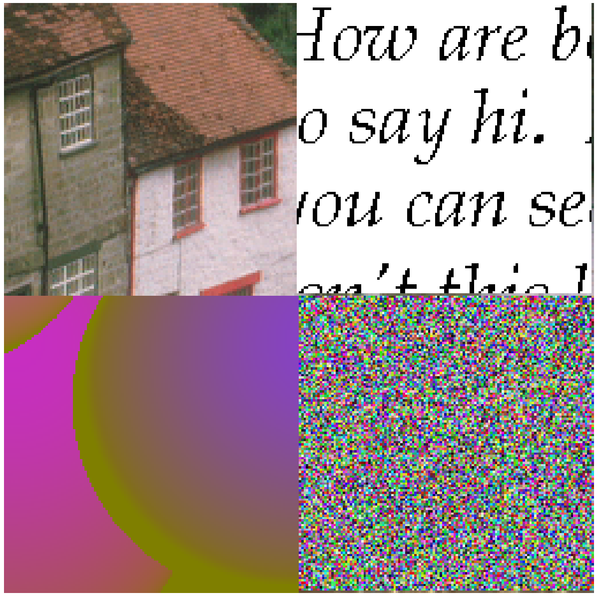
LOCO and JPEG-LS

A new method for lossless and near lossless image compression



Nimrod Peleg
Update: May 2009

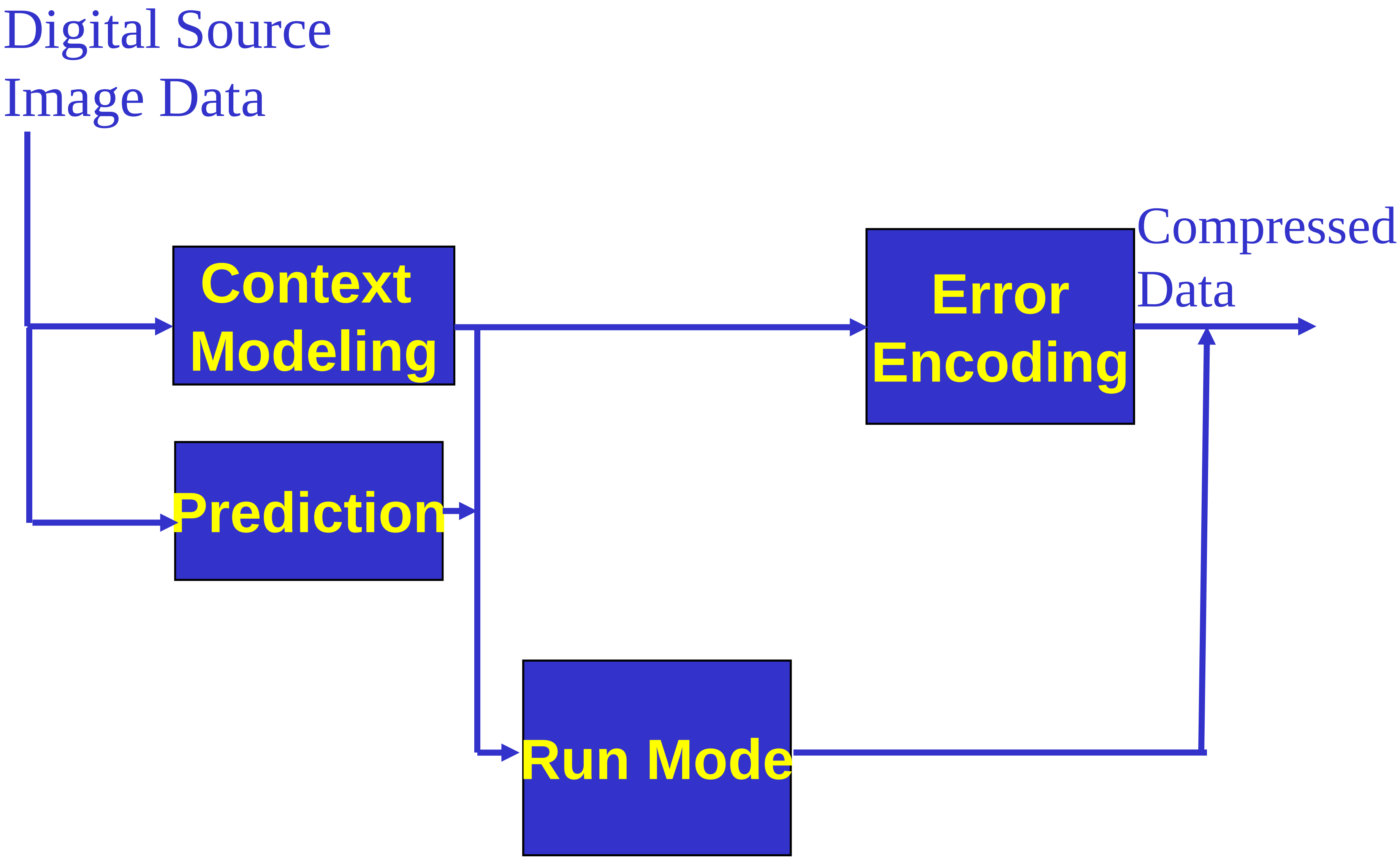
Credit...

