On Digital Images Which Cannot be Generated by Small Generalised Stochastic Automata

Bruce Litow and Olivier de Vel[1]

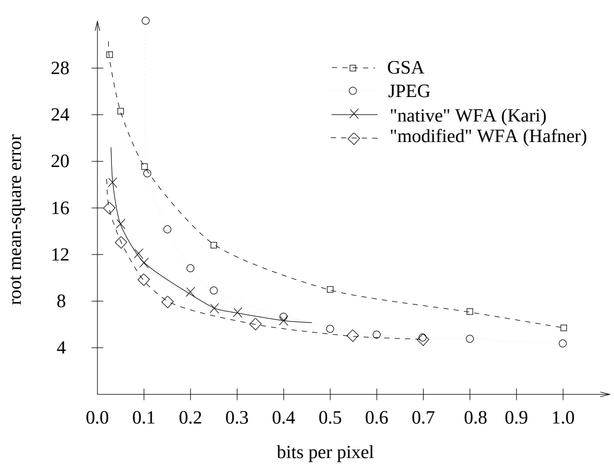
Abstract

It is known, chiefly through the work of Čulik and Kari that generalised stochastic automata (GSA) can be used to compress digital (pixel) images. A theoretical account of GSA-based image compression has not been carried out. This paper contributes to such an account by exhibiting a family of images such that a member image having S pixels cannot be generated by a generalised stochastic automaton having fewer than approximately $S^{1/2}$ states. This lower bound on the number of states holds even when a certain type of loss is permitted. These images are deterministically defined and are based on work of Ablayev.

1 Introduction

Čulik and Kari were the first to observe that digital images could be generated by *generalised stochastic automata* (GSA), and that in some cases lossy compression of an image could be obtained by regarding a GSA as a representation for the image that it 'nearly' reproduces. Čulik and Kari used the term *weighted finite automaton* (WFA), rather than than GSA, but the two terms have the same meaning. GSA have long been studied in connection with formal languages and probabilistic computation, so it seems to be a good idea to retain the term in order to facilitate references to an extensive literature. This paper, and a companion [6] are examples of how the GSA literature can be applied to the study of image compression by GSA.

A number of papers, largely experimental in nature have shown that GSA can yield large compression ratios for some images while incurring acceptable loss. For example, see Hafner [5] or Litow and de Vel [7]. However, principled accounts of the range of applicability of GSA for image compression, and of the quantitative relationship between compression ratio and loss have not yet been carried out. Typical of the experimental data that is presented on behalf of GSA as an image compression method is the following graph.



[1] Bruce Litow and Olivier de Vel,Computer Science, James Cook University, Townsville, Qld. 4811, Australia
bruce,olivier@cs.jcu.edu.au

Figure 1: *Root mean-square error versus compressibility*

The three GSA-based techniques, due to Čulik and Kari [4], Hafner [5] and de Vel and Litow [7] are compared to the JPEG standard. Only one test image ('Lena') at resolution 8 was used and the loss measure is the Euclidean vector norm. The threshold phenomenon at high compression ratios (low bits-per-pixel) which may indicate serious deterioration of image recognisability has not been carefully analysed.

A preliminary analysis of some of the basic questions concerning how well GSA perform in compression is given in [6], but many questions remain. The object of this paper is to continue the theoretical analysis by exhibiting a family of images which cannot be generated by small GSA. More precisely, we show that an image in this family consisting of S pixels cannot be generated by a GSA with fewer than approximately $S^{1/2}$ states. This lower bound holds even under certain types of loss. The image structure that forces a large number of states is entirely deterministic and is adapted from work by Ablayev [1, 2, 3].

We work with two-dimensional digital images, but our method applies to any finite number of dimensions. Let q and n be positive integers. A (q, n)-image P is a formal polynomial in four noncommuting variables $0, 1, 2, 3$ (often called *quadtree coordinates* in computer graphics) and with coefficients from the set $\{0, 1/2^q, 2/2^q, \ldots, 1 - 1/2^q, 1\}$. One can write P as

$$P = \sum_{w \in \{0,1,2,3\}^n} \mu_w \cdot w$$

The monomial $\mu_w \cdot w$ tells us the pixel location in quadtree coordinates at resolution n, i.e., the string $w \in \{0, 1, 2, 3\}^n$, and the pixel value (generalised grey-scale) $\mu_w \in \{0, 1/2^q, \ldots, 1 - 1/2^q, 1\}$. We refer to q as the pixel quantisation level. The formal polynomial approach makes connection with language theory entirely natural. In particular, a $(0, n)$-image, i.e., bi-level, can be identified with the subset of monomials w such that $\mu_w = 1$.

