The University Of Sheffield.

This is a repository copy of *Dynamical Maps Applied to Image Compression*.

White Rose Research Online URL for this paper:
http://eprints.whiterose.ac.uk/79931/

Monograph:
Djirar, S., Banks, S.P. and Williams, K. (1995) Dynamical Maps Applied to Image Compression. Research Report. ACSE Research Report 563 . Department of Automatic Control and Systems Engineering

Reuse
Unless indicated otherwise, fulltext items are protected by copyright with all rights reserved. The copyright exception in section 29 of the Copyright, Designs and Patents Act 1988 allows the making of a single copy solely for the purpose of non-commercial research or private study within the limits of fair dealing. The publisher or other rights-holder may allow further reproduction and re-use of this version - refer to the White Rose Research Online record for this item. Where records identify the publisher as the copyright holder, users can verify any specific terms of use on the publisher's website.

Takedown
If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

White Rose university consortium
Universities of Leeds, Sheffield & York

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk/

629.8 (S) X

Dynamical Maps Applied to Image Compression.

S. Djirar, S.P.Banks and K. Williams

Department of Automatic Control and Systems Engineering

University of Sheffield

Mappin Street

SHEFFIELD S1 4DU

Research Report No 563

January 1995

Abstract

Arnold's ergodic map on the torus is applied as an image compression technique. The method is based on the fact that the map is recurrent. It is given in the discrete case, by $f : \begin{pmatrix} x_{i+1} \\ y_{i+1} \end{pmatrix} = \begin{pmatrix} (x_i + y_i) \; mod \;\; n \\ (x_i + 2y_i) \; mod \;\; n \end{pmatrix}$
(x_i, y_i: integers 0,1, 2, ..., n-1).
This method is very simple compared to the existing methods. We are also interested in the use of cellular automata in the generation of images. The extraction of cellular automata rules using genetic algorithms is compared with direct identification methods.

200291653

SHEFFIELD UNIV. LIBRARY ST. GEORGE'S

1 Introduction Consider the following map (Arnold's map) on a unit two torus T^2 [2].

$$\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} \ mod\ 1 = A \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}$$

The map is area preserving (det A=1) and generates a highly chaotic form of dynamics.

The map can be discretized to a nxn integer lattice. The map is then defined as

$$A : \begin{pmatrix} x_{i+1} \\ y_{i+1} \end{pmatrix} = \begin{pmatrix} (x_i + y_i)\ mod\ n \\ (x_i + 2y_i)\ mod\ n \end{pmatrix}$$

Where (x_i, y_i) are integers 0,1,....,n-1. This map is ergodic and has a maximum period of n^2. The Poincare periodicity or recurrence dependence is presently unknown.(This is simply the number p(n) of iterations for the original 'image' to reappear). The Poincare property has been illustrated on a 230x230 picture of Poincare with the recurence time p(230)=120 [4]. Table 1 shows the periods of a number of different sized lattices. Figure 1 shows the effect of four iterations of the map A.

The recurrence times for non square lattices of size mxn are given in table 2 for various values of m and n.

n	p(n)
256	192
230	120
128	96
85	90
80	60

Table 1: Period of different square lattices.

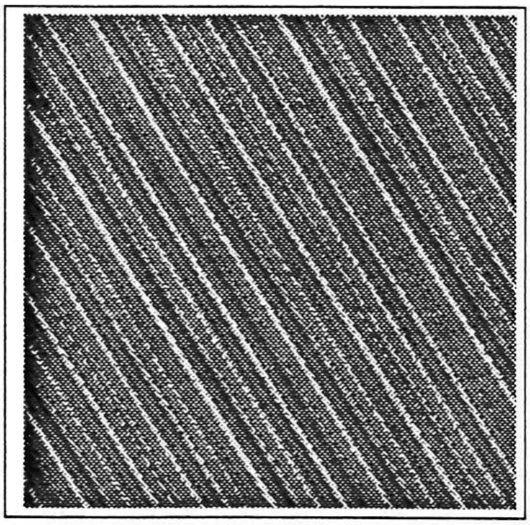

Figure 1:Example of an Image and its iteration (k=4).

m	n	p(m,n)
256	128	192
128	64	96
64	32	48

Table 2: Periods of various nonsquare lattices.