- *Suggested by HP Labs, 1996*
- Developed by: M.J. Weinberger, G. Seroussi and G. Sapiro,

"LOCO-I: A Low Complexity, Context-Based, Lossless Image Compression Algorithm",

IEEE Proceedings of Data Compression Conference, pp. 140-149, 1996

General Block Diagram



Context Based Algorithm

- An efficient coding needs a statistical model, for a good prediction of pixel value
- The statistical distribution of a pixel is predicted according to the previous pixels
- The best distribution for coding is minimum entropy distribution
- To achieve it, each pixel is assigned to a "context" (728 context types in LOCO)

Prediction

- After context definition, a simple prediction is activated.
- The predictor is NOT context dependent, but same for all contexts.
- It's a simple and effective edge detector, based on 3 gradients.
- Prediction error is calculated between predicted and "real' pixel value.

Error Coding

- Basic assumption: for each context, a statistical information is available.
- A Geometric distribution is assumed for the prediction error, and LOCO uses 2 parameters for each context: Average and Decay factor.
- LOCO uses a Golomb-Rice entropy coding, which is optimal for two-sided geometric distribution (it's a 1 parameter Huffman like method).

Why Golomb-Rice ?

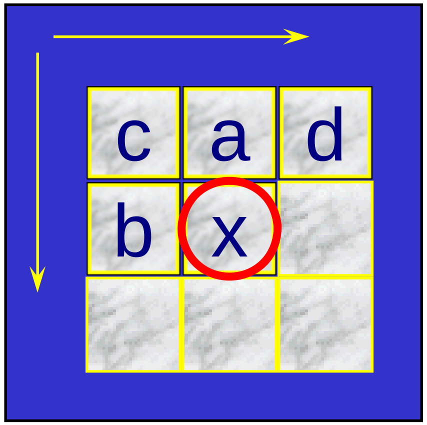
The Golomb-Rice technique is simple, low-complexity and efficient:

- Huffman-like complexity
- Almost Arithmetic coding efficiency

Run Mode

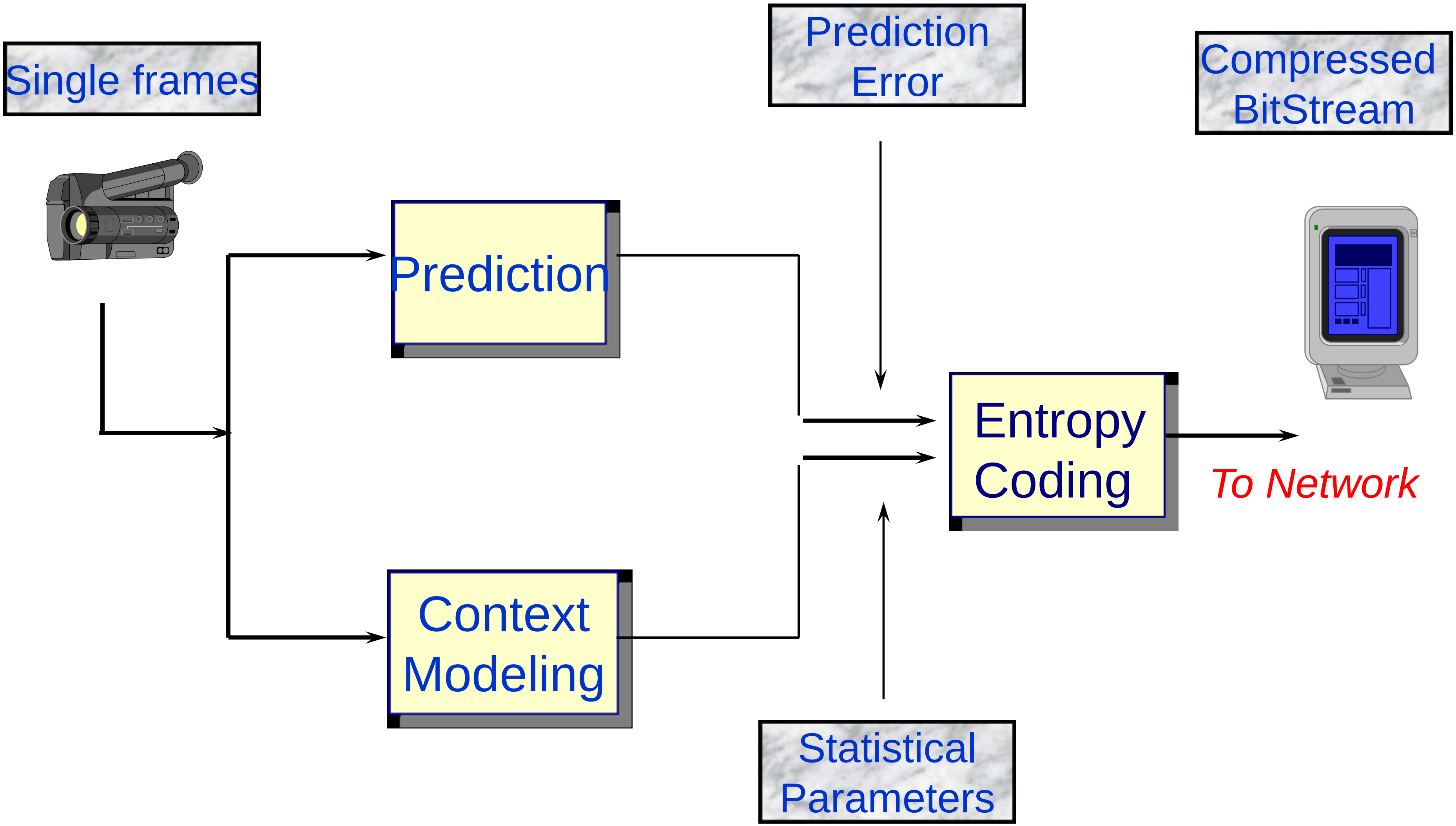
- When all local gradient are zero, we can assume that it's a "smooth" area in the image.
- In this situation we skip the prediction and the prediction error coding stages.
- We go on with the "run-mode" until the condition x=b is no more TRUE.
- This process saves lots of bits

