LENA First-Exam Project

- Write down algorithms of

  local Huffman code,
  modified local Huffman code, } for differential

  and implement them.

- Write down algorithm of

  arithmetic code } for differential (Tree Structure)

  and implement it.

- No hand writing
- Due: in two weeks

Runlength Encoding of Bit Planes

- The simplest method of encoding binary data such as binary text and documents in the field of facsimile transmission
- A new message set constructed based on the runs of 0's and 1's.
- In 1-D runlength coding, the runs of 0's and 1's are variable-length encoded using separate Huffman tables tailored to the statistics of each.
- In 2-D runlength coding, the basic idea is to encode the starting position of a run

in the current line relative to the previous line

- In general, 2-D runlength coding achieves higher compression ratios since it uses the vertical correlation in an image.
- The 1-D runlength entropy:

$$H = \frac{H_0 + H_1}{\bar{r_0} + \bar{r_1}}$$

H_0: entropy in bits/runlength symbol 0
H_1: entropy in bits/runlength symbol 1
$\bar{r_0}$: the average runlength of 0's
$\bar{r_1}$: the average runlength of 1's

- The entropy values for both the binary representation and Gray code representation increase substantially as one moves from the MSB plane to the LSB plane
  - a lot of redundancy in the higher bit planes
  - The lower bit planes behave almost like random noise
  - Except the MSB bit plane, the Gray code entropies $<$ the binary code entropies, significantly $<$ for the higher bit planes

Arithmetic Encoding of Bit Planes

- adaptive binary arithmetic coder such as the Q-coder
- The m neighboring pixels of the current pixel form a context, c, for encoding that pixel.
- The image is assured to have been generated by an mth-order Markov source:
  - each state determined by a certain combination of the m neighboring pixels: a total of 2^m states

- The entropy of the mth-order Markov source
  - $H(S) = \Sigma_c \, p(c) H(S|c)$

$$\begin{aligned} H(S|c) = & -p(0|c) \log p(0|c) \\ & -(1 - p(0|c)) \log(1 - p(0|c)) \end{aligned}$$

  - $p(c), p(0|c) : c = 1, 2, \ldots, 2^m$
- A nonadaptive binary arithmetic coder that uses knowledge of $p(0|c)$ would encode the Markov source at a bit rate $\approx$ its entropy

- If an adaptive coder is employed, the resulting bit rate may be even lower than the entropy in certain cases:
  - updating the estimate of $p(0|c)$ each time it visits a particular context
  - its knowledge of $p(0|c)$ more accurate than a global estimate based upon a stationary assumption

LENA:

- For the four most significant bit planes, the total bit rate:
  - Gray code bit planes:
    - Q-coder: 1.02 bits/pixel
    - 1-D runlength: 1.60 bits/pixel
  - binary bit planes:
    - Q-coder: 1.60 bits/pixel
    - 1-D runlength: 2.18 bits/pixel

△ multiple context masks
△ mth order Markov source generalization

- For each of the two least significant bit planes, Q-coder costs 1 bit/pixel more than 1-D runlength:
  - The inherent inefficiency of the Q-coder in encoding a stationary source
  - Transmitting the last two or three bit planes unencoded
- Creating decorrelated bit planes to be encoded independently (Gray code)
  - Information contained in the previously encoded planes can be used to encode the current plane:

- Extending the context pixel assignment to include pixels from both the current and previous planes: 3D context
  - The binary and Gray code get closer in 3D context
  - The total bit rates by using 3D context are slightly lower than by using 2D context

Chapter 7
Lossless Predictive Coding

- For typical images, the values of adjacent pixels are highly correlated and a great deal of information about a pixel value can be obtained by inspecting its neighboring pixel values:
  - This property is exploited in predictive coding techniques.
- An image modeled as an mth-order Markov source, each pixel represented by k bits:
  - $x_m \sim \underline{x_0, x_1, \ldots, x_{m-1}}$
  - # (states) $= K^m, K = 2^k$

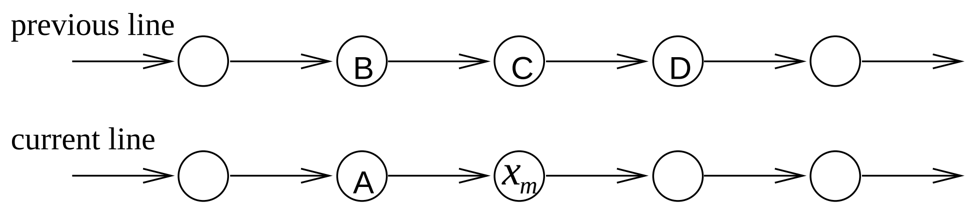
- For each state, K conditional probabilities $p(x_m|x_{m-1},...,x_0)$ for $x_m = 0, 1, ..., K-1$
- $H(S|x_{m-1},...,x_0) = -\Sigma_{x_m} p(x_m|x_{m-1},...,x_0) \log p(x_m|x_{m-1},...,x_0)$
- $H(S) = \Sigma_{x_{m-1},\cdots,x_0} p(x_{m-1},\cdots,x_0) H(S|x_{m-1},\cdot,x_0)$
  $= -\Sigma_{x_{m-1},\cdots,x_0} p(x_{m-1},\cdots,x_0) \log p(x_m|x_{m-1},\cdot,x_0)$

- Direct method:
  - K^m different codebooks (one for each state), each of which contains K entries:
    - needing conditional probabilities

- Lossless differential pulse code modulation (DPCM)
  - $\hat{x}_m = \text{argmax}_{x_m} p(x_m | x_{m-1}, \ldots, x_0)$
  - $e_m = x_m - \hat{x}_m$
  - a look-up table of K^m entries for $\hat{x}_m$
  - a single code book for entropy encoding e_m based on the histogram of the differential signal
    * a local codebook
    * a global codebook

- The prediction can be formed as a linear combination of the m previous pixel values:
    * no need for a lookup table
    * no need for conditional probabilities
  - Linear prediction is suboptimal compared to non-linear prediction
    * compensated for significant reduction in complexity and storage

DPCM predictor



- A higher order predictor generally outperforms lower order ones
- Studies on television images and radiography demonstrate: only a marginal gain beyond a third-order predictor
- Predictor coefficients:

  fixed for all images (global prediction)
  varied from image to image (local ...)
  varied within an image (adaptive ...)

- The gain achieved by local prediction is usually small comparing to global prediction
  $\hat{x}_m = 0.75A - 0.50B + 0.75C$

Huffman Encoding of Differential Images

- The differential image typically has a largely reduced variance compared to the original image and is also significantly less correlated.
  - LENA: $\sigma = 47.94 \ \rightarrow \sigma_e = 6.94$
  - BOOTS: $\sigma = 59.98 \ \rightarrow \sigma_e = 13.37$

- The differential image is usually entropy encoded to achieve lossless compression.
  - To encode an image with a local Huffman code:
    - First pass:
      - e_m
      - histogram
      - codebook
    - Second pass:
      - encode e_m
  - With a global Huffman code:
    - An average histogram of e_m
    - No need for two passes

- A modified Huffman code
  - Large differences (less probable) combined into one symbol
  - Codeword associated with that symbol used as a prefix to the original 8-bit pixel value rather than the 9-bit differential
- The global modified Huffman code:
  a viable technique in applications when maintaining a small size codebook is of primary importance.