Readers who are unfamiliar with quadtree coordinates may find the following picture and brief explanation helpful.

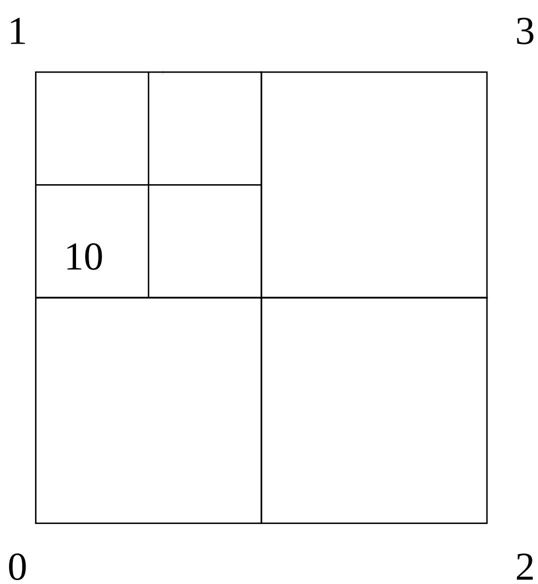


Figure 2: Pixel at resolution two and location 10

The correspondence between quadtree coordinates and Cartesian coordinates is given by the following rule. Let $x_1 x_2 \cdots x_n$ be a resolution n pixel. Write $x_i = y_i \cdot 2 + z_i$, where $y_i, z_i \in \{0, 1\}$. The binary string $y_1 y_2 \cdots y_n$ is the binary representation of an integer, say, Y, and $z_1 \cdots z_n$ is the representation of an integer Z. It is the case that (Y, Z) is the Cartesian coordinate of the lower left-hand corner of the pixel $x_1 \cdots x_n$. This is particularly transparent for the quadtree symbols themselves.

The paper is organised as follows. Section 2 contains definitions and results about GSAs. Section 3 contains a discussion of Ablayev's method and its adaptation to images. Section 4 contains the statement and proof of our main result, Theorem 2. Section 5 discusses whether or not Theorem 2 poses a fundamental limitation on image compression achievable by GSA.

2 GSA theory

We carry through our discussion of basic notion from GSA theory for the quadtree alphabet, but the theory generalises to any finite alphabet. A generalised stochastic automaton (GSA) F is a tuple of the form $F = (u, v, M_0, M_1, M_2, M_3)$.

- u is a $1 \times g$ rational matrix. (row vector)
- v is a $g \times 1$ matrix all of whose entries are in $\{0, 1\}$. (column vector)
- M_0, M_1, M_2, M_3 are $g \times g$ rational matrices.

The value F assigns to the pixel with coordinates $w_1 \cdots w_n$, where $w_i \in \{0, 1, 2, 3\}$ is given by

$$u \cdot M_{w_1} \cdots M_{w_n} \cdot v \ .$$

We let $F(w)$ designate this value. The value of g is the number of states in the GSA.

For image generation, the definition of GSA is modified in that one restricts u to being a $0,1$-matrix, and the entries of v reflect pixel values. These differences are immaterial because we can convert our version of GSA into the standard version by working with the transposes of all the matrices, including the square ones. The only real effect of transposition is to reverse the pixel coordinates. That is if $w \in \{0, 1, 2, 3\}^*$ is the reverse of w', and G is the transposed version of the GSA F, then $G(w') = F(w)$. The main point is that the same generating power is available in both versions.

There is an issue that we do not consider here, although an analysis of it has been given in [6]. What is the quantitative relationship between the precision and range of magnitude of the weights of a GSA and the required pixel quantisation level? Theorem 4 of [6] says, essentially, that the precision of the weights need not exceed by very much the pixel quantisation level q. Notice that the definition of image-generation by a GSA is not entirely satisfactory from the standpoint of the pixel quantisation level because it does not specify how a GSA-generated pixel value is to be assigned to the set of admissible values $\{0, 1/2^q, \ldots, 1\}$. In this paper, we assume that GSA-generated values are admissible.