for long runs.



A more detailed Description

General Scheme



Context Modeling

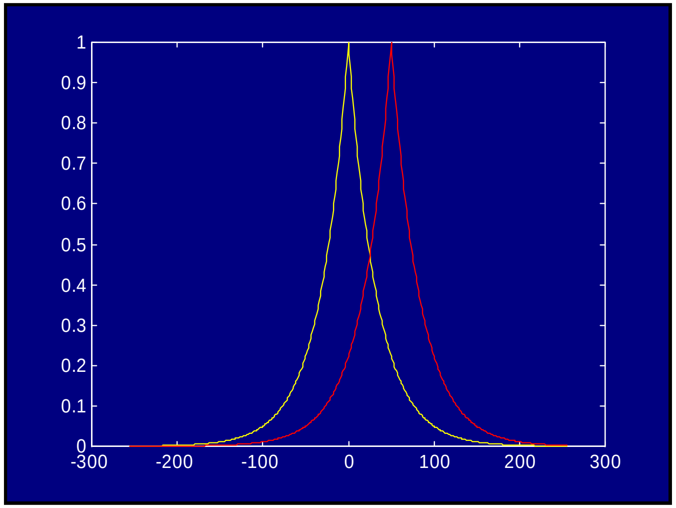
Every pixel is assigned to a context based on locally calculated parameters.

Why ?

Prediction errors for pixels in same context, have the same statistical distribution.

The assumption is -

The prediction error is distributed as a two sided geometric distribution.



- The extinction factor is context dependent.
- There is systematic bias for every context.

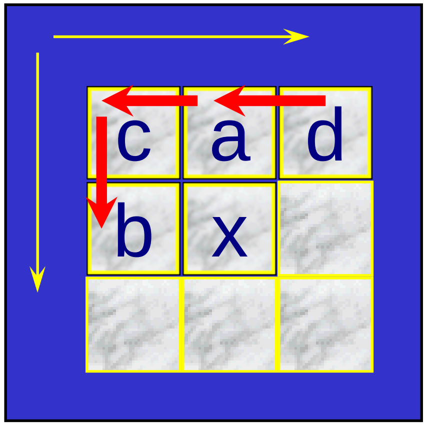
Calculating the context number

There is a <u>tradeoff</u> between the number of contex
and the quality of the estimated statistics
for each context.

In the LOCO algorithm:

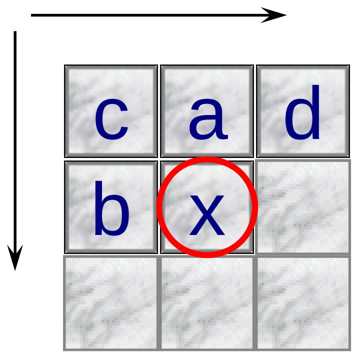
There are 728 contexts created by using the loca
Gradients:

