Convergence of Fractal Encoded Images

John Kominek

Department of Computer Science
University of Waterloo
Waterloo, Ontario, Canada
(jmkominek@jeeves.uwaterloo.ca)

1. INTRODUCTION

Because fractal compression is highly asymmetric, most research to date has been devoted to improving compression performance [6,8,9]. Yet, it is the process of decompression that is most intriguing. Whether one uses the 'classical' approach originated by Barnsley [1] or the more practical method of Jacquin [5] that is considered here, an image is represented compactly in terms of an Iterated Function System (IFS). To recover this image the IFS is iterated, with each iteration bringing the process closer to the final result.

To ensure convergence, Jacquin asserts that an IFS must be contractive and establishes a contractivity criterion. Jacobs, Fisher, and Boss found instead that an IFS may converge without being fully contractive, and introduce the idea of *eventual contractivity* [2]. Unfortunately, they provide but a rough guideline as to whether a fractal image will converge. Hurtgen offers a contractivity criterion in [3], but only for the highly restricted case of [7]. This paper examines convergence of fractal encoded images in greater detail.

2. BASICS

2.1 Theory

An IFS, ***W***, is a set of affine transformations that compose a contractive mapping:

$$W = \bigcup_{i=1}^{n} w_{i}, \quad w_i : R^3 \to R^3, \quad \text{where } w_i \text{ are affine transforms} \tag{1}$$

$$d(W(\mu), W(\upsilon)) < s\, d(\mu, \upsilon), \qquad 0 \le s < 1 \tag{2}$$

where μ and υ are any two images of the same size. The metric d measures the difference between two images (commonly the root mean square error). Because s lies in the range [0,1) ***W*** causes any two images to become more alike. It follows that when applied to any input image μ_{any}, the IFS has the following remarkable property:

$$\mu_\infty = \lim_{n\to\infty} (W^n(\mu_{any})) \qquad \text{is a unique attractor,} \tag{3}$$

$$\mu_\infty = W(\mu_\infty) \qquad \text{is a fixed point image.} \tag{4}$$

That is, start with any image and repeatedly apply ***W*** — after a certain stage the process stabilizes to a unique attractor image μ_∞ independent of μ_{any}. More difficult is the reverse: given an arbitrary digital image μ_{given} try to find an IFS which, when evaluated by iteration, produces another image similar to the original. In other words, find a set of transforms $\{w_i\}$ such that $d(\mu_{given}, \mu_\infty)$ is small. This challenge is known as the "inverse problem."

2.2 Practice

In its simplest form, the image to be compressed is partitioned at two scales, one twice the other (e.g. into 8x8 and 4x4 blocks). Using Jacquin's notation, the larger are *domain blocks*, and the smaller ones *range blocks*. The range blocks are non-overlapping and contain every pixel. The domain blocks may overlap and need not contain every pixel.

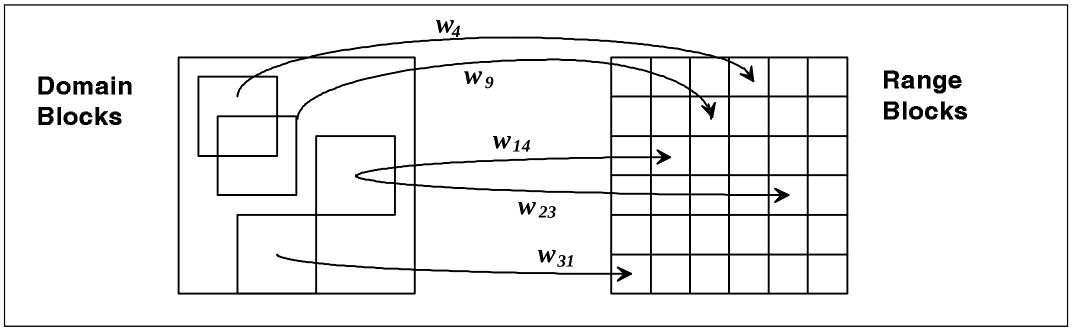


Figure 1. Some domain blocks on the left are mapped onto range blocks on the right.

