BETTER COMPRESSION PERFORMANCE BY COMPRESSING SLICED BIT PLANES OF IMAGE USING WEIGHTED FINITE AUTOMATA

A. Mishra¹, G. Sahoo² and A. Mustafi³

Department of Computer Science & Engineering, Birla Institute of Technology, Mesra, Ranchi-835215 India
E-mail: ¹ anandmishra.jsp@gmail.com, ²gsahoo@bitmesra.ac.in, ³abhijit@bitmesra.ac.in

ABSTRACT: Weighted Finite Automata (WFA) is an efficient structure for digital image storage and is successfully applied for the image compression purpose. This paper introduces the idea of slicing image into eight bit planes and then compressing each bit planes individually using proposed WFA inference algorithm. Furthermore, the proposed algorithm uses the concept of invertible resolution wise mapping in order to create rich families of similarity and thus minimize the WFA size and eventually achieve better compression. The algorithm is simulated successfully for gray scale test images.

Keywords: WFA, PSNR, Resolution wise and resolution driven mapping

1. INTRODUCTION

The efficient storage and transmission of digital image is a challenging field, especially in multimedia and web applications where image require a huge amount of storage capacity, even in compressed form like JPEG and GIF. This is the reason why researchers are taking enormous interest in the field of image compression.

For a randomly selected pixel it can be seen that its neighbors will have the same gray values or very similar gray values. Thus the fact that the neighboring pixels of image are highly correlated can be used for image compression. This correlation is known as spatial redundancy and this type of similarity in image is called – self similarity may be explored for compression purpose.

In this regard, Weighted Finite Automata (WFA) [3] can be used as an efficient structure for image representation. The WFA is basically composed of various state images, where quadrant of each state image is a linear combination of other state image(s). The WFA with minimum states provide better compression. Hence, the main investigation of our research work will be how to minimize the size of WFA that can enhance image compression. For this we have proposed an enhanced version of WFA inference [10, 15, 19] algorithm which exploits the concept of invertible resolution driven mappings proposed in [11] in order to find similarity among sub-images and state images and which is applicable to bit planes of image.

The paper is organized as follows: in section 2 we explored related work. Mathematical background of WFA technique is discussed in section 3. Section - 4 talk about the concept of resolution driven and resolution wise mapping. In section 5 we proposed a new version of WFA inference algorithm applicable to the bit planes of an image. Section 6 discusses results followed by conclusions in section 7.

2. RELATED WORK

Image compression is a mature area of research started way back in 1940’s but yet one of the burning topics because of advancements in multimedia data [2, 12]. We can categorize image compression techniques into three major generations: first generation (1940-1960), second generation (1970-1990) and modern generation (1990 – present). [17]

In first generation entropy encoding, 2D and 3D prediction coding were used for image compression. Some of popular techniques in this period were Shannon law, Huffman tree [7], Arithmetic Coding [17], Run length Encoding [18] and DPCM [17]. Only very low compression ratio was achieved by these techniques.

In Second generation transform coding like DCT [16], Block truncation coding [17] are used for image compression. This improved the compression performance significantly.

The modern generation uses either the concept of self similarity in the image or hybrid techniques and
achieves far better compression ratio than the earlier ones. Some linear algebraic concepts are also used for compressing facial image in recent years [6]. The JPEG, IFS and WFA approach are the products of this era.

JPEG [1] is a sophisticated compression method for color or grayscale still image. The Iterated Function System [14] encoder partitions the image into parts called ranges; it then matches each range to some other part called a domain and produces an Affine Transformation from domain to range. The WFA approach introduced by Culik et al. [3] is heavily motivated by IFS technique but differs by the fact that it uses weighted finite automata to represent image and thus takes the advantage of representing sub-image as linear combination of state images. In this way, WFA approach is more powerful than IFS and JPEG [11].

WFA inference algorithm proposed by Culik et al. (see [4, 13]) and thereafter (see [5, 9, 17]) tries to find out weights which can be used for expressing the sub-image as a linear combination of state images. But, finding such weights for a real life image is an exhaustive task. Moreover, till now no well-defined algorithms for finding weights has been applied to the state images in order to express a sub-image. In order to solve this problem we split the given gray scale image into eight bit planes and compress each bit plane individually which ultimately seems to be an easy task. Because by merely applying invertible resolution wise mapping to states of domain pool, we can get range image (sub-images).

3. MATHEMATICAL BACKGROUND

3.1. Quadtree Approach of Image Partitioning

Quadtree approach of image partitioning was discussed in [13]. The quadtree is a hierarchical data structure, based on divide and conquer principle leading to recursive partitioning. A quadtree is basically a tree where every node except the leaf nodes has four children. The four children are known as SW, NW, SE and NE.

In quadtree partitioning approach, image X is divided into four equal quadrants recursively and numbering is done from top left (NW) by appending 1, 3, 2 and 0 respectively to the left of existing address of the respective sub-quadrant as shown in Figure 1. In general, if an address of a sub-image is $a_1a_2...a_k$, then it means the sub-image is in quadrant $a_k$ of $a_{k-1}$ of ... of $a_2$ of $a_1$ quadrant of the given image.