D1= d-a
D2= a-c
D3= c-b



Prediction

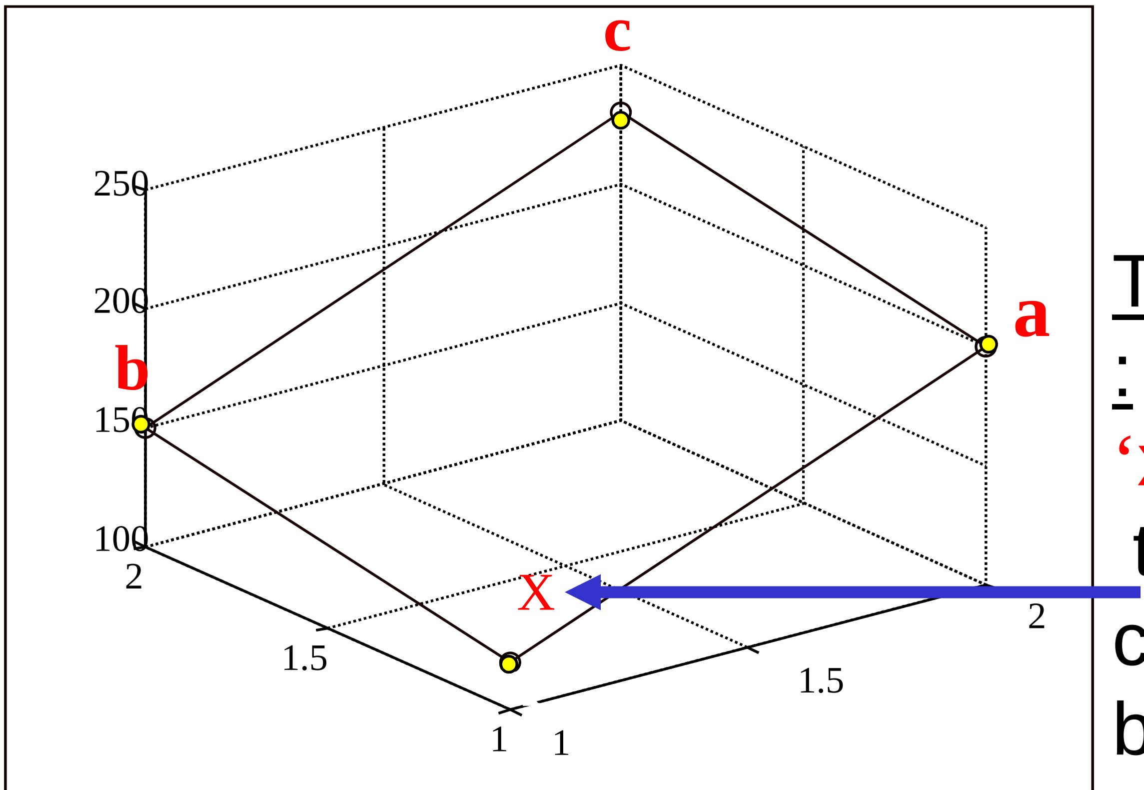
The Loco predictor is "casual", it uses nearby pixels which already have been scanned



$$Px = \begin{cases} \min(a,b) & c \geq \max(a,b) \\ \max(a,b) & c \leq \min(a,b) \\ a+b-c & otherwise \end{cases}$$

Why ???

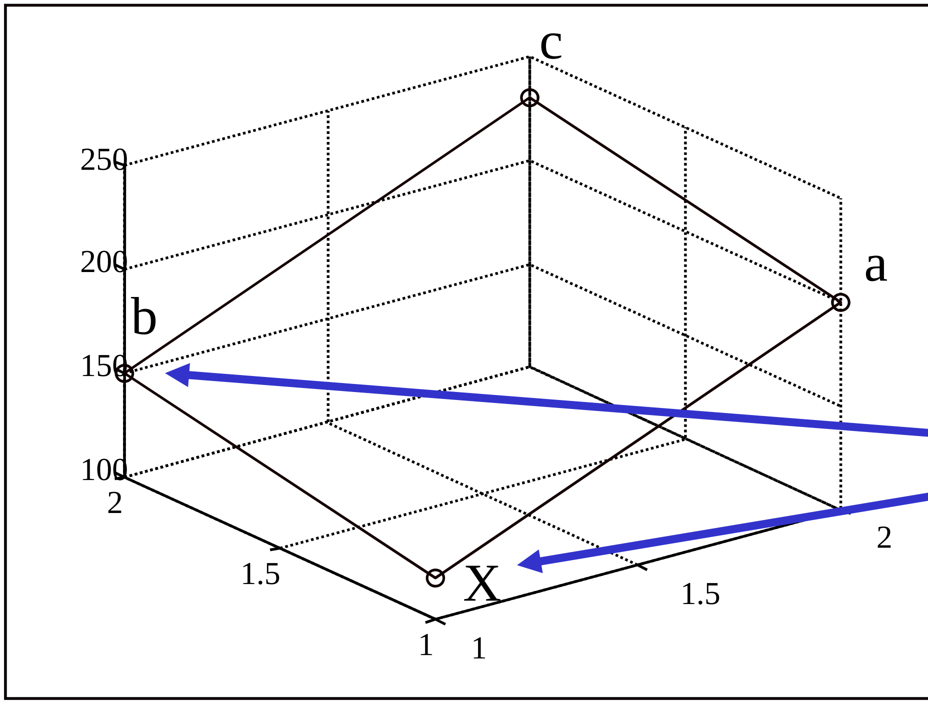
The principle :



The assumption is :

'x' should be on the same plane created by a, b and c .