The goal of the compression process is to find an affine mapping w_i of the form

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = w_i \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & s_i \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \\ o_i \end{bmatrix} \quad (5)$$

for each range block, such that when applied to a domain block the difference between them is small. In (5) the point (x,y) is a domain pixel with a grayscale value z, and (x',y') is a range pixel with new value z'. The coefficients e_i and f_i translate the domain block to the position of the range block as it is shrunk in size by a factor of 2. The coefficient s_i scales the luminance value, while the coefficient o_i introduced a luminance offset.

2.3 Convergence

An analysis of IFS convergence must grapple with the "disparity of scale" problem. Convergence in the theoretical sense is concerned with the image as a whole. Convergence in the practical sense deals with operations on each pixel. When a pixel is considered alone, its value converges to a fixed value on the real number line. Taken together they approach the attractor of (3). This attractor is an *image*, not a geometric point in the ordinary sense, despite being called a "fixed point." Furthermore, the construction of an Iterated Function System (during compression) occurs midway between these two perspectives, at the level of pixel blocks. These perspectives are summarized in Table 1.

Level	Entity	Space	Equation
high	continuous image	$\boldsymbol{R}^2$	(2)
mid	range blocks	$\boldsymbol{R}^{mn}$	(5)
low	individual pixels	$\boldsymbol{R}^1$	(8)

Table 1. Three perspectives on fractal image convergence.

The consequence is that s of (2) does not directly constrain $\{s_i\}$, the set of scale factors that maps domain blocks onto range blocks, as per equation (5). Establishing a link between the two is the purpose of a contractivity criterion. According to Jacquin, $\boldsymbol{W}$ is contractive if

$$s_i \in (-1, 1)\ \forall i \quad \text{or} \quad 0 \le s_{\max} < 1, \text{ where } s_{\max} = \max_{i=1..n}\{|s_i|\} \tag{6}$$

In other words, all individual transforms must be contractive. This is a sufficiency condition and it is easy to proceed from (6) to (2). Is it also necessary? Experimentally, Jacobs *et al.* found that image quality may be improved by raising s_{max}= 2.0 [4]. In the author's experience, this limit can be raised while still retaining convergence. Or, an IFS may diverge with s_{max} only slightly larger than 1.0. The sections that follow investigate the behavior of fractal convergence and attempt to synthesize the three levels of Table 1.

3. OBSERVATIONS

During the compression process a domain block is mapped onto each range block under an affine transformation. The contractivity criterion of equation (6) can be verified experimentally by forcing all scale factors to some fixed value. All IFSs with s_{max} between -1 and +1 converge; those outside this range diverge. For one test image called Bird, a plot of compression error vs. scale value is graphed in Figure 2.

Certainly, if $|s_i|$ is allowed to assume any value between 0 and 1.0 the resulting IFS will be contractive. If s_{max} is raised beyond 1.0 the IFS may still produce a good likeness of the original, as is indicated by the graph of Figure 3. Sample images for s_{max}= { 0.5, 1.0, 2.0} are display in Figure 4, plus a fourth case where s_{min}= 0.5 and s_{max}= 2.0. Here we observe two interesting artifacts. First, when all scale factors are restricted to less than 0.5, steep edges in the image are poorly represented. This is because edges possess a high variance, and s_i < 0.5 effectively eliminates high variance domain blocks. Second, when scale factors are allowed to be as large as 2.0 the Bird image is still evident, but is speckled with pits and spikes, i.e. isolated pixels that are either dark or bright when the surrounding area is grey. This is called speckle impulse noise. Not all IFSs with s_{max} > 1.0 exhibit this kind of defect, but it is always a threat.