Note that the recurrence times are the same as square images with the larger side. Note also that we can use the map $\overline{A} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$ and we have $A = \overline{A}^2$ so that $p_{\overline{A}}(n) = 2p_A(n)$.

2 Image compression [3] Compression algorithms exist for different types of images. JPEG (Joint Photographic Experts Group) is a Discrete Cosine Transform based algorithm that removes redundant image and colour data. JPEG is intended to provide a wide range of compression ratios (2:1 to 160:1). The (2:1) ratio restores the original image exactly when decompressed but the higher ratios progressively degrade the image quality .

JBIG (Joint Bi-Level Image Experts Group) is another standard intended to complement JPEG for images smaller than six bits per pixel.

The MPEG (Motion Picture Experts Group) compression is used in digital motion video supports compression ratios of up to 50 to 1. It is also based on the Discrete Cosine Transformation DCT [3]. In all these standard forms of compression DCT based algorithms are used. The method proposed in this report is much simpler than these algorithms and is trivial to implement on a computer. The algorithm is also very well suited for parallel processing so that the execution time can be made short.

3 Arnold Maps applied to Image Compression Given an image I of size nxn and p(n) its smallest period, we apply the A map to I and observe the iterated image I_k for different values of k. When k=p(n) the image is the initial one I_0.

The A map is applied to images in following way. Let I(x,y) be the intensity

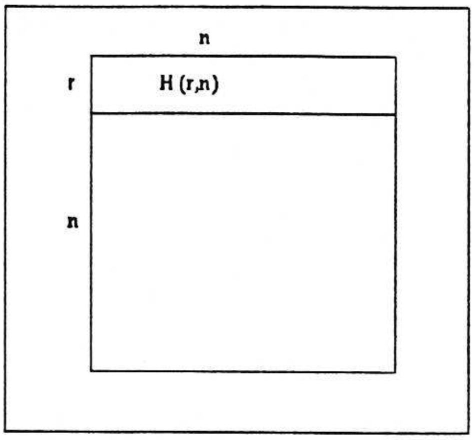

Figure 2:Selected band from the original image.

(grey level) of a pixel at the coordinates x,y of image I. When the map A is applied to image I, the pixel I(x,y) is moved to I(x+y mod n,x+2y mod n). The idea is then to take a horizontal band $H(r,n)$ in figure 2, and copy it periodically together with a shift to produce an image I_l .

When the image I_l is iterated p(n)-k times the image I_R obtained is the same image I_0 with some perturbations figure 3. Using this method we can have different alternative of image compression depending on the intial iteration used. In the following two alternatives are given.

3.1 First alternative k=4. Experiments have shown that when r=21 we obtain the optimal compression ration for k=4 using the A map. In this case the compression is

$$\rho(n) = r * n/n^2$$

so that, when r=21

$$\rho(n) = 21/n \quad .$$

The equation shows that when the size n of picture increase we have a higher compression, for instance $\rho(256) = 0.08$ and $\rho(512) = 0.04$. The method could be used when dealing with large size images.