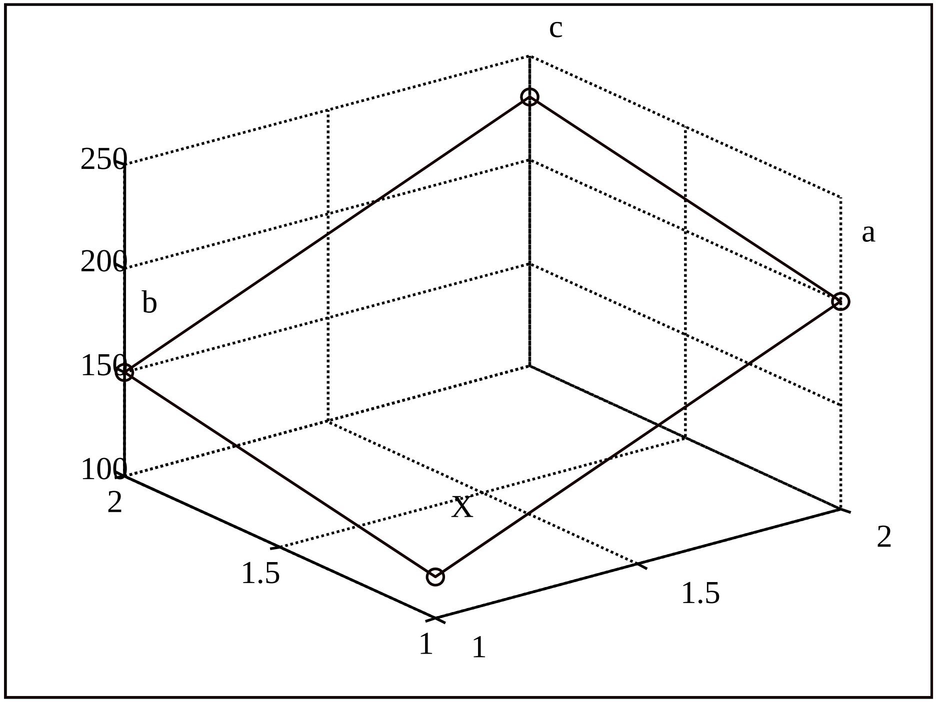
For example :



Here, the lowest value x can get is the value of b (from what we "have").

$$Px = \begin{cases} \min(a,b) & c \geq \max(a,b) \\ \max(a,b) & c \leq \min(a,b) \\ a+b-c & otherwise \end{cases}$$

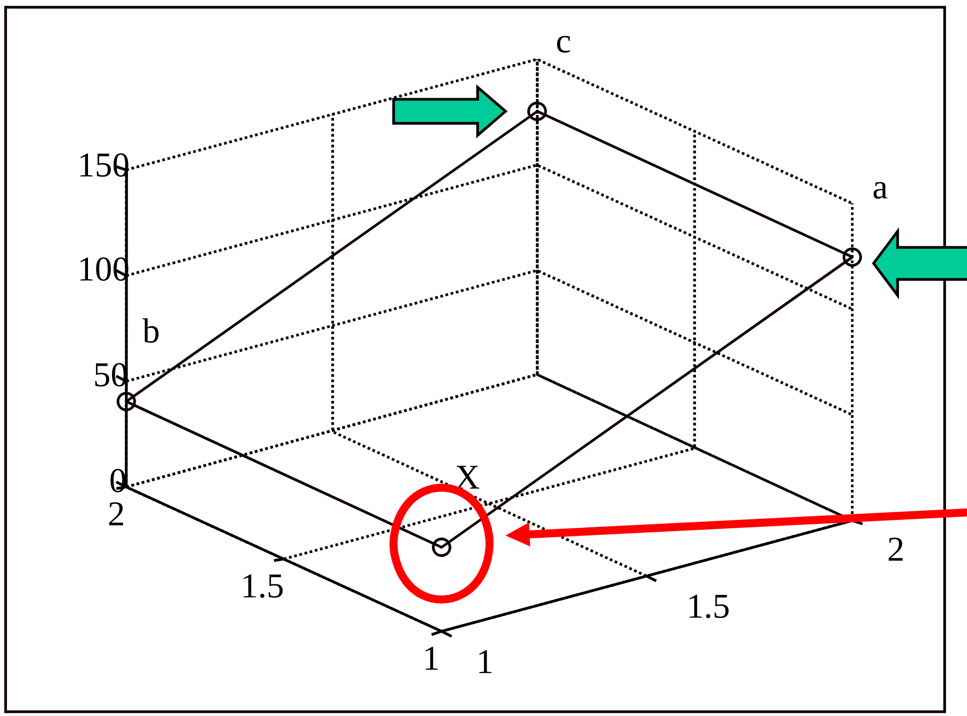
Why not take the Average ?



$$x = \frac{a + b + c}{3}$$

Let's take a look on this case -

When c is very close to a . . .



$$x_{avg} \approx \frac{2a+b}{3}$$

But,

Px = b

Prediction Example

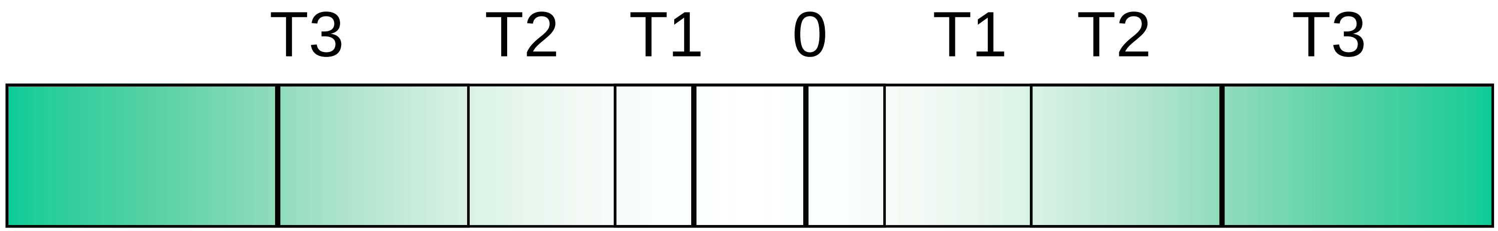
Original Image

Predicted Image (SNR=25.3dB

and more details...