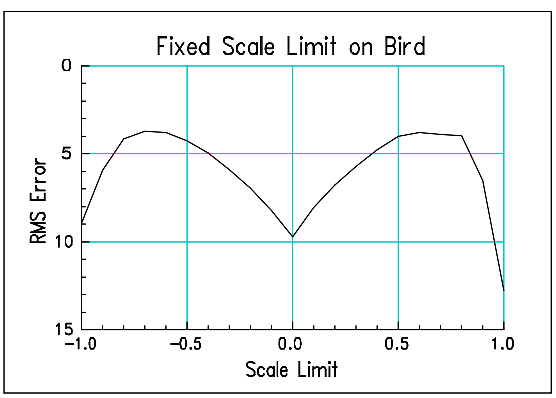


Figure 2. Fractal compression of the Bird image for fixed scale values, i.e. $s_i = s_{fixed}$ for every range block.

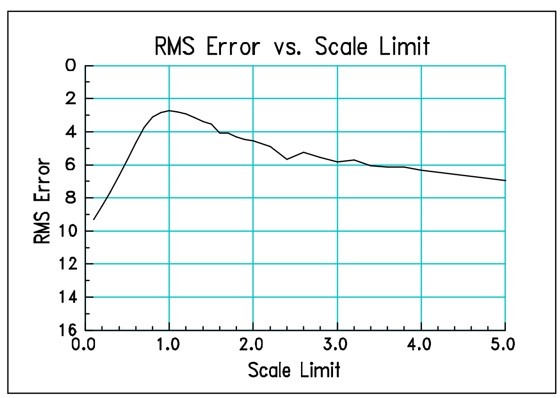


Figure 3. Fractal compression of the Bird image with a fixed ceiling on scale values, i.e. $|s_i| \le s_{\max}$ for every range block.