A stochastic automaton (SA), sometimes called a probabilistic automaton, is a special kind of GSA where the square matrices and u are stochastic (rows sum to 1). If F is a SA, then $F(w)$ is automatically constrained by the stochastic condition to be between 0 and 1. A language

$L \subseteq \{0,1,2,3\}^*$ is said to be λ-stochastic if $L = \{w \mid F(w) > \lambda\}$, where $0 < \lambda < 1$. A language L is said to be $(1/2, \epsilon)$-stochastic if $L = \{w \mid F(w) \geq 1/2 + \epsilon\}$, where $0 < \epsilon < 1/2$. Rabin showed that the $(1/2, \epsilon)$-stochastic languages coincide with the regular languages. See [10].

We establish the fact that it is relatively easy and inexpensive in terms of states to add GSAs.

Lemma 1 *Let α, β be real numbers. let $F = (u, v, M_0, M_1, M_2, M_3)$ and $F' = (u', v', M'_0, M'_1, M'_2, M'_3)$ be GSAs where F has g states and F' has g' states. A GSA G such that $G(w) = \alpha \cdot F(w) + \beta \cdot F'(w)$ can be constructed having $h = 2\max(g, g')$ states. If F and F' are SAs, $0 < \alpha, \beta < 1$ and $\alpha + \beta = 1$, then G can be built as a SA.*

Proof : Let $G = (x, y, H_0, H_1, H_2, H_3)$. Assume $g \geq g'$, and let F have g states.

- For $1 \leq i \leq g$, $x_{1,2i-1} = \alpha \cdot u_{1,i}$, and for $1 \leq i \leq g'$, $x_{1,2i} = \beta \cdot u'_{1,i}$. If $g' < i \leq g$, then $x_{1,2i} = 0$.
- For $1 \leq i \leq g$, $y_{2i-1,1} = v_{i,1}$, and for $1 \leq i \leq g'$, $y_{2i,1} = v'_{i,1}$. If $g' < i \leq g$, then $y_{2i,1} = 0$.
- Let $\gamma \in \{0,1,2,3\}$. H_γ can be thought of as a $g \times g$ matrix whose entries are 2×2 matrices. The 2×2 matrix at row i, column j has zeros in the off-diagonal entries. The upper diagonal entry is $(M_\gamma)_{i,j}$. The lower diagonal entry is $(M'_\gamma)_{i,j}$ if both $i, j \leq g'$, otherwise it is zero.

It is easy to check that $G(w) = \alpha \cdot F(w) + \beta \cdot F'(w)$.

If F and F' are SAs, and α and β are properly constrained, then it is clear that the row vector x is already stochastic. If $g' < g$, then for $i > g'$, make the lower diagonal entry of $(M'_\gamma)_{i,g} = 1$. This ensures that M'_γ is stochastic. This will not have any effect because in this case $y_{2g,1} = 0$.
□

The next result is attributed to Paz [9], but the formulation we use is due to Macarie. See [8] for the proof.

Lemma 2 (Paz-Macarie) *Let F be a GSA, and let a be the maximum over the sums of the absolute values of row entries for its four square matrices, and let b be the sum of the absolute values of the entries of its row vector. There is a SA, F', with at most twice as many states as F such that*

$$F'(w) = \frac{F(w)}{2 \cdot b \cdot a^n} + \frac{1}{2},$$

where n is the length of w.

We point out that it is always true that $|F(w)| \leq ba^n$ for any length n string w. This means that $0 \leq F'(w) \leq 1$.

3 Ablayev's language K_N

We recall the right-congruence relation $x \sim_L x'$ induced on A^* by a language $L \subseteq A^*$. Here A is a finite alphabet. We write $x \sim_L x'$ if for all $y \in A^*$, $xy \in L$ iff $x'y \in L$. The Myhill-Nerode Theorem asserts that L is a regular language iff the number of $\sim_L$-equivalence classes is finite. It is also true that if L is regular, the number of its $\sim_L$-classes coincides with the minimum number of states in any deterministic finite automaton that accepts L. We let $m(L)$ be the number of $\sim_L$-classes of the language L. This material is classical [11].

Let $L \subseteq A^*$ be a regular language and let $x_1, \ldots, x_m$ be representatives of its $\sim_L$-classes. A test set $G \subseteq A^*$ is a set of strings that distinguishes these representatives. That is, for each pair $x_i \neq x_j$ of representatives, there is some $y \in G$ such that exactly one of $x_i y$ or $x_j y$ is in L. Define $\delta(L)$ to be the minimum cardinality of any test set for L. Notice that $\log m \leq \delta(L) \leq m$. We follow Ablayev and define $T(L)$ to be

$$T(L) = \frac{\delta(L)}{\log m}$$

The next result, due to Abalyev provides the basis of our approach. See [2]. The function $\mathcal{H}(p) = -(p \log p + (1-p) \cdot \log(1-p))$, where $0 \leq p \leq 1$, and log designates the binary logarithm.