The Quantizer

The gradients are quantized by a 9 level quantizer



(Q1,Q2,Q3)

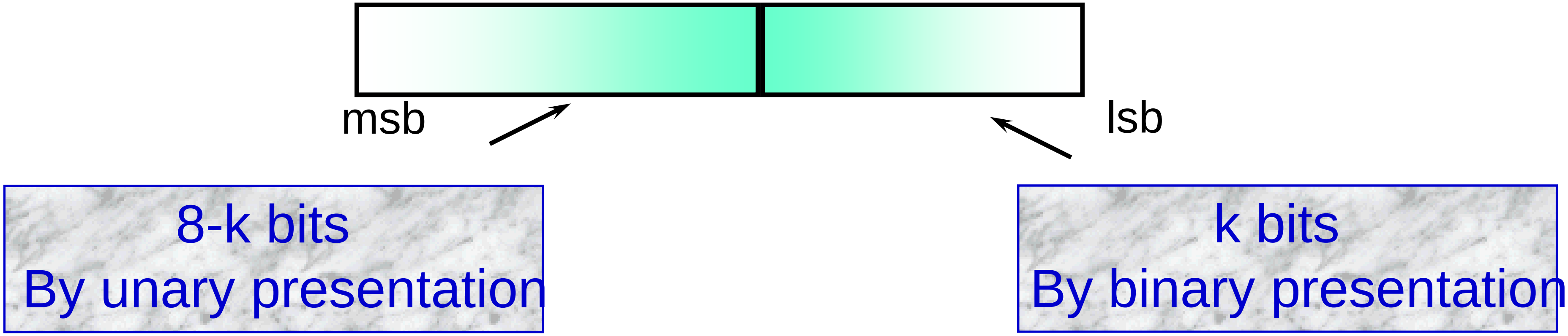
The context number - Q

Entropy Coding

Golomb-Rice is an <u>optimal</u> entropy coder for geometric distribution

Golomb - Rice Codes

- Does not require tables (vs. Huffman)
- One parameter only: *k*



Golomb-Rice coding example

k = 2

Let’s take the number: *x=14* → ***0000 1110***

Unary presentation of 11_2 (= 3_{10}) = $0\ 0\ 0_1$

Binary presentation - 1 0

Unary Code	**0 0 0**
Separating one	**1**
Binary Code	**1 0**

The BitStream

Run mode

A special mode which allows efficient compression of a sequence of pixels with <u>same value</u>.

For example

Lossy Mode

In Lossy Mode

Maximal allowed restoration error per pixel = ~~0~~ *NEAR*

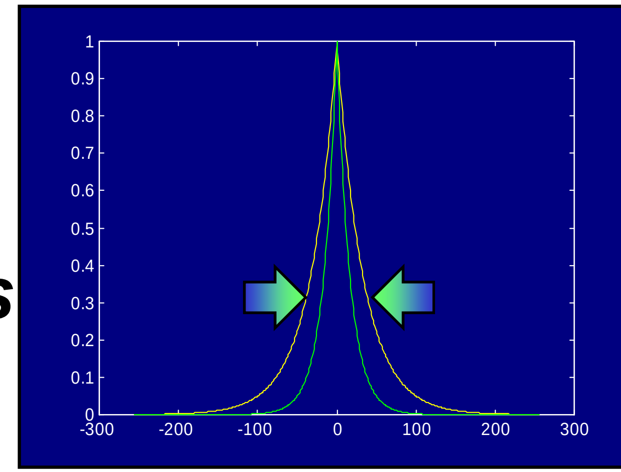
The method :

The prediction error is quantized by a *2Near + 1* step quantizer .

$$\text{Prediction Error} \Rightarrow \left\lfloor \frac{\text{Prediction Error}}{2Near + 1} \right\rfloor$$