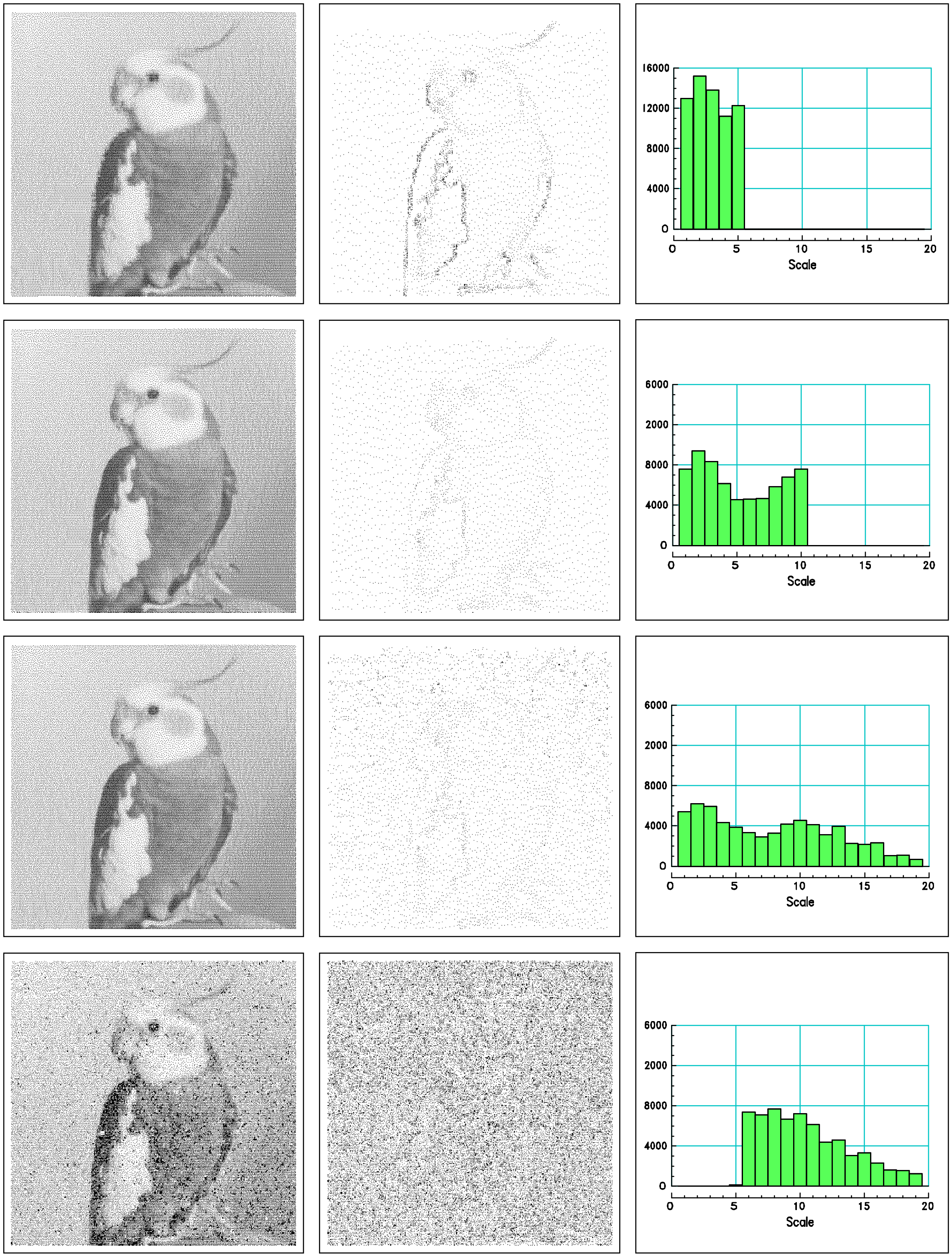


Figure 4. Fractal compression of Bird for different restrictions on the scaling parameter.

(a) $s_{min} = 0.0$, $s_{max} = 0.5$, 6 cycles, partial convergence (1 iteration) = 100.0%

(b) $s_{min} = 0.0$, $s_{max} = 1.0$, 10 cycles, partial convergence (1 iteration) = 100.0%

(c) $s_{min} = 0.0$, $s_{max} = 2.0$, 7 cycles, partial convergence (1 iteration) = 67.2%

(d) $s_{min} = 0.5$, $s_{max} = 2.0$, 21 cycles, partial convergence (1 iteration) = 55.3%

4. ANALYSIS

4.1 Compound Contractivity

When a fractal image is decompressed, the process usually begins by setting the image buffer to some uniform value. This seed value may be black, white, or anything in between. Then, when the IFS is evaluated, each pixel in a particular range block is derived by transforming the corresponding domain block. In Figure 5, the top left corner pixel is derived from the pixel labeled z_1 (currently set to the seed value), by applying the transform $w_{0,1}$. The value at z_1, in turn, is derived by applying $w_{0,2}$ to the pixel labeled z_2. Therefore, after the second iteration, the value at z_0 is a product of both $w_{0,1}$ and $w_{0,2}$. As the iterative process continues, the influence on z_0 extends "backwards," jumping around the image plane in an irregular pattern. This may be termed the "path of influence."

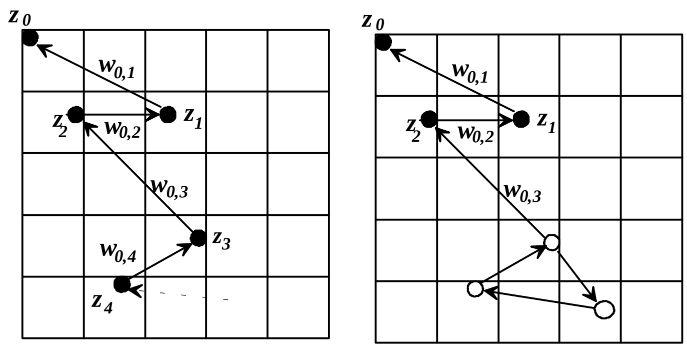


Figure 5. The path of influence on the top left corner pixel, z_0. The symbol $w_{0,n}$ represents the transform that affects z_0 after n iterations. The open circles identify a limit cycle of size three.

Because the affine transformation shrinks a domain block by a factor of 1/2 in each direction, a 2x2 pixel cell maps onto one range block pixel, usually by taking the average. By sampling only one pixel in the cell this computational cost is avoided, though the fidelity of the resulting image may not be as high. For the sake of simplicity the effect of averaging 2x2 pixel cells is not considered here, except in passing.

