

Fractal Compression

- Basic Idea: fixed point transform.
- \underline{X}_0 is a fixed point representation for f if $f(\underline{X}_0) = \underline{X}_0$.
- Transform: $ax+b \rightarrow f$
- Fixed point: $\underline{X}_0 = a\underline{X}_0 + b$

$$\underline{X}_0(1-a) = b$$

$$\underline{X}_0 = \frac{b}{1-a}$$

- Idea: instead of transmitting \underline{X}_0 , transmit a and b .
- Then iterate: $\underline{X}_0^{(n+1)} = a\underline{X}_0^{(n)} + b$
- As $n \rightarrow \infty$ $\underline{X}_0^{(n)} \rightarrow \underline{X}_0$
- One can show this converges regardless of the initial guess.



Image Compression

- Image $I = \text{array of numbers}$
- Find a function f such that $f(I) = I \Rightarrow I$ is a fixed point representation for f
- Approach: Divide the image into a domain block and a range block
 - Divide image into an $M \times M$ “Range block”
 - For each range block find another $2M \times 2M$ domain block from the same image such that for some transform f_k we get $f_k(D_k) = R_k$
- Ref: Jacquin



Image Compression

- Given an f_k and an N range block, for each f_k , $k=1, \dots, N$:

$$f = \bigcup_k f_k$$

$$I \approx f(I)$$

$$\hat{I} \approx I$$

$$\hat{I} = f(\hat{I})$$

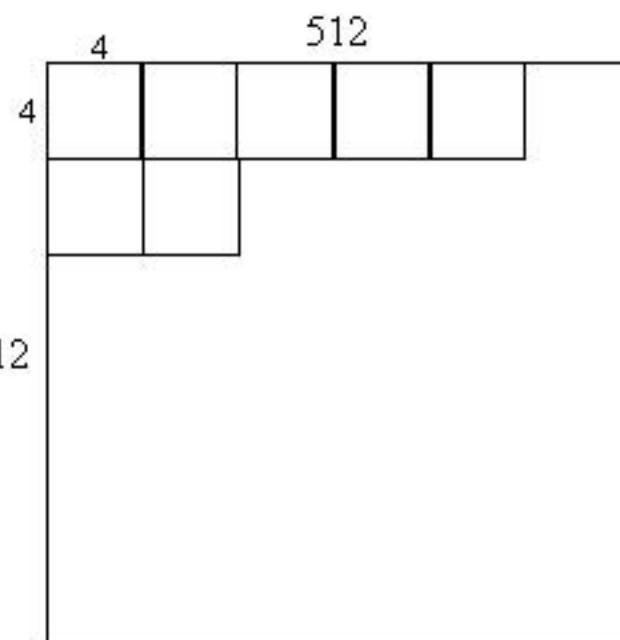
- Collage Theorem: Guarantees convergence to \hat{I} using any arbitrary initial guess for the image.



Vector Quantization

Code Book

B_1	1
B_2	2
B_{19}	19
B_{52}	52
B_{200}	200



Codes

19	B_{19}
52	B_{52}



Vector Quantization

- Let \vec{f} denote an N dimensional vector consisting of N real valued, continuous amplitude scalars.
- Map \vec{f} into L possible N dimensional reconstruction vectors \vec{r}_i for $1 \leq i \leq L$

$$\hat{\vec{f}} = VQ(\vec{f}) = \vec{r}_i, \quad f \in C_i$$

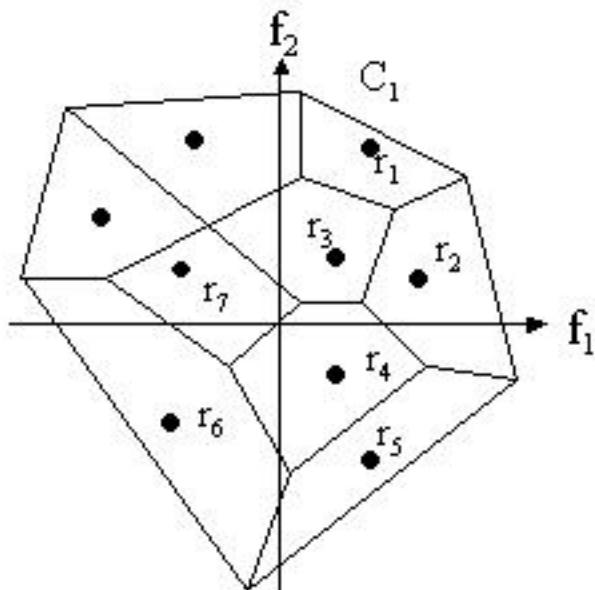


- Advantage of VQ: removes linear dependency of R.V.s



Codebook Design

- $L = 9$ cells, $C_i = i^{\text{th}}$ cell



- Assume $r_1^{(0)}, r_2^{(0)}, \dots, r_3^{(0)}$
- Classify M training vectors into L clusters
- Recompute r_i based on classification of previous step



Complexity Design

- $M = \text{number of training vectors}$
- $L = \text{number of codewords} = 2^{NR}$
- $N = \text{dimension of vectors}$
- $R = \text{bits/scalar}$
- There are MLN operations (adds and mults) per iteration.
 - $N = 10, R = 2, M = 10L \Rightarrow 100 \text{ trillion operations}$
- Note that there are $\frac{\log_2 L}{N} = \text{bits/scalar}$
- Complexity at the receiver:
 - LN operations per vector $\Rightarrow 10 \text{ million operations in this example}$

