Similarity Enrichment in Image Compression through Weighted Finite Automata

Zhuhan Jiang[1], Bruce Litow[2], and Olivier de Vel[3]

[1] School of Mathematical and Computer Sciences, University of New England, Armidale NSW 2351, Australia
zhuhan@mcs.une.edu.au

[2] School of Information Technology, James Cook University, Townsville, QLD 4811, Australia
bruce@cs.jcu.edu.au

[3] Information Technology Division, DSTO, PO Box 1500, Salisbury SA 5108, Australia
olivier.devel@dsto.defence.gov.au

Abstract. We propose and study in details a similarity enrichment scheme for the application to the image compression through the extension of the weighted finite automata (WFA). We then develop a mechanism with which rich families of legitimate similarity images can be systematically created so as to reduce the overall WFA size, leading to an eventual better WFA-based compression performance. A number of desirable properties, including WFA of minimum states, have been established for a class of packed WFA. Moreover, a codec based on a special extended WFA is implemented to exemplify explicitly the performance gain due to extended WFA under otherwise the same conditions.

1 Introduction

There are many different compression schemes each of which concentrates on often specific image or compression characteristics. For instance, a wavelet based compression scheme makes use of the multiresolution analysis, see e.g. [1] and the references there, while the JPEG standard hinges on the signal superposition of the Fourier harmonics. Alternatively, similarities among given images and their subimages may be explored to decorrelate the image components for the perceived compression purpose. In this regard, weighted finite automata (WFA) have been successfully applied to both image and video compressions, see e.g. [2,3,4]. These WFA are basically composed of subimages as the states, and the linear expansions of the state images in terms of themselves and the rest of state images. For a given image under consideration, a typical inferred WFA will select a collection of subimages to form the WFA states in such a way that the given original image can be reconstructed, often as a close approximation, from the inferred WFA. Such a WFA, when further encoded, may then be regarded as a compressed image. The main objective of this work is therefore to propose a similarity enrichment scheme that will reduce the overall size of an inferred

D.-Z. Du et al. (Eds.): COCOON 2000, LNCS 1858, pp. 447–456, 2000.

WFA, to set up the basic structure and develop interrelated properties of the extended WFA, and to establish a mechanism that determines systematically what type of similarity enrichment is theoretically legitimate for a WFA based compression. In fact we shall also devise a simplified yet still fully-fledged image compression codec based on a newly extended WFA, and demonstrate explicitly that, under otherwise the same conditions, a codec based on similarity enriched WFA will outperform those without such an enrichment.

Throughout this work we shall follow the convention and assume that all images are square images, and that all WFA based compressors are lossy compressors. We recall that an image $\mathbf{x}$ of the resolution of $n \times n$ pixels is mathematically an $n \times n$ matrix whose (i, j)-th element $x_{i,j}$ takes a real value. For any image of $2^d \times 2^d$ pixels, we define its *multiresolution representation* $\rho(\mathbf{x})$ by $\rho(\mathbf{x}) = \{\rho_{_D}(\mathbf{x}) : D \in \mathbb{N}\}$, where $\mathbb{N} = \{0, 1, 2, 3, ...\}$ and $\rho_{_D}(\mathbf{x})$ denotes the same image at the resolution of $2^D \times 2^D$ pixels. In fact we will identify any image $\mathbf{x}$ with its multiresolution representation $\rho(\mathbf{x})$ unless otherwise stated.

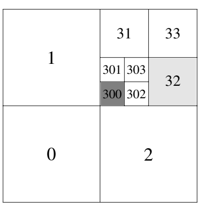

Fig. 1. Subimages addressable via a quadtree

In order to address pixels of all resolutions in $\rho(\mathbf{x})$, we follow the convention in [2] and divide $\mathbf{x}$ into 4 equal-sized quadrants, numbering them clockwise from the top left by 1, 3, 2 and 0 respectively. These numbers are the *addresses* of the corresponding quadrant subimages. Let ϵ be an empty string and $Q = \{0, 1, 2, 3\}$ be the set of alphabet. Then $\sigma_a(\mathbf{x})$ with $a \in Q$ will denote all four quadrant subimages of $\mathbf{x}$. Furthermore the subimage defined by

$$\sigma_{a_1 a_2 \cdots a_k}(\mathbf{x}) = \sigma_{a_k}(\sigma_{a_{k-1}}(\cdots \sigma_{a_2}(\sigma_{a_1}(\mathbf{x})) \cdots)) \quad (1)$$

for $a_1, a_2, ..., a_k \in Q$ will thus represent a_k-th quadrant of a_{k-1}-th quadrant of $\cdots$ of a_2-th quadrant of a_1-th quadrant of $\mathbf{x}$. For instance, $\sigma_{32}(\mathbf{x})$ represents the larger shaded block in Fig. 1 and $\sigma_{300}(\mathbf{x})$, the smaller shaded block.