For this analysis we introduce the notation $z_{p,n}$ to denote the value of the pixel p at some fixed position after n iterations. To avoid an overabundance of subscripts in equation (7) below, s_n and o_n will stand for $s_{p,n}$ and $o_{p,n}$ respectively, the scale and offset of transform $w_{p,n}$. The symbol $w_{p,n}$ represents the affine transform n steps up the path of influence from pixel z_p. Thus, the process depicted in Figure 5 can be expressed as:

$$
\begin{aligned}
z_{0,0} &= z_{seed} \\
z_{0,1} &= o_1 + s_1 z_{1,0} &&= o_1 + s_1 z_{seed} \\
z_{0,2} &= o_1 + s_1(o_2 + s_2 z_{2,0}) &&= o_1 + s_1 o_2 + s_1 s_2 z_{seed} \\
z_{0,3} &= o_1 + s_1 o_2 + s_1 s_2 (o_3 + s_3 z_{3,0}) &&= o_1 + s_1 o_2 + s_1 s_2 o_3 + s_1 s_2 s_3 z_{seed} \\
&\dots \\
z_{0,n} &= o_1 + s_1 o_2 + s_1 s_2 o_3 + \dots + s_1 s_2 \dots s_{n-1} o_n + s_1 s_2 \dots s_n z_{seed}
\end{aligned} \tag{7}
$$

Note that the terms in (7) are ordered from most significant to least significant — in most circumstances the closer a transform is to the current pixel, the greater influence it exerts. The qualifier "in most circumstances" is necessary because the sequence

$$z_{p,n} = \sum_{i=1}^{n} o_{p,i} \left(\prod_{j=1}^{i-1} s_{p,j} \right) + z_{seed} \prod_{i=1}^{n} s_{p,j} \quad \text{converges only if} \quad \prod_{i=1}^{n} s_{p,j} \to 0 \text{ as } n \to \infty. \tag{8}$$

If all scaling factors are set to some fixed value (as is the case in Figure 2), then convergence demands that s_{max} be less than 1.0. When this condition is relaxed the convergence of a particular pixel is still possible, so long as equation (8) holds. In other words, so long as the product of contractive scale factors outweighs the expansive ones, a limit value exists.

Naturally, the IFS is not iterated into infinity, but only until the emergent image stabilizes; between four and eight iterations often suffices in practice. To capture the behavior after a finite number of iterations, we introduce the term *compound contractivity*, a concept that applies to the pixel level, primarily, but may be extended to the entire image.

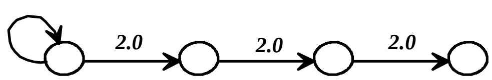
Definition: After n iterations, pixel p has a *compound contractivity* of

$$s_p^n = \left(\prod_{i=1}^{n} |s_{p,j}| \right)^{1/n} \tag{9}$$

In other words, it is the multiplicative average of the sequence of scale factors along the path of influence. The importance of this concept is illustrated with a simple example.

4.2 Simple Example

Suppose that a line of four pixels in a fractal image are related in the manner shown below. The arrows indicate mappings from domain pixels to range pixels, and the numbers above the arrows indicate scale factors. The leftmost pixel maps onto itself — a situation that is a bit unusual, but not forbidden. This mapping is contractive and will, therefore, converge to a fixed value. (The precise number also depends on the offset.) The values of the other three ultimately derives from the leftmost pixel, and, despite having scale factors of 2.0, they too will converge. That is, equation (8) applies. To be specific, the compound contractivity of each pixel evolves as listed. The leftmost pixel is always contractive but the other three are not. Eight iterations are required before the rightmost pixel — and hence entire system — is contractive.



n = 1	0.6	2.0	2.0	2.0
n = 2	0.6	1.095	2.0	2.0
n = 3	0.6	0.896	1.339	2.0
n = 4	0.6	0.811	1.095	1.480

4.3 Eventual Contractivity

As previously mentioned, Jacobs *et al.* observed a similar phenomena in real images. Namely, that an IFS may converge despite being comprised, at least in part, by expansive transforms. To account for this they introduced the following notion.