When n=256 $\rho(256) = 0.08$, we have then a (1:12) compression ratio, which

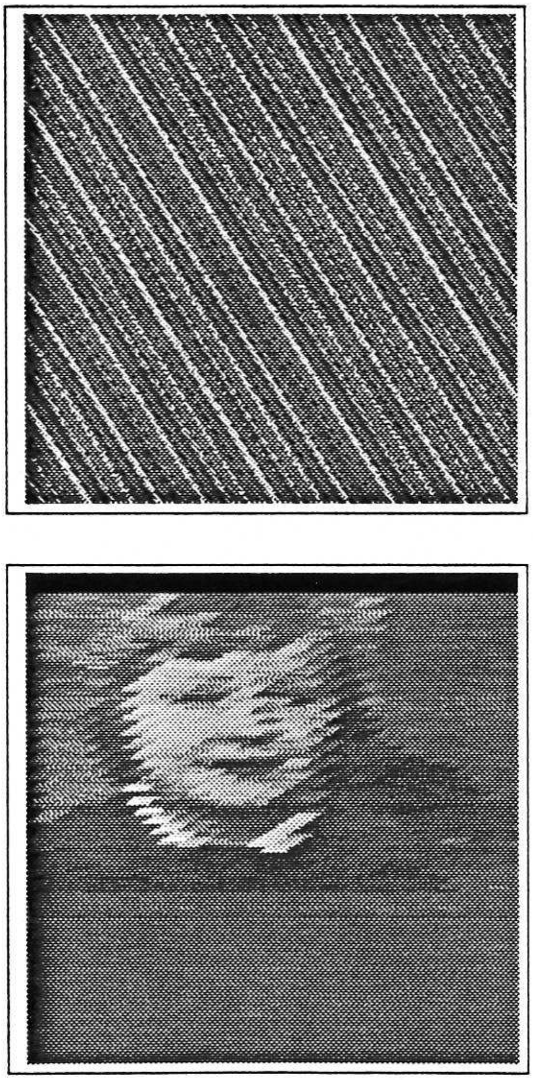

Figure 3:Image I_l and the decompressed image.

Figure 4:Restored image of a 256x256 one, compression ratio was (1:50).

is relatively good considering the simplicity of the algorithm used. Table 2 shows some typical values of n and their corresponding values $\rho(n)$.
A 2% compression can be reached very easily in the following way. An image (nxn) (e.g 256x256) is compressed twice up to (64x64) by simply deleting every other line. The map compression method is then applied to the obtained picture. In this case the ratio is (1:50) figure 4. The quality of the restored image is not particularly good but this example shows the simplicity and the high compression ratio of the method proposed. Figure 4 shows the restored image of the 256x256 picture, the compression ration was (1:50).

3.2 Filtering The reconstructed image suffers some degradation which is manifested as "jagged edges". This phenomenon is caused by the sampling method of the original picture which, in effect takes points along diagonal lines and then copies these points horizontally..
The effect of this is that horizontal changes between points located on subsequent diagonal lines are not recorded and hence high contrast changes are accentuated. To circumvent this effect, the reconstructed image is filtered. Other classical processing methods could be of some benefit. However, the most effective results are obtained by applying a horizontal moving average

Figure 5: Average filtering applied to images in (a) figure 3, (b) figure 4 .

with a window of 7. The size of the window relates to the horizontal distance between the diagonal lines.

This filtering method improves the overall picture. The remaining "jagged edges" can be filtered out by the repeated use of a vertical average with window of 3. The filtered images are shown in figure 5.

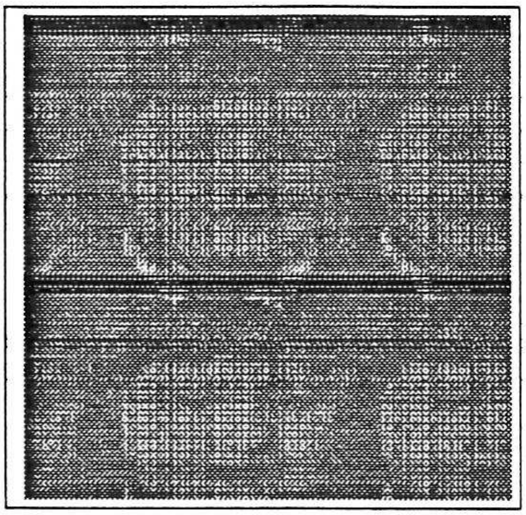

Figure 6:Image of figure 1 at k=96.