Let the set of quadrant addresses be given by Q^* where $Q^* = \bigcup_{m=0}^{\infty} Q^m$ with $Q^0 = \{\epsilon\}$ and $Q^m = \{a_1 \cdots a_m : a_1, ..., a_m \in Q\}$ for $m \geq 1$. If we define a *multiresolution function* Γ as a mapping $\Gamma : Q^* \to \mathbb{R}$ satisfying the area-preserving condition $\Gamma(\omega) = \frac{1}{|Q|} \sum_{a \in Q} \Gamma(\omega a)$, where $|Q|$ is the cardinality of Q and $\mathbb{R}$ denotes the set of all real numbers, then each multiresolution image $\rho(\mathbf{x})$ corresponds to a unique multiresolution function $\Gamma_{\mathbf{x}} : Q^* \to \mathbb{R}$ defined by $\Gamma_{\mathbf{x}}(\omega) = \sigma_\omega(\mathbf{x})$. In fact all images of finite resolution will be automatically promoted to multiresolution images. For simplicity we shall always denote $\sigma_\omega(\mathbf{x})$ for any $\omega \in Q^*$ by $\mathbf{x}^\omega$ when no confusion is to occur, typically when $\mathbf{x}$ is in bold face, and denote by $\mu(\mathbf{x})$ the average intensity of the image $\mathbf{x}$.

There exist so far two forms of WFA that have been applied to image compression. The first was adopted by Culik et al [2,5] while the second was due to Hafner [3,4]. For convenience, Culik et al's WFA will be referred to as a *linear* WFA while Hafner's (m)-WFA will be termed a *hierarchic* WFA due to its hierarchic nature. We note that a linear WFA may also be regarded as a generalised stochastic automaton [6], and can be used to represent real functions [7] and wavelet coefficients [8], to process images [9], as well as to recognise speech [10].

In a WFA based image compressor, the design of an *inference algorithm* plays an important role. An inference algorithm will typically partition a given image into a collection of *range blocks*, i.e. subimages of the original given image, then find proper *domain blocks* to best approximate the corresponding range blocks. In general, the availability of a large number of domain blocks will result in fewer number states when encoding a given image. This will in turn lead to better compression performance. Some initial but inconclusive attempts to engage more domain blocks have already been made in [11] for bi-level images. The extra blocks there were in the form of certain transformations such as some special rotations, or in the form of some unions of such domain blocks. Due to the lack of theoretical justification there, it is not obvious whether and what extra image blocks can be *legitimately* added to the pool of domain images. Hence one of the main objectives of this work is to provide a sound theoretical background on which one may readily decide how *additional* domain images may be artificially generated by the existing ones, resulting in a similarity enriched pool of domain images.

This paper is organised as follows. We first in the next section propose and study two types of important mappings that are indispensible to a proper extension of WFA. They are the resolution-wise and resolution-driven mappings, and are formulated in practically the broadest possible sense. Section 3 then sets out to the actual extension of WFA, its associated graph representation as well as the unification of the existing different forms of WFA. A number of relevant properties and methods, such as the determination of minimum states for an extended WFA, have also been established. Finally in section 4, the benefits of the proposed similarity enrichment are demonstrated on the compression of the Lena image through a complete codec based on a specific extended WFA.

We note that the important results in this work are often summarised in the form of theorems. Although all proofs are fairly short, they are not explicitly presented here due to the lack of space. Interested readers however may consult [12] for further details.

2 Resolution-wise and Resolution-Driven Mappings

A fundamental question related to the generalisation of WFA is to find what kind of similarity enrichment will preserve the original WFA's crucial features that underpin their usefulness in the image compression. The resolution-driven and resolution-wise mappings are therefore specifically designed for this purpose and for their broadness. A *resolution-driven mapping* f is a mapping that maps a finite sequence of multiresolution images $\{\mathbf{x}_i\}$ into another multiresolution image $\mathbf{x}$, satisfying the following invariance condition: for any $k \in \mathbb{N}$, if $\mathbf{x}_i$ and $\mathbf{y}_i$ are indistinguishable at the resolution of $2^k \times 2^k$ pixels for all i, then image $\mathbf{x} = f(\{\mathbf{x}_i\})$ and image $\mathbf{y} = f(\{\mathbf{y}_i\})$ are also indistinguishable at the same resolution. A resolution-driven mapping is said to be a *resolution-wise* mapping if the resulting image $\mathbf{x}$ at any given resolution is completely determined through f by the images $\{\mathbf{x}_i\}$ at the *same* resolution.

Let $\mathfrak{F}$ and $\mathfrak{G}$ be the set of resolution-wise mappings and the set of resolution-driven mappings respectively, and let $\mathfrak{I}$ be the set of all *invertible* operators in $\mathfrak{F}$. We will always use the addressing scheme (1) associated with $Q = \{0, 1, 2, 3\}$ unless otherwise stated. Then the following theorem gives a rich family of resolution-wise mappings.