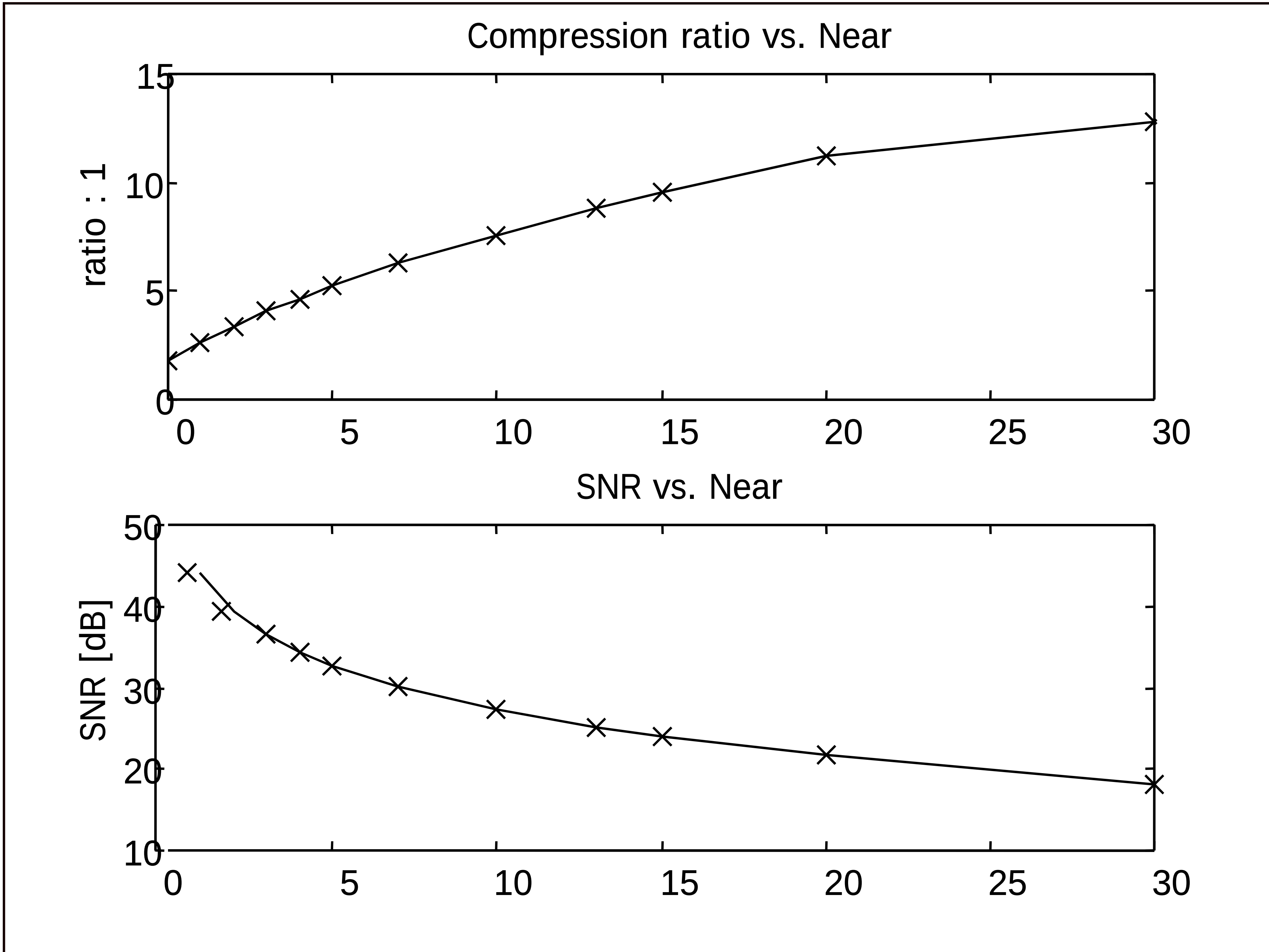
And the profit is . . .

- Narrower geometric distribution



<u>Shorter</u> Golomb-Rice Codes

- Spending more time in Run-Mode

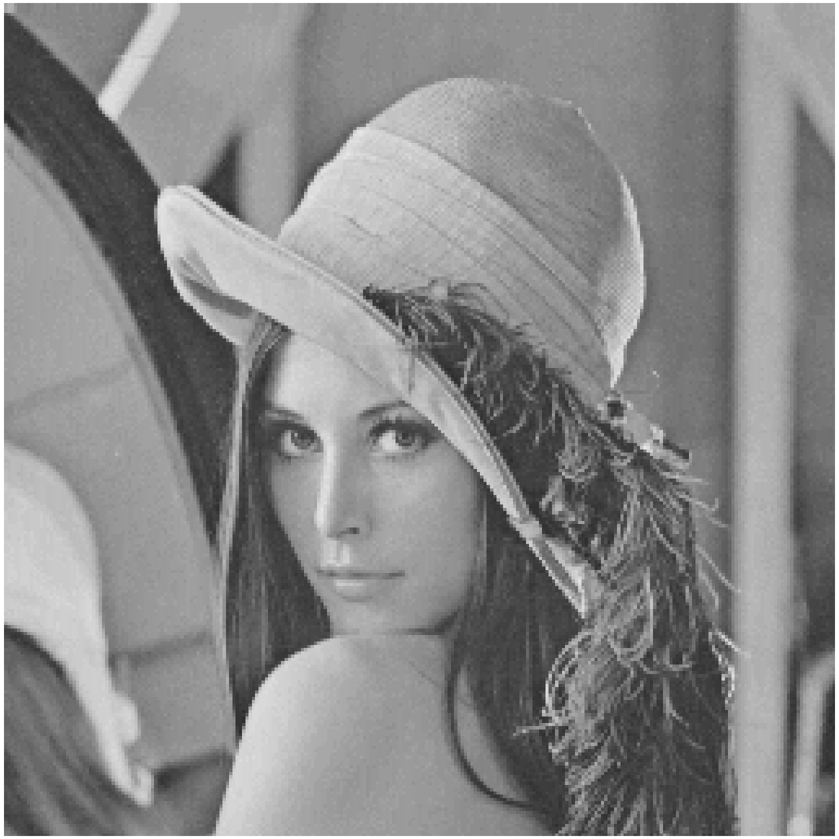

Compression ratio vs. Near
ratio : 1
15
10
5
0
0
5
10
15
20
25
30
SNR vs. Near
SNR [dB]
50
40
30
20
10
0
5
10
15
20
25
30