Definition: An IFS, ***W***, *is eventually contractive* if there exists some positive integer n such that $\boldsymbol{W}^n$ is contractive. The integer n specifies the number of iterations required of ***W*** before the system is contractive.

Note that this definition applies to the IFS in whole — the top level in Table 1 — not to the middle level of individual affine transforms. The connection between the two levels is established by extending equation (9) to the image as a whole.

Definition: After *n* iterations, an IFS, ***W***, has *compound contractivity*

$$s_{\max}^{n} = \max\{s_p^n\} = \max\{\prod_{i=1}^{n} |s_{p,j}|\text{, for all pixels } p \text{ in the image}\} \qquad (10)$$

Therefore, $\boldsymbol{W}^n$ is eventually contractive if $s_{\max}^{n} < 1.0$. It can be easily established that this is both a necessary and sufficient condition.

4.4 Cycle Contractivity

For a particular pixel to converge to some fixed value rather than diverge without limit, its compound contractivity must eventually fall within the range [0, 1). This may happen, or it may not, depending on the dynamics of the IFS. The example in section 4.2 suggests an answer: the contractivity of a pixel is ultimately determined by its *limit cycle*. The limit cycle is that portion of the path of influence that forms a closed loop, i.e. those pixels that are visited more than once.

In Figure 5, the triad of three open circles form the limit cycle of the pixel z_0 (and the others as well). If the limit cycle is contractive, then all pixels that fall within its path of influence will be eventually contractive.

A limit cycle may contain one pixel, or it may traverse the entire image. More typically, an image will contain ten to twenty limit cycles of varying sizes. By application of the "pigeon hole" principle, every image has at least one limit cycle and every pixel belongs to one and only one limit cycle. Therefore, the limit cycle to which a pixel belongs dominates its eventual contractivity (i.e. as $n \to \infty$). The contractivity of a limit cycle, *lc*, is simply equation (10), appropriately restricted.

$$s_{lc}^{n} = \left(\prod_{i=1}^{n} |s_{p,j}|\right)^{1/n}, \; n = \text{size of limit cycle}, \; p \in \{\text{limit cycle pixels}\} \qquad (11)$$

With these concepts in place, identifying a satisfactory contractivity criterion — one that integrates all three levels of Table 1 — is straightforward.

Contractivity Criterion: an IFS is contractive if and only if all the component limit cycles are contractive.

On the other hand, if all of the limit cycles are expansive then the IFS will not converge to an attractor image. If there happens to be a mixture — some contractive and some expansive cycles — then the situation is as in Figures 4c and 4d. A recognizable rendition of the bird exists, but is afflicted with impulse noise. Mathematically speaking the IFS does not converge, but to the human eye an image does emerge, albeit with a random distribution of point-wise defects. This behavior may be termed *partial convergence*.

4.5 Partial Contractivity

Although the limit cycles of an IFS determine its overall contractivity, in practice they are seldom the determining factor. This is because many pixels reside on long branches and may, therefore, require tens of iterations before they "feel" the influence of a limit cycle. If the decompression process is halted after four iterations, say, then the crucial factor is the compound contractivity ($s_p^{\,n}$) when n = 4. Pixels remaining expansive at this stage are susceptible to large fluctuations that may run outside the range [0..255]. If a significant portion of such pixels exist, then the fractal image is prone to speckle impulse noise. This is the situation in Figure 4d. The severity of speckle noise can be measured with the formula in (12).

Definition: After n iterations, an IFS has a *partial contractivity* of

$$\tfrac{1}{MN}\sum_{p=1}^{MN} count(s_p^n), \qquad count(x) = \left\{ \begin{array}{ll} 1, & \text{if } |x| < 1.0 \\ 0, & \text{otherwise} \end{array} \right\} \tag{12}$$

where M and N are the image dimensions, and p enumerates over all pixels.

The partial contractivity as $n \to \infty$ is determined by counting all the pixels that lie in the path of influence of contractive limit cycles. In Table 2, measurement for the images of Figures 4c and 4d are presented for various stages of decompression. The partial contractivity is expressed both as a percentage and as an absolute count of pixels.