![Figure 1: (a) Quadtree Numbering (b) Sub-images Addressable via Quadtree](image)

3.2. Weighted Finite Automata (WFA)

We define Weighted Finite Automata (WFA) A formally as quintuple $(Q, S, W, I, F)$ where,

(i) $Q = \{1, 2, ..., n\}$ is the finite set of states.

(ii) $S = \{a_1, a_2, ..., a_m\}$ is finite set of alphabet. (In particular, for quadtree partitioning $S = \{0, 1, 2, 3\}$).

(iii) $W(a): \mathbb{Q} \times \mathbb{Q} \rightarrow \mathbb{R}$ is the weight function.

(iv) $I: \mathbb{Q} \rightarrow \mathbb{R}$ is the initial distribution.

(v) $F: \mathbb{Q} \rightarrow \mathbb{R}$ is the final distribution.

Following Figure 2 illustrates how to construct a WFA for a given image:

![Figure 2: Constructing WFA for a Given Image (State q0)](image)
Gray value at any quadrant can be calculated by multiplying weight matrices, initial distribution and final distribution. For example, gray value at quadrant 01 can be calculated as follows:

\[
f(0\ 1) = I \cdot W_0 \cdot W_1 \cdot F
\]

\[
= (10) \begin{pmatrix} 1 & 2 & 0 \\
0 & 1 & 0
\end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \\
0 & 2 & 4
\end{pmatrix} \begin{pmatrix} 1 \\
0
\end{pmatrix} = \frac{1}{4}
\]

In the similar way Extended WFA [10] is computed for an image. The only difference is rather than searching self-similarity in image, sub-image is expressed as rotation or combination of rotation and weighted sum of states. Extended WFA outperforms linear WFA in image compression. [10]

4. Resolution-Driven and Resolution-Wise Mappings

**Definition 4.1:** A resolution-driven mapping \( f \) is a mapping that maps a finite sequence of multi resolution images \( \{x_i\} \) into another multi resolution image \( X \), satisfying the following invariance condition:

For any \( k \in \mathbb{N} \), if \( x_i \) and \( y_i \) are indistinguishable at the resolution of \( 2^k \times 2^k \) pixels \( \forall i \), then image \( x = f(x_i) \) and image \( y = f(y_i) \) are also indistinguishable at the same resolution.

**Definition 4.2:** A resolution-driven mapping is said to be a resolution-wise mapping if the resulting image \( X \) at any given resolution is completely determined through \( f \) by the images \( \{x_i\} \) at the same resolution. An invertible resolution wise mapping contains the horizontal mirror mapping \( m \), the diagonal mirror mapping \( d \), the anticlockwise rotation \( r \) by 90°, as well as the permutation of quadrants \( p \). Figure 3 shows some typical invertible resolution wise mapping.

5. Proposed Work

In this work we present a novel WFA inference algorithm which uses concept of invertible resolution wise mapping to find self similarity in image. This algorithm works for bit plane level of gray scale image. Representing bit planes as linear combinations of state images using invertible resolution wise mapping is an easier task than that of representing gray scale image directly. Additionally, in the least significant bit planes we can drastically compress the image using proposed algorithm as least significant bit planes simply has no or very minute significance in overall appearance of the image. Further, as most significant bit planes are less random, they can also be highly compressed.

5.1. WFA Inference Algorithm

The proposed WFA inference algorithm starts with empty domain pool (i.e. zero states). In each outer for loop we add one bit plane in the domain pool(set of states) \( Q \). The image is sliced into bit planes using following formula:

\[
X_i = \gg i (\& (X, 2))
\]

Here \( X \) is the \( i \)th bit-plane of the gray scale image \( X \) and \( \gg \) and \( \& \) denotes i bit right shift and bit wise AND operation respectively. We then divide the image planes using traditional quad tree representation (QuadPart in our algorithm) based on similarity and encode each part as the linear combination of invertible resolution wise mapping of available states.

QuadPart procedure also takes consideration into the bit-plane number while dividing image into quadrants based on similarity. As least significance bit planes can be highly compressed without loss of much information and thus QuadPart partitions least significance bit planes into larger sub images. For best approximation of a sub-image \( q_j \) with linear combination of invertible resolution wise mapping of available states \( \sum_{k=1, l=1}^{k=n, l=m} \Psi_k (Q_i) \) we use L2-norm of their difference to be as less as possible (of course with respect to the size of \( q_j \)) i.e.

\[
\left\| q_j - \sum_{k=1, l=1}^{k=n, l=m} \Psi_k (Q_i) \right\| < \alpha, \alpha > 0
\]
6. RESULTS AND DISCUSSIONS

The experiment investigates the compression ratio of the test image in terms of bits per pixel (bpp) against loss of information in terms of Peak Signal to Noise Ratio (PSNR). The compression is defined as the memory space holding the original data to that of holding the compressed data. If the pixels of the original image and reconstructed image are denoted as \( P_i \) and \( Q_i \) respectively, then Peak Signal to Noise Ratio (PSNR) of the reconstructed image is defined as follows:

\[
PSNR = 20 \log_{10} \frac{\max P_i}{RMSE}
\]

where RMSE (Root Mean Square Error) is defined as square root of Mean Square Error (MSE) which defined by:

\[
MSE = \sum_{i=0}^{n} (P_i - Q_i)^2
\]

Algorithm:

Input: Given Image X
Output: State and weight information.