Explanation
(last graph: Lena for different Near values)

ניתן לראות, שבערכים נמוכים העלייה ביחס הדחיסה ליניארית עם Near, אולם בערכים גבוהים של Near כמעט ואין שיפור ביחס הדחיסה. הסיבה לכך, היא שהLOCO מאבד את ה Contexts בערכי Near גבוהים, ולכן מאבד את יעילותו. גם איכות התמונה הדחוסה בערכים הגבוהים ירודה ביותר, כפי שניתן לראות מגרף ה SNR.

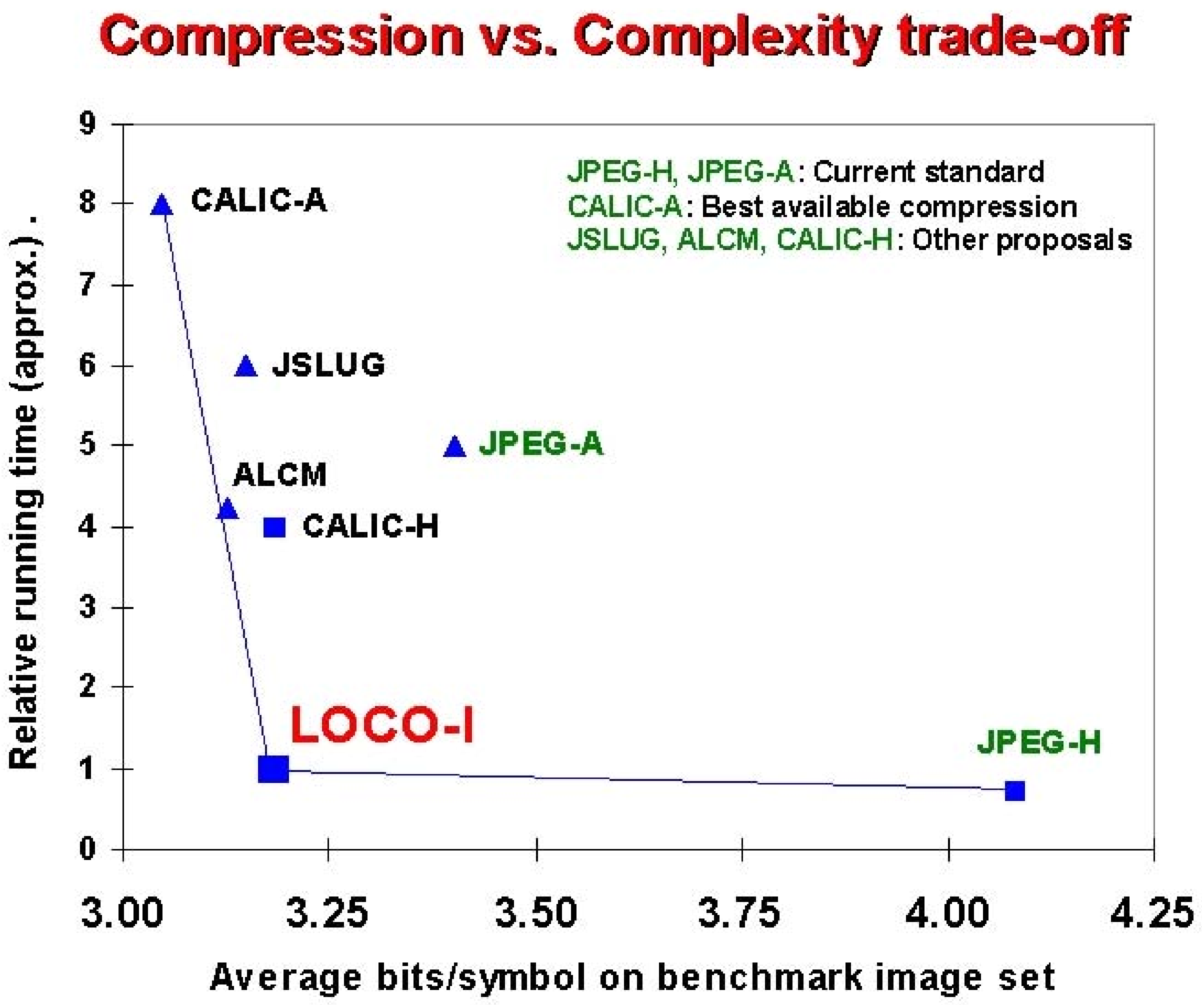
Different Near values compression

Near = 3



Near = 10

Benchmark Comparison



Download Free from UBC