Theorem 1. *Suppose a mapping $\mathfrak{h}$ permutes pixels at all resolutions. That is, for any multiresolution image $\mathbf{x}$, the multiresolution image $\mathfrak{h}(\mathbf{x})$ at any resolution is a pixel permutation of $\mathbf{x}$ at the same resolution. If every addressible block in the form of $\mathbf{x}^\omega$ at any resolution will always be mapped onto an addressible block $\mathfrak{h}(\mathbf{x})^{\omega'}$ for some $\omega' \in Q^*$ in the mapped image, then the mapping $\mathfrak{h}$ is an invertible resolution-wise mapping, i.e. $\mathfrak{h} \in \mathfrak{I}$.*

A straightforward corollary is that $\mathfrak{I}$ contains the horizontal mirror mapping $\mathfrak{m}$, the diagonal mirror mapping $\mathfrak{d}$, the anticlockwise rotation $\mathfrak{r}$ by 90°, as well as the permutation of the $|Q|$ quadrants. Obviously $\mathfrak{I} \subset \mathfrak{F} \subset \mathfrak{G}$, identity mapping $\mathbf{1} \in \mathfrak{I}$, and $\mathfrak{I}$ is a group with respect to the "$\cdot$" product, i.e. the composition of functions. Moreover $\mathfrak{F}$ and $\mathfrak{G}$ are both linear spaces. We also note that, for any $\omega \in Q^*$, the zoom-in operator $\mathfrak{z}^\omega$ defined by $\mathfrak{z}^\omega(\mathbf{x}) = \mathbf{x}^\omega$ for all $\mathbf{x}$ belongs to $\mathfrak{G}$.

To understand more about $\mathfrak{I}$, we first introduce $\mathbb{P}$, the group of all permutations of Q's elements. A typical permutation $p \in \mathbb{P}$ can be written as $p = \begin{pmatrix} 0 & 1 & 2 & 3 \\ n_0 & n_1 & n_2 & n_3 \end{pmatrix}$ which means symbol $i \in Q$ will be permuted to symbol $n_i \in Q$. As a convention, every permutation of k elements can be represented by a product of some disjoint cycles of the form $(n_1, n_2, ..., n_m)$, implying that n_1 will be permuted to n_2 and n_2 will be permuted to n_3 and so on and that n_m will be permuted back to n_1. Any elements that are not in any cycles will thus not be permuted at all. Hence for instance the permutation $\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 \\ 4 & 3 & 2 & 1 & 5 & 0 \end{pmatrix}$ can be written as $(1, 3)(0, 4, 5)$, or equivalently $(0, 4, 5)(1, 3)(2)$ because the "(2)" there is redundant.

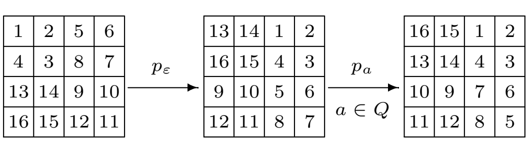

Fig. 2. A resolution-wise mapping in action

Let $\mathfrak{p} = \{p_\omega\}_{\omega \in Q^*}$ be a sequence of permutations $p_\omega \in \mathbb{P}$, then $\mathfrak{p}$ represents an invertible resolution-wise mapping in the following sense: for any multiresolution image $\mathbf{x}$, the mapped multiresolution image $\mathfrak{p}(\mathbf{x})$ at any resolution of $2^k \times 2^k$ pixels is defined by, for $m = 0$ to $k - 1$ in sequence, permuting the $|Q|$ quadrants of the subimage $\mathbf{x}^\omega$ according to p_ω for all $\omega \in Q^m$. The horizontal mirror mapping $\mathfrak{m}$ thus corresponds to $\{p_\omega\}$ with $p_\omega = (0, 2)(1, 3)$ for *all* $\omega \in Q^*$. As another example, let $\mathfrak{p} = \{p_\omega\}$ with $p_\varepsilon = (0, 1, 3, 2)$, $p_0 = (0, 2)(1, 3)$, $p_1 = (0, 1)(2, 3)$, $p_2 = (0, 3)$, $p_3 = (1, 2)$ and $p_\omega = (0)$ for all $|\omega| > 1$. Then the mapping process of $\mathfrak{p}$ is shown in Fig. 2 in which the leftmost image is gradually transformed to the rightmost resulting image. Additional properties are summarised in the next theorem.