Iterations	**Fig 4c**		**Fig 4d**	
1	44,064	67.2%	36,208	55.3%
2	51,880	79.2%	37,248	56.8%
4	58,977	90.0%	39,236	59.9%
8	64,011	97.7%	47,709	72.8%
16	65,401	99.8%	51,005	77.8%
limit	65,442	99.9%	60,246	91.9%

Table 2. The evolution of partial contractivity for the two IFSs that suffer from impulse noise (refer also to Figures 4 and 6). The images are 256x256 pixels in size.

These percentages summarize the evolution of total compound contractivity. As an image, the manner of evolution is visualized in Figure 6. Black represents areas that are expansive and are hence likely to cause impulse noise. Compared to the right column of Figure 6, the more rapid convergence (to white) of the left column correlates well with the numbers in Table 2 and the appearance the bird in Figure 4.

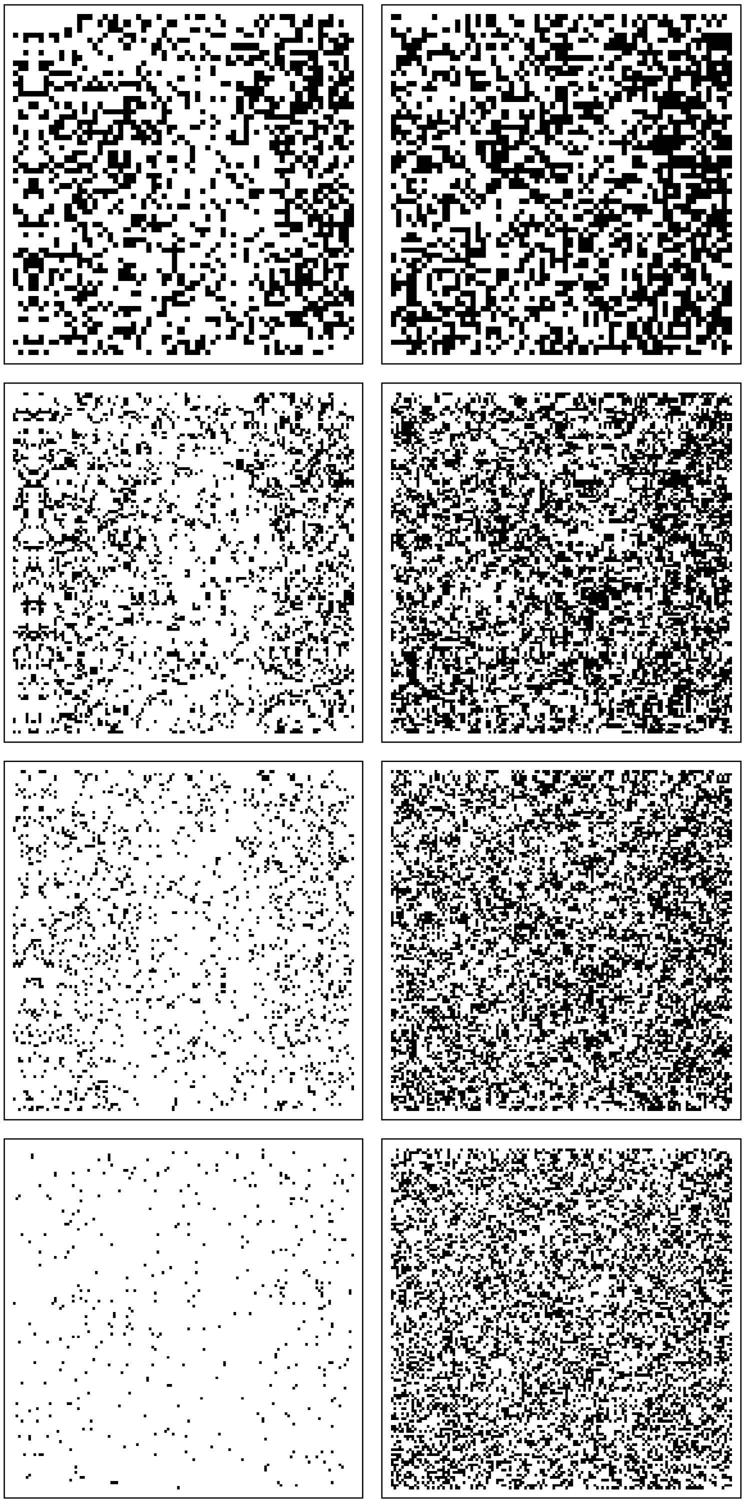
5. CONCLUSION

Fractal image compression, despite its great potential, suffers from some flaws that may prevent its adaptation from becoming more widespread. One such problem is the difficulty of guaranteeing convergence, let alone a specific error tolerance. To help surmount this problem, we have introduced the terms *compound*, *cycle*, and *partial contractivity* — concepts indispensable for understanding convergence of fractal images. Most important, they connect the behavior of individual pixels to the image as a whole, and relate such behavior to the component affine transforms.

For simplicity, certain issues have been left untreated. For example, the IFS has been studied in the discrete domain as an array of pixels, rather than as a continuous function in $\boldsymbol{R}^2$. Most mathematical treatments adopt this latter perspective. And second, the operation of averaging 2x2 pixel cells (when mapping from domain to range blocks), has been omitted. This operation acts to "mix together" four times as may scale factors during each iteration. Interestingly, the rate and quality of convergence is improved. This suggests that convergence may be improved by increasing the relative size of domain to range blocks. Most systems use a 2:1 ration because it works and is computationally efficient. Nonetheless, other possibilities warrant further study.