Theorem 1 (Ablayev) *If L is a regular language with m $\sim_L$-classes, $0 < \lambda < 1/2$, and F is a SA such that*

$$L = \{w \mid F(w) \geq 1/2 + \lambda\} ,$$

then F must have at least

$$m^{1-T(L)\cdot\mathcal{H}(1/2+\lambda)}$$

states.

The language K_N, about to be defined has been discussed in [2]. If $x \in \{0,1\}^*$, let $\|x\|$ designate the integer that is one more than the integer whose binary notation is x. Let $N = n + \lceil \log n \rceil$, and define the language $K_N \subseteq \{0,1,2\}^*$ to be the set of words of the form $w = x2y$, where

- $x \in \{0,1\}^k$ where $1 \leq k \leq n$ and $y \in \{0,1\}^{\lceil \log n \rceil}$.
- x has a 1 in position $\|y\|$.

For example, with $n = 3$ we have $N = 5$ and $010201 \in K_5$, but $010210 \notin K_5$.

Lemma 3 *$m(K_N) \geq 2^n$ and $\delta(K_N) \leq 2n = O(\log m(K_N))$.*

Proof : First we argue that $m(K_N) \geq 2^n$. Let $x, x' \in \{0,1\}^n$ and let x have a 1 and x' have a 0 in position i, respectively. If y is a binary string which is the binary notation for $i-1$, then $x2y \in K_N$ and $x'2y \notin K_N$.

Second, we argue that there is a test set for K_N of size $2n$. This will establish that $\delta(K_N) = O(\log m(K_N))$, since $m(K_N) = 2^{O(n)}$. Define G as

$$G = \{0,1\}^{\lceil \log n \rceil} \cup 2\{0,1\}^{\lceil \log n \rceil}$$

It is straightforward to check that if x is a string for which there exists y such that $xy \in K_N$, and $x' \not\sim_{K_N} x$, then there exists such a y in G. □

4 Proof of the main theorem

Let $\mathcal{K}_N$ designate any (q, N)-image such that every pixel in K_N has a value greater than any pixel not in K_N.

Theorem 2 *For any* $0 < \epsilon < 1$, *and* $n \geq 3$, $\mathcal{K}_N$ *cannot be generated by a GSA with fewer than* $2^{n-\epsilon}$ *states.*

Proof : The idea of the proof is to show that a GSA F that generates $\mathcal{K}_N$ can be converted into a SA, G that has at most eight times as many states as F, and accepts K_N with a probability of at least $1 - \epsilon/n \log n$. By Lemma 3, $T(K_N) = O(1)$, and $m(K_N) \geq 2^n$. Note also that $\mathcal{H}(1 - \epsilon/n \log n)) = O(\epsilon/n)$. These facts, and Theorem 1 show that any SA accepting K_N with a probability of at least $1 - \epsilon/n \log n$ must have at least $2^{n(1-O(\epsilon/n))} = 2^{n-O(\epsilon)}$ states.

Next, we need to ensure that $ba^n \geq 3$. Since $a \geq 1$, we concentrate on rescaling b so that it is at least 3. If $b < 3$, then we replace the GSA F with a GSA R such that $R(w) = F(w)$ and the sum of the absolute values of the row vector of R is at least 3. Let R' and R'' be GSAs identical to F, except that their row vectors are $\alpha \cdot u$ and $(1 - \alpha) \cdot u$, respectively. Here, u is the row vector of F and $\alpha = (3/b + 1)/2$. By Lemma 1, let R be the GSA with twice as many states as F such that $R(w) = R'(w) + R''(w) = F(w)$. However, notice that the sum of the absolute values of the entries of the row vector for R is $(\alpha + \alpha - 1) \cdot b = 3$.

At this stage we can now assume that F is a GSA for which $ba^n \geq 3$. Let f be the minimum of $F(w)$ for $w \in K_N$. Since $\mathcal{K}_N$ is an image, $0 < f \leq 1$. By Lemma 2, there is a SA, F' such that

$$F'(w) = \frac{F(w)}{2 \cdot b \cdot a^n} + 1/2 \ ,$$

where $b > 0$, $a \geq 1$, and a and b can be determined from F. We seek a rational $0 < \alpha < 1$ such that

$$\alpha \cdot (\frac{f}{2 \cdot b \cdot a^n} + 1/2) + (1 - \alpha) > 1 - \epsilon/n \log n \ , \tag{1}$$

and

$$\alpha \cdot (\frac{f - 1/2^q}{2 \cdot b \cdot a^n} + 1/2) + (1 - \alpha) < 1 - \epsilon/n \log n \tag{2}$$

Eq. 1 is the constraint that for pixels $w \in K_N$, the acceptance probability must exceed $1 - \epsilon/n \log n$, and Eq. 2 is the constraint that if $w \notin K_N$, then the acceptance probability is smaller than $1 - \epsilon/n \log n$.