Theorem 2. *Let* $\mathfrak{P} = \bigcup_{m=0}^{\infty} \mathfrak{P}_m$ *and* $\mathfrak{P}_m = \left\{ \mathfrak{p} \in \mathfrak{I} : \mathfrak{p} = \{p_\omega\}_{\omega \in Q^*} \text{ such that } p_{\omega'} = p_{\omega''} \ \forall \omega', \omega'' \in Q^{m'} \text{ with } m' \geq m \right\}$, *and let* m *be any non-negative integer. Then*

(i) $\mathfrak{P}_m \subset \mathfrak{P}_{m+1} \subset \mathfrak{P} \subset \mathfrak{I} \subset \mathfrak{F} \subset \mathfrak{G}$.
(ii) *For any* $\mathfrak{p} \in \mathfrak{P}_m$*, its inverse,* $\mathfrak{p}^{-1}$*, is also in* $\mathfrak{P}_m$.
(iii) *For any* $\mathfrak{p}, \mathfrak{q} \in \mathfrak{P}_m$*, the composition* $\mathfrak{p} \cdot \mathfrak{q} \equiv \mathfrak{p}(\mathfrak{q})$ *is again in* $\mathfrak{P}_m$.
(iv) $\mathfrak{P}_m$ *is a finite group under the mapping composition.*
(v) *For any* $\mathfrak{p} = \{p_\omega\}_{\omega \in Q^*} \in \mathfrak{P}_m$ *and any zoom-in operator* $\mathfrak{z}^a$ *with* $a \in Q$*, we set* $a' = p_\varepsilon(a) \in Q$*,* $\mathfrak{q} = \{p_{a'\omega}\}_{\omega \in Q^*}$*, and* $\mathfrak{P}_{-1} \equiv \mathfrak{P}_0$ *in the case of* $m = 0$*. Then* $\mathfrak{q} \in \mathfrak{P}_{m-1}$ *and* $\mathfrak{z}^{a'}\mathfrak{p} = \mathfrak{q}\mathfrak{z}^a$*. Moreover the mapping* $a \to a'$ *on* Q *is one-to-one and onto.*

For easy reference, a resolution-wise mapping will be called *permutative* if it belongs to $\mathfrak{P}$. Fig. 3 exemplifies some of the typical operators in $\mathfrak{G}$: $\mathfrak{m}$ is the horizontal mirror, $\mathfrak{d}$ is the diagonal mirror, $\mathfrak{r}$ is an anticlockwise rotation, $(0, 3)$ is a quadrant permutation/exchange and $\mathfrak{z}^3$ is a zoom-in operation into quadrant 3. In fact, $\mathfrak{m}$, $\mathfrak{d}$ and $\mathfrak{r}$ all belong to $\mathfrak{P}_0$. Moreover we can conclude from the properties of $\mathfrak{G}$ that the image negation is for instance also a resolution-wise mapping. We note that the function space $\mathfrak{F}$ is of great importance as is manifested in the following theorem.

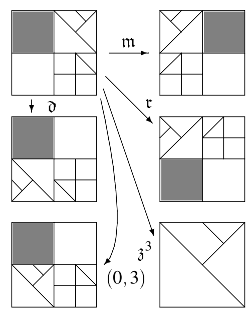

Fig. 3. Some typical resolution-driven mappings

Theorem 3. *Let* $f_i^a \in \mathfrak{F}$ *and* $\beta_i \in \mathbb{R}$ *for all* $i \in \mathcal{I}$ *and* $a \in Q$*. If*

$$\sum_{q \in Q} f_i^a(\{\beta_j\}) = |Q| \cdot \beta_i \,, \qquad i \in \mathcal{I} \,, \tag{2}$$

then the images $\mathbf{x}_i^{(k)}$ *of* $2^k \times 2^k$ *pixel resolution determined iteratively by*

$$\begin{aligned} &\mathbf{x}_i^{(0)} = \beta_i \,, \quad \left[\mathbf{x}_i^{(1)}\right]^a = f_i^a(\{\mathbf{x}_j^{(0)}\}) \,, \quad \cdots \\ &\left[\mathbf{x}_i^{(m)}\right]^a = f_i^a(\{\mathbf{x}_j^{(m-1)}\}) \,, \quad \left[\mathbf{x}_i^{(m+1)}\right]^a = f_i^a(\{\mathbf{x}_j^{(m)}\}) \,, \quad \cdots \end{aligned} \tag{3}$$

uniquely define for each $i \in \mathcal{I}$ *a multiresolution image* $\mathbf{x}_i$ *of intensity* $\mu(\mathbf{x}_i) = \beta_i$*. The representation of* $\mathbf{x}_i$ *at the resolution of* $2^k \times 2^k$ *pixels is given by* $\mathbf{x}_i^{(k)}$ *precisely for all* $i \in \mathcal{I}$*. Moreover the* $\mathbf{x}_i$ *will satisfy*

$$\mathbf{x}_i^a = f_i^a(\{\mathbf{x}_j\}) \,, \qquad f_i^a \in \mathfrak{F} \,, \quad i \in \mathcal{I} \,. \tag{4}$$

Conversely if (4) holds for all $i \in \mathcal{I}$ *, then (2) must be valid for* $\beta_i = \mu(\mathbf{x}_i)$.