3.3 Second alternative From the observation at different iteration k of different images, we noted that at k=96 (n=256), an image composed of four similar parts is obtained as shown in figure 6. If the first quarter is copied then the whole image is iterated l times (l=p(n)-k). The obtained image is the initial one figure 7. The same method can be used for different k. In this case

$$\rho(n) = ((n/2) * (n/2))/n * n \quad .$$

which simplifies to

$$\rho(n) = 0.25$$

4 Rule Extraction The objective in this section is to show how to improve the compression by cellular automata rule extraction. The idea is to extract a cellular automaton rule which will reproduce the band chosen in figure 2 from the first line. This would allow us to have a further compression, since we will need only the initial condition (first line) of the band and the cellular automata rule.

There are mainly two search methods:

1. Direct identification.

Figure 7:The decompressed image of figure 4.

2. Genetic algorithm.

4.1 Direct identification An identification algorithm has been presented in [1]. The method has been tried up to 9-bit rules. The algorithm might find a high order rule which is not efficient in our work since our aim is to compress image data. Hence we fixed our search up to 7-bit rules then we proceed as follows .

Let $R(x_1, ..., x_7)$ denote the rule, for each binary value $(x_1, ..., x_7)$ we count the number of times, within the selected band, for which $(x_1, ..., x_7)$ is mapped to 0, and for which $(x_1, ..., x_7)$ is mapped to 1.

Then, based on the percentage of 1's and 0's for each $x_1, ..., x_7$ we choose the value 0 or 1 for $R(x_1, ..., x_7)$. We obtain in this way the transition table for the best rule. There is, of course, not likely to be a single rule which generates the band exactly.

4.2 Genetic algorithm We use a genetic algorithm to find a cellular automaton rule that would best reproduce the selected band.

In our experiment the cellular automata rules in the population are all 7-bit rules. Thus the bit strings representing the rules are of length $2^7 = 128$ and the size of the the search space is huge. The number of possible cellular au-

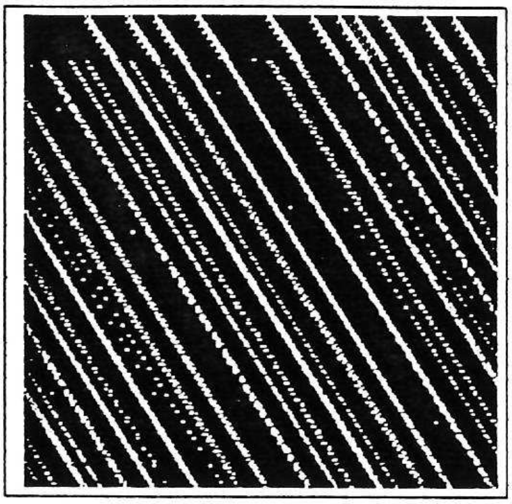

Figure 8: The selected band of figure 2 reproduced by a CA rule.

tomata rules is 2^{128}. The tests for each rule are carried out on configurations of length N=230 with a periodic boundary condition.

The initial population size is 500, generated at random. Proportional fitness has been used for the rules as the fitness function. A best rule obtained is shown in figure 8. Since in this case we are more interested in the global effect of the rule, the genetic algorithm method seems to perform better than the direct one.

5 Conclusion A new image compression method is given. The method is based on the Poincare recurrence property of Arnold's map. The technique is very simple and is easy to implement digitally. Extraction of cellular automata rules to have further compression using genetic algorithms is compared with a direct identification method. Genetic algorithm methods seem to perform better because the selection of the rules is based on the total error.

References

[1] A.I. Adamatskii,"Complexity of Identification of Ca" ,Automation and Remote Control 1992 vol 53 no 59 p 1449-1458.

[2] Atlee Jackson E, "Perspectives of Nonlinear Dynamics",Cambridge University Press,1990, p 80-118.

[3] Nelson L J,"Why Alternatives to JPEG and MPEG Compression Keep on Coming", Advanced Imaging,September 1993, p 108-110.

[4] Crutchfield,J P,Farmer J D, Packard N H, "Chaos",Sci. American, 255(6), p 46-57.

SHEFFIELD UNIV. LIBRARY ST. GEORGE'S