REFERENCES

1. Michael Barnsley, Alan Sloan; "Methods and Apparatus for Image Compression by Iterated Function Systems," U.S. patent no. 4,941,193, July 10, 1990.
2. Yuval Fisher, Bill Jacobs, Roger Boss; "Iterated Transform Image Compression," NOSC Technical Report 1408, Naval Ocean Systems Center, San Diego, CA, pp. 1-27, April 1991.
3. B. Hurtgen; "Contractivity of Fractal Transforms for Image Coding," *Electronics Letters*, vol. 29, no. 20, pp. 1749-1750, 1993.
4. Bill Jacobs, Yuval Fisher, Roger Boss; "Image Compression: A Study of the Iterated Transform Method," *Signal Processing*, vol. 29, pp. 251-263, 1992.
5. Arnaud Jacquin; "Image Coding Based on a Fractal Theory of Iterated Contractive Image Transformations," *IEEE Transactions on Image Processing*, vol. 1, no. 1, January 1992, pp. 18-30.
6. John Kominek; "Algorithm for Fast Fractal Image Compression," *IS&T Proceedings of the Digital Video Compression Symposium: Algorithms and Technologies 1995*, February 1995.
7. Donald Monro, Frank Dudbridge; "Fractal Block Coding of Images," *Electronics Letters*, vol. 28, no. 11, pp. 1053-1055, 1992.
8. Donald Monro, David Wilson, Jeremy Nicholls; "High Speed Image Coding with the Bath Fractal Transform," *IEEE International Symposium on Multimedia Technologies*, April 1993.
9. Geir Oien, Skjalg Lepsoy, Tor Ramstad; "Reducing the Complexity of a Fractal-Based Image Coder," *Signal Processing VI: Theories and Applications — Proceedings of European Signal Processing Conference (EUSIPCO-92)*, pp. 1353-1356, 1992.



(a) 1 iteration
left: contractivity = 67.2%
right: contractivity = 55.3%

(b) 2 iterations
left: contractivity = 79.2%
right: contractivity = 56.8%

(c) 4 iterations
left: contractivity = 90.0%
right: contractivity = 59.9%

(d) 8 iterations
left: contractivity = 97.7%
right: contractivity = 72.8%

Figure 6. Visualization of partial contractivity. Pixels that are expansive are represented as black. Contractive pixels are represented as white. The left column corresponds to Figure 4c and the right column corresponds to Figure 4d.