In the case that corresponds eventually to a linear WFA, the results can be represented in a very concise matrix, see [12] for details. To conclude this section, we note that the symbols such as $\mathfrak{P}_m$, $\mathfrak{I}$, $\mathfrak{F}$, $\mathfrak{G}$ and $\mathfrak{z}^a$ will always carry their current definitions, and that resolution-driven mappings can be regarded as the similarity generating or enriching operators.

3 Extended Weighted Finite Automata

The objective here is to show how resolution-wise or resolution-driven mappings come into play for the representation, and hence the compression, of images. We shall first propose an extended form of WFA and its graph representation, and then devise a compression scheme based on such an extended WFA. In fact we shall establish a broad and unified theory on the WFA for the purpose of image compression, including the creation of extended WFA of minimum number of states.

3.1 Definition and Graph Representation of Extended WFA

Suppose a finite sequence of multiresolution images $\{\mathsf{x}_k\}_{1\le k\le n}$ satisfy

$$\mathsf{x}_i^a = f_i^a(\mathsf{x}_1, ..., \mathsf{x}_r)\,,\ \mathsf{x}_j = g_j(\mathsf{x}_1, ..., \mathsf{x}_{j-1})\,,\quad f_i^a \in \mathfrak{F}\,,\ g_j \in \mathfrak{G},$$

for $1 \le i \le r$ and $r < j \le n$. Then all images x_k can be constructed at any resolution from the image intensities $\beta_k = \mu(\mathsf{x}_k)$. For all practical purposes we however consider the following canonical form

$$\mathsf{x}_i^a = \sum_{k=1}^{r} f_{i,k}^a(\mathsf{x}_k)\,,\ \mathsf{x}_j = \sum_{k=1}^{j-1} g_{j,k}(\mathsf{x}_k)\,,\ \ a \in Q\,,\ 0 \le q < i \le r\,,\ r < j \le n\,. \quad (5)$$

We first represent (5) by an extended WFA of n states. This extended WFA will consist of (i) a finite set of states, $S = \{1, 2, ..., n\}$; (ii) a finite alphabet Q; (iii) a target state; (iv) a final distribution $\beta_1, ..., \beta_n \in \mathbb{R}$; (v) a set of weight functions $f_{i,k}^a \in \mathfrak{F}$ and $g_{j,\ell} \in \mathfrak{G}$ for $a \in Q$, $q < k \le r$, $r < j \le n$ and $1 \le \ell < j$. The first q states will be called the *pump states*, and the next $r - q$ states and the last $n - r$ states will be called the *generic states* and the *relay states* respectively. For each state $k \in S$, image x_k is the corresponding *state image* although a more convenient but slightly abusive approach is to identify a state simply with the state image. The *target image* is the state image corresponding to the target state. In general the pump states refer to some preselected base images, and these base images at any resoltions may be generated, or pumped out, independent of the other state images. As an example, we consider the following extended WFA with $Q = \{0, 1, 2, 3\}$, $r = n = 5$, $q = 0$ and

$$\begin{array}{llll}
\mathsf{x}_1^0 = \mathsf{x}_3\,, & \mathsf{x}_2^0 = \mathsf{x}_4\,, & \mathsf{x}_3^0 = \mathsf{x}_5\,, & \mathsf{x}_4^0 = \mathsf{x}_4 + \mathfrak{d}(\mathsf{x}_4)\,, \\
\mathsf{x}_1^1 = \mathsf{x}_2\,, & \mathsf{x}_2^1 = \mathsf{x}_2\,, & \mathsf{x}_3^1 = \frac{1}{2}\mathsf{x}_3 + \frac{1}{2}\mathsf{x}_4, & \mathsf{x}_4^1 = \mathsf{x}_4\,, \\
\mathsf{x}_1^2 = \mathsf{x}_1\,, & \mathsf{x}_2^2 = \mathsf{x}_2\,, & \mathsf{x}_3^2 = \frac{1}{2}\mathsf{x}_3\,, & \mathsf{x}_4^2 = \mathsf{x}_4\,, \\
\mathsf{x}_1^3 = \mathfrak{d}(\mathsf{x}_1), & \mathsf{x}_2^3 = 0\,, & \mathsf{x}_3^3 = 0\,, & \mathsf{x}_4^3 = 0\,, \\
\mathsf{x}_5^0 = \frac{1}{2}\mathsf{x}_5\,, & \mathsf{x}_5^1 = \frac{1}{2}\mathsf{x}_5 + \frac{1}{4}\mathsf{x}_4 + \frac{1}{4}\mathfrak{d}(\mathsf{x}_4)\,, & & \\
\mathsf{x}_5^2 = \frac{1}{2}\mathsf{x}_5\,, & \mathsf{x}_5^3 = \frac{1}{2}\mathsf{x}_5 + \frac{1}{4}\mathsf{x}_4 + \frac{1}{4}\mathfrak{d}(\mathsf{x}_4)\,, & &
\end{array} \quad (6)$$