It is straightforward to check that

$$\frac{\epsilon \cdot f}{n \cdot \log n \cdot (1/2 - 1/2ba^n + 1/2^q ba^n)} < \alpha < \frac{\epsilon \cdot f}{n \cdot \log n \cdot (1/2 - 1/2ba^n)} \tag{3}$$

satisfies both Eq. 1 and Eq. 2. Notice that since $ba^n \geq 3$, and $0 < f, \epsilon \leq 1$, we have that $0 < \alpha < 1$ provided $n \geq 3$.

Let F'' be the one-state SA such that $F''(w) = 1$ for all w. By Lemma 1, a SA G can be constructed with twice as many states as F' (so up to eight times as many states as the original F) such that

$$G(w) = \alpha \cdot F'(w) + (1 - \alpha) \cdot F''(w)$$

From the value range of α, we have that the set of strings accepted by G with probability exceeding $1 - \epsilon/n \log n$ is precisely K_N. □

5 Discussion and open problems

Theorem 2 shows that there is a (q, N)-image $\mathcal{K}_N$ ($N = n + \log n$) that cannot be generated by a GSA with fewer than $\Theta(2^n / \log n)$ states. Since the number of pixels is $S = 4^{n+\log n}$, we have shown that at least $\sqrt{S}/8n$ states are needed.

There are two matters that need to be considered in order to assess the significance of Theorem 2 as a limitation on the compression achievable by GSA. The theorem only tells us that *exact* generation of $\mathcal{K}_N$ with S pixels requires approximately $\sqrt{S}$ states. What happens when loss is permitted? Notice that up to this point, the pixel quantisation level has not played an important role. The only type of loss or noise that might make possible a reduction in the number of states below $\sqrt{S}$ is one that sends the value of some pixel in K_N below that of some pixel not in K_N.

Theorem 2 is not an 'asymptotic' result. It applies for practical values of the resolution $N = n + \log n$. For example, if $N = 20$, then $n = 16$, and we get that a $\mathcal{K}_{20}$-image would require a GSA with at least

$$\frac{2^{20}}{8 \cdot \log 16} = 2^{13}$$

states, which is already a very large automaton.

It remains to investigate whether a language other than K_N can be found which yields a lower bound on the number of states of the form $S^{1-\delta}$, where $\delta \ll 1$. Another critical task is to investigate in what ways tolerable forms of loss (and how one might measure loss) help GSA to achieve high compression ratios.

References

[1] F. Ablayev. On comparing probabilistic and deterministic automata complexity and languages. In *Math. Foundations of Comp. Sci. 89*, pages 599–605. Springer LNCS, 1989.

[2] F. Ablayev. Lower bounds for probabilistic space complexity: automata approach. Dept. of Theor. cyber., Kazan U., Russia, 1992.

[3] F. Ablayev. Lower bounds for one-way probabilistic communication complexity and their application to space complexity. *Theor. Comp. Sci.*, 157:139–159, 1996.

[4] K. Culik and J. Kari. Inference algorithms for wfa and image compression. In Y. Fisher, editor, *Fractal Image Compression*, pages 243–258. Springer, 1995.

[5] U. Hafner. Asymmetric coding in (m)-wfa image compression. Technical report, Lehrstuhl fuer Informatik, U. Wuerzburg, 1995. TR 132.

[6] B. Litow and O. de Vel. Generalised stochastic automaton image compression. manuscript, James Cook University Computer Science Dept., 1997.

[7] B. Litow and O. de Vel. A recursive GSA acquisition algorithm for image compression. In *ICML Workshop on automata induction*, 1997.

[8] I. Macarie. Closure properties of stochastic languages. Computer Science Tech. Report 441, University of Rochester, 1993.

[9] A. Paz. *Introduction to Probabilistic Automata.* Academic Press, 1971.

[10] M. Rabin. Probabilistic automata. *Inf. and Control*, 6:230–244, 1963.

[11] M. Sipser. *Introduction to the Theory of Computation.* PWS Publishing Co., 1997.