the probabilities of data bits need to be optimized by calculating factors α and β. Some cross-point regions satisfying step 1 but not satisfying step 5 will be canceled. Step 2 (Gray coding) carries out Gray code transformation over the original data. After that, Step 3 (Coarse probabilities) computes the probabilities of data bits with kth-order and k'th-order estimates. Step 4 computes the factors α and β, Step 5 evaluates cross-point regions. Step 5 will cancel entire cross-point regions that are not good at the encoding process; these ECRs contain very small ICRs. Step 6 optimizes probabilities of data bits in ECRs. The process of encoding at Step 7 (coding) uses Jones' algorithm [10]; it will give us the data, a compressed image.

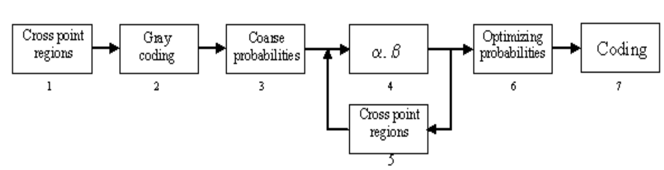

Fig. 1 The CRIC scheme with the theory of cross-point regions for lossless image compression

TABLE I

EXPERIMENTAL RESULTS OF LOSSLESS IMAGE COMPRESSION WITH CRIC

Image	Size	CRIC	AES (WinRAR 3.40) [13]	JPEG 2000 [14]
COUPLE	512 x 512	1.61501 : 1	1.42748 : 1	1.51374 : 1
LENA	512 x 512	1.65900 : 1	1.56707 : 1	1.75220 : 1 *
ZELDA	256 x 256	1.77204 : 1	1.51162 : 1	1.60314 : 1
FROG	621 x 498	1.58712 : 1	2.12112 : 1 *	1.21746 : 1
MOUNTAIN	640 x 480	1.47520 : 1	1.52020 : 1 *	1.17767 : 1
MANDRILL	512 x 512	1.23541 : 1	1.20657 : 1	1.29101 : 1 *

* These results are better than those of CRIC.

V. EXPERIMENTAL RESULTS

Table I presents results for images compressed by the scheme CRIC. The compression ratio used here is the ratio between files of images, i.e. including the headers of the original image and the compressed image. With these results we can see that CRIC is good for images that have many similar grey levels, especially medical images containing backgrounds with few different grey levels. These results are also better than those in [12], which compressed only medical images.

The algorithm for CRIC uses $k = k' = 4$, so there are 3 neighbor bits of the bit being coded. These are chosen beforehand. From the results in Table I we can see that when using CRIC to losslessly compress images (Couple, Lena, Zelda, Frog, Mountain, Mandrill), we can obtain higher compression ratios than other schemes (AES, JPEG 2000). For images whose grey levels change a great deal in cross point regions, like Mandrill, CRIC may give better or worse results than other methods, depending on the analyses of each method.

The encoding process here uses Jones' method [10]. This algorithm is like arithmetic coding but uses integers in the processing. At this point the program can process 8-bit grey images of bitmap file format. The restored images must be identical to the original images.

VI. CONCLUSION

The encoding scheme CRIC and some results were presented to illustrate the use of the theory of cross-point regions to optimize the probabilities of data bits in those regions. Generally, the scheme CRIC is a process of entropy coding. It includes two parts: modeling and coding. The cross-point region theory can be used in the first part in order to reduce interpixel redundancy; the second part uses Jones' method to reduce coding redundancy.

The basic concepts were introduced in [1], [2], which are now the foundation for the theory of optimizing probabilities of cross points with the new concept of ideal cross-point regions. A meaningful improvement in compression ratio has been obtained, compared to other authors' methods. From this mathematically sound foundation, the problem of improving the compression ratio of image processing and transmission can be developed further in the future.

REFERENCES

[1] D. T. Tin, V. D. Thanh, and N. K. Sach, *Effect of Gray codes on data areas around cross points*, Vietnamese journal of science and technology, vol 42, No 2, 2004.

[2] D. T. Tin, V. D. Thanh, N. K. Sach, and S. Higuchi, *Effect of Gray coding on the State and the Distribution of data bits in cross point regions*, Proceedings of ICCEA, IEEE catalog No. 04EX875, Beijing, China, Nov 2004, pp. 553 – 556.

[3] D. T. Tin, *Effect of Gray codes on data areas around cross points*, The 2nd report of Doctor thesis, HCMUT, Vietnam, 2004.

[4] C. J. Lu and S. C. Tsai, *A note on iterating an α-ary Gray code*, SIAM J. Discrete Math vol 14, No 2, 2001, pp. 237-239.

[5] N. K. Sach, *Digital Images and video processing,* Science and Technology Publisher, Vietnam, 1997.

[6] M. C. Er, *On generating the N-ary reflected Gray codes*, IEEE transactions on computers, 33 (1984), pp. 739-741.

[7] W. Gilbert, *A cube- filling Hilbert curve*, Math Intell 6 (1984) 78.

[8] M. B. Squire, *Gray codes and Efficient Generation of Combinatorial Structures*, Ph.D. dissertation, Department of Computer Science, North Carolina State University, 2001.

[9] R. C. Gonzalez, R. E. Woods, *Digital Image Processing*, Prentice Hall, 2002, pp 438- 440.

[10] C. B. Jones, *Efficient Coding System for Long Source Sequences*, IEEE Trans. Inform. Theory, 27(3), 280-291, 1981.

[11] D. T. Tin, V. D. Thanh, and N. K. Sach, *An improvement of Jones' method of lossless data compression*, Posts and Telecommunications journal, No. 7, 5-2002.

[12] D. T. Tin and S. Higuchi, *A lossless coding scheme for image,* Medical Imaging and Information sciences journal (Japan), vol.18, No. 2, May 2001.

[13] Federal information processing standards publication 197, *Announcing the Advanced Encryption Standard*, National Institute of Standards and Technology (NIST), Nov 2001.

[14] T. Acharya and P.-S. Tsai, *JPEG2000 Standard for image compression*, Wiley, 2005.