Set \( Q \leftarrow NULL \) (\( Q \) is set of states)
Set \( n \leftarrow 0 \) (\( n \) is number of states)
For every bit plane \( X_i \)
    \( n \leftarrow n + 1 \)
    \( Q \leftarrow Q \cup \{X_i\} \)
For every unprocessed state \( Q_i \in Q \)
    For every \( q_j \in \text{QuadPart}(Q_i) \)
        If \( \exists \) invertible resolution wise mapping \( \Psi_i, \ldots, \Psi_n \) and states \( Q_i, \ldots, Q_n \) such that
        \[
        \| q_j - \sum_{i=1}^{n} \Psi_i(Q_i) \|_{\text{size}(q_j)} < \alpha, \alpha > 0
        \]
        Create edge(s) between states \( Q_i \) and \( q_j \) having weight \( \Psi_i(Q_i) \) for \( k = 1, 2, \ldots, n \) and \( l = 1, 2, \ldots, m \)
    Else
        Create state \( q_j \)
        Set \( Q \leftarrow Q \cup \{q_j\} \)
        Set \( n \leftarrow n + 1 \)
        Create an edge:
        Between \( Q_i \) and \( q_j \) having weight
        \( (\Psi_i \text{ is identity transformation}) \)
    (End if-else)
(End for)
Mark \( Q_i \) as processed
(End for)
(End for)

Here without loss of generality we treat the image part and state image as vectors. Further, by notation \( \Psi_i(Q) \), we mean invertible resolution wise transformation \( \Psi_i \) is applied to state \( Q \). We vary the value of \( \alpha \) from 0.2 to 0.6 based on significance of sub image.

If we do not found any such approximation we add \( q_j \) in the domain pool having weight \( \Psi_0(q_j) \) where \( \Psi_0(q_j) \) is identity transformation on state \( q_j \). The number of states grows fast in the first iteration but with very next iterations existing states are used and thus only few new states are included.

The novelty and superiority of the method lies in the facts that (1) It creates rich families of similarity (2) Individual bit-plane compression makes the weight finding process faster and better. (Note that finding weights to encode gray images directly is an exhaustive task).

Figure 4: Test Images used for Experiments

We used the standard gray scale test images shown in Figure 4 for all our experiment. Resolutions of the images are 256 × 256 pixels.

We compared the compression performance and loss of information in terms of PSNR for the linear WFA proposed by Culik et al., Extended WFA proposed by Jiang et al. against proposed algorithm. Results are shown in Figure 5 and Table 1. We
Better Compression Performance by Compressing Sliced Bit Planes of Image using Weighted...

demonstrate PSNR for the reconstructed image when standard test images are compressed by proposed algorithm and compare it with linear and extended WFA methods.

Figure 5 and table 1 clearly illustrate a higher PSNR (i.e. less noise) is achieved by proposed algorithm which explicitly means better compression performance by proposed algorithm.

Further table 1 also supports our claim that proposed algorithm yields best compression performance for image which has high self similarity like airplane image (see figure 4(c)) where we are getting PSNR = 61.22 dB.

<table>
<thead>
<tr>
<th>Test Image</th>
<th>Linear WFA (in dB)</th>
<th>Extended WFA</th>
<th>Proposed Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>36.23</td>
<td>38.92</td>
<td>42.42</td>
</tr>
<tr>
<td>House</td>
<td>36.38</td>
<td>40.00</td>
<td>45.08</td>
</tr>
<tr>
<td>Airplane</td>
<td>49.73</td>
<td>50.34</td>
<td>61.22</td>
</tr>
<tr>
<td>Mandrill</td>
<td>40.12</td>
<td>44.23</td>
<td>54.10</td>
</tr>
</tbody>
</table>

Table 1 Performance of Proposed Approach v/s Linear and Extended WFA on Various Standard Test Images

Figure 5: Proposed WFA Inference Algorithm v/s Extended WFA and Linear WFA in terms of PSNR

7. CONCLUSIONS

The WFA inference algorithm proposed in this paper creates a more affluent family of similarity compared to existing WFA inference algorithms Moreover the proposed algorithm which is applied to sliced bit-planes of an image individually solves the problem of finding appropriate weights in the WFA. The algorithm works both for the least significant bit planes (where the effect of loss of information is negligible) and most significant bit planes (where huge loss of information is not tolerable). Actually, this algorithm works even better for the most significant bit planes where the amount of self similarity is higher. We conclude here that the proposed algorithm yields excellent compression ratio with minute effect on the reconstructed image. We have verified our claims through experiments.

References