where $\mathfrak{d}$ denotes the diagonal reflection, see Fig. 3. Let $\beta_i = \mu(\mathbf{x}_i)$ for $i = 1, ..., 5$. Then (6) defines a valid WFA (5) if, due to (2), $(\beta_1, \beta_2, \beta_3, \beta_4, \beta_5) = \left(\frac{5}{12}, \frac{1}{2}, \frac{1}{3}, 1, \frac{1}{2}\right)\xi$ for some $\xi \in \mathbb{R}$. The diagram for the WFA, with $\mathbf{x}_1$ as the target image, is drawn in Fig. 4. It is not difficult to see that the WFA will generate the target image $\mathbf{x}_1$ given in Fig. 5. We note that, in a graph representation, $i : \beta_i$ inside a circle represents state i with intensity β_i, and label $a(w)$ or $a : \mathfrak{h}(w)$ on an arrow from state i to state j implies $\mathbf{x}_i^a$ contains linearly $w\mathbf{x}_j$ or $w\mathfrak{h}(\mathbf{x}_j)$ respectively, where w is the weight. Moreover the doubly circled state denotes the target state.

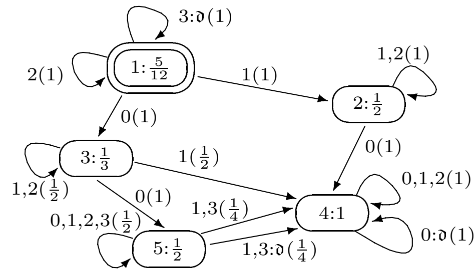

Fig. 4. WFA of 5 states for (6)

We note that the target image determined by the extended WFA Fig. 4 can also be generated by a linear WFA, at the cost of having more than doubled number of states. The linear WFA is given in Fig. 5 in which each state is now labelled by the corresponding state image and the image intensities $\beta_i = \mu(\mathbf{x}_i)$ are given by $\beta_1 = \beta_7 = \frac{5}{12}$, $\beta_2 = \beta_5 = \beta_8 = \beta_{11} = \frac{1}{2}$, $\beta_3 = \beta_9 = \frac{1}{3}$, $\beta_4 = \beta_{10} = 1$ and $\beta_6 = 2$. Incidentally, if we keep just the first 6 states in Fig. 5 and assign $\beta^T = (5/36, 1/4, 1/6, 1/2, 1/4, 1)\xi$ to the final distribution β in Theorem 3, then the equations in (4) again define a valid WFA with 6 states and 23 edges. The target image $\mathbf{x}_1$ in this case reduces to the diminishing triangles by Culik and Kari [5], which is just the image $\mathbf{x}_1$ in Fig. 5 without the details above the diagonal.

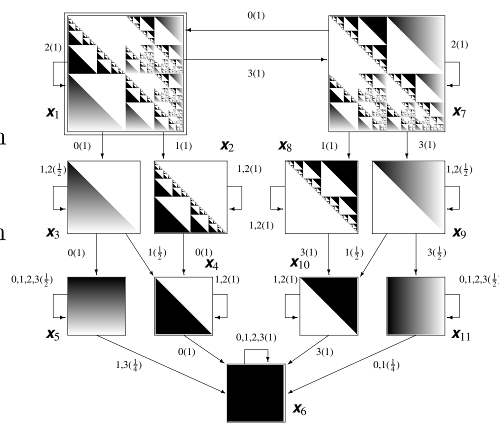

Fig. 5. Transition diagram of a WFA for the target image determined by (6)

3.2 Packed WFA and the Associated Properties

For the practicality of applications, it is of interest to restrict ourselves to the following special yet sufficiently general form of WFA relations

$$\mathbf{x}_i^a = \sum_{k=1}^{n} \sum_{\tau \in \mathfrak{R}} W_{i,k}^{a,\tau} \tau(\mathbf{x}_k)\,, \ \mathbf{x} = \sum_{k=1}^{n} \sum_{\tau \in \mathfrak{R}} \alpha_k^{\tau} \tau(\mathbf{x}_k)\,, \ a \in Q\,, \ 0 \le q < i \le n\,, \quad (7)$$

where $W_{i,k}^{a,\tau}, \alpha_k^{\tau} \in \mathbb{R}$ and $\mathfrak{R} \subset \mathfrak{F}$ is a nonempty finite set of resolution-wise mappings. The target image $\mathbf{x}$ corresponds to the only relay state, implicitly the state $n+1$. However $\mathbf{x}$ may be absent if the target state is just one of the generic states. An extended WFA defined through (7) will be called a *packed* WFA, and the WFA is also said to be *packed under* $\mathfrak{R}$.

To analyse properly the properties of packed WFA, we first introduce a feature that refines a set of resolution-driven mappings. More specifically, a set $\mathfrak{R} \subset \mathfrak{I}$ is said to be *quadrantly-supportive* if all mappings in $\mathfrak{R}$ are linear and that for all $\tau \in \mathfrak{R}$ and $a \in Q$, there exist $\tau' \in \mathfrak{R}$ and $a' \in Q$ such that $\mathfrak{z}^a \tau = \tau' \mathfrak{z}^{a'}$, i.e. $\left[\tau(\mathbf{x})\right]^a = \tau'(\mathbf{x}^{a'})$ holds for all multiresolution images $\mathbf{x}$. As an example, $\mathfrak{P}$ and $\mathfrak{P}_m$ for $m \geq 0$ are all quadrantly-supportive. With these preparations, we now summarise some additional finer properties for packed WFA in the next two theorems.

Theorem 4. *A packed WFA under a finite group* $\mathfrak{R}$ *can always be expanded into a linear WFA if* $\mathfrak{R}$ *is also quadrantly-supportive. In particular, a WFA packed under a finite subset of* $\mathfrak{P}$ *can always be expanded into a linear WFA.*

Alternatively, a packed WFA can be regarded as a linear WFA reduced or packed under the similarities or symmetries dictated by $\mathfrak{R}$. We note that symmetries or similarities are immensely useful in various applications, including for example the group theoretical analysis by Segman and Zeevi in [13], the image processing and understanding in [14], and the solutions and classification of differential and difference systems in [15,16].

Given a linear WFA and a set $\mathfrak{R}$ of resolution-wise mappings, we may reduce the linear WFA to a packed WFA of fewer number of states. For this purpose we say a set of images $\{\mathbf{x}_i\}$ are $\mathfrak{R}$*-dependent* if there exists an i_0 such that $\mathbf{x}_{i_0} = \sum_{i \neq i_0}^{\text{finite}} \sum_{\tau \in \mathfrak{R}}^{\text{finite}} c_i^\tau \tau(\mathbf{x}_i)$ for some constants $c_i^\tau \in \mathbb{R}$. It is easy to see that $\mathfrak{R}$-independent images are also linearly independent if $\mathbf{1} \in \mathfrak{R}$. Moreover one can show for a WFA packed under a finite group $\mathfrak{R} \subset \mathfrak{I}$, if the generic state images are $\mathfrak{R}$-dependent, then the WFA can be shrunk by at least 1 generic state.

Naturally it is desirable to have a WFA with as few states as possible while not incurring too many additional edges. For any given image there is a lower bound on the number of states a packed WFA can have if the WFA is to regenerate the given image.

Theorem 5. *Suppose* $\mathfrak{R} \subset \mathfrak{I}$ *is a finite group and is quadrantly-supportive, and that a packed WFA* $\mathfrak{A}$ *under* $\mathfrak{R}$ *has a target image* $\mathbf{x}$ *and has no pump states. Let* $\{\mathbf{x}_i\}_{i=1}^n$ *be all the state images of the WFA. Then* $\mathfrak{A}$ *has the minimum number of states, the minimum required to reconstruct the target image* $\mathbf{x}$*, if and only if*

(i) *all the state images* $\mathbf{x}_i$ *are* $\mathfrak{R}$*-independent;*

(ii) *all state images are combinations of addressible subimages of* $\mathbf{x}$*, i.e. all state images are of the form*

$$\sum_{\tau \in \mathfrak{R}} \sum_{\omega \in Q^*}^{\textit{finite}} c_{\tau,\omega} \tau(\mathbf{x}^\omega)\,, \quad c_{\tau,\omega} \in \mathbb{R}\,;$$

(iii) *target image* $\mathbf{x}$ *corresponds to a generic state.*

To conclude this section, we note that the set $\mathfrak{R}$ practically never exceeds the scope of $\mathfrak{P}$ in the application to image compression. This means that all assumptions for the above two theorems are in fact automatically satisfied if $\mathfrak{R}$ is just a finite group. As far as WFA based image compression is concerned, $\mathfrak{R}$ doesn't even have to be a group: it suffices to be just a finite subset of $\mathfrak{F}$. We also remark that the two apparently different formulations of WFA due to Culik et al and Hafner respectively can be easily unified: for each hierarchic WFA due to Hafner we can struct explicitly a linear WFA adopted by Culik et al, but not vice versa. The details are again available in [12].

4 Implementation of a Packed WFA and the Resulting Compression Improvement

The purpose here is to show, through an implemented image compressor based on the theory developed in the previous sections, that similarity enriched WFA are indeed more desirable and beneficial to the image compression. For clarity and simplicity of the actual implementation, we do not consider any pump states here and will use only three resolution-wise operators. They are respectively the horizontal, vertical and diagonal reflections.

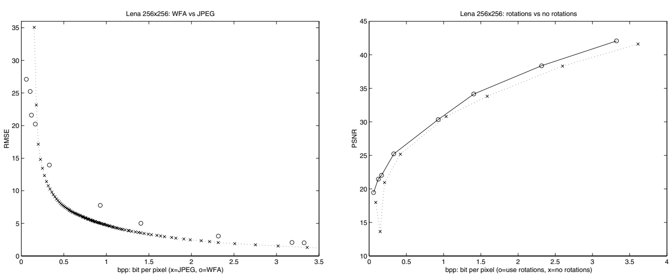

Fig. 6. Extended WFA vs JPEG and vs linear WFA

Hence our compressor is in fact built on the simplistic treatment of the auxiliary techniques associated with the compression scheme based on an extended WFA. For instance no entropy coding is performed after the quantisation of the WFA weights. Despite these simplifications, the compression performance is still on a par with the JPEG standard on natural images: slightly better than JPEG at very low bitrates but otherwise marginally worse than JPEG in the present simplistic implementation, see the l.h.s. figure in Fig. 6 for the compression of the standard Lena image at 256×256 pixels in terms of RMSE. The little circles there correspond to the WFA based compressor.

We now compare the compression performance on the Lena image for the linear WFA and that for the packed WFA. First we note that in the encoding of a linear WFA, the storage spaces for the resolution-wise mappings are completely eliminated. Despite this apparent equal footing, a codec based on extended WFA demonstrates a consistently superior compression ratio. In the case of compressing the standard Lena image at 256×256 pixels, for instance, the r.h.s. figure in Fig. 6 illustrates explicitly the better compression performance by the codec based on an extended WFA in terms of PSNR. The curve traced out there by the little circles corresponds to our currently implemented codec based on a packed WFA. The other curve corresponds to the same codec when the similarity enrichment is turned off. We finally note that the above benchmark codec, for simplicity and speed, made use of only a topdown non-recursive inference algorithm. Hence a sizable improvement is naturally expected if a more computation-intensive *recursive* inference algorithm, akin to that adopted in [5] for the *linear* WFA, is instead implemented to the codec. These will however be left to our future pursuit.

References

1. Jiang Z, Wavelet based image compression under minimum norms and vanishing moments, in Yan H, Eades P, Feng D D and Jin J (edrs), Proceedings of PSA Workshop on Visual Information Processing, (1999)1–5.
2. Culik II K and Kari J, Image compression using weighted finite state automata, Computer Graphics **17** (1993)305–313.
3. Hafner U, Refining image compression with weighted finite automata, Proceedings of Data Compression Conference , edrs. Storer J and Cohn M, (1996)359–368.
4. Hafner U, Weighted finite automata for video compression, IEEE Journal on Selected Areas in Communications , edrs. Enami K, Krikelis A and Reed T R, (1998)108–119.
5. Culik II K and Kari J, Inference algorithm for WFA and image compression, in Fisher Y (edr), Fractal Image Compression: theory and application, Springer, New York 1995.
6. Litow B and De Vel O, A recursive GSA acquisition algorithm for image compression, TR **97/2**, Department of Computer Science, James Cook University 1997.
7. Culik II K and Karhumaki, Finite automata computing real functions, SIAM Journal of Computations **23** (1994)789–814.
8. Culik II K, Dube S and Rajcani P, Effective compression of wavelet coefficients for smooth and fractal-like data, Proceedings of Data Compression Conference , edrs. Storer J and Cohn M, (1993)234–243.
9. Culik II K and Fris I, Weighted finite transducers in image processing, Discrete Applied Mathematics **58** (1995)223–237.
10. Pereira F C and Riley M D, Speech recognition by composition of weighted finite automata, At&T Labs, Murray Hill 1996.
11. Culik II K and Valenta V, Finite automata based compression of bi-level and simple color images, presented at the Data Compression Conference, Snowbird, Utah 1996.
12. Jiang Z, Litow B and De Vel O, Theory of packed finite state automata, Technical Report 99-173, University of New England, June 1999.
13. Laine A (edr), Wavelet Theory and Application: A Special Issue of the Journal of Mathematical Imaging and Vision, Kluwer, Boston 1993.
14. Lenz R, Group Theoretical Methods in Image Processing, Lecture Notes in Computer Science **413**, Springer, Berlin 1990.
15. Jiang Z, Lie symmetries and their local determinancy for a class of differential-difference euqations, Physics Letters A **240** (1998)137–143.
16. Jiang Z, The intrinsicality of Lie symmetries of $u_n^{(k)}(t) = F_n(t, u_{n+a}, ..., u_{n+b})$, Journal of Mathematical Analysis and Applications **227** (1998)396–419.