ECE6258 Lecture 18

Coding Binary Images
Quantization

Announcements

- Problem Set #4 is posted on the web site.
- Due: Friday, October 10, 2003

Coding Binary Images

- Binary images, such as printed documents contain, on average many fewer than 1 bit/sample.
- They are usually coded losslessly (within sampling limits).
- Markov models can be successfully applied.
- Examples,
  - Group 3 and Group 4 fax standards
  - JBIG (Joint Binary Images Group) standard (ITU).

Run-length coding

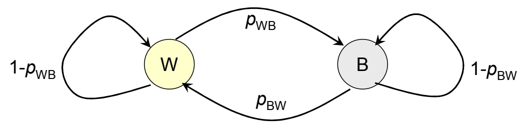
- For sources that emit "runs" of identical symbols (e.g. line art)
- Replace a sequence $\{x_n\}$ by a shorter sequence of symbol pairs $\{a_k, r_k\}$ such that

$$x_n = a_n \quad \text{for all } n\text{:} \quad \sum_{j=1}^{k-1} r_j < n \leq \sum_{j=1}^{k} r_j$$

- Entropy coding of new symbol pairs $\{a_k, r_k\}$
- For binary images, a_k, can usually be omitted.

Statistical model for binary images of line art

- Markov model [Capon, 1959]



- State probabilities

$$Pr(W) = 1 - Pr(B) = \frac{p_{BW}}{p_{BW} + p_{WB}}$$

Pr{k successive white pixels} = $(1-p_{WB})^{k-1}p_{WB}$ for k=1,2,3,…
Pr{k successive black pixels} = $(1-p_{BW})^{k-1}p_{BW}$ for k=1,2,3,…

Measured parameters for Capon model

Document	Weather Map	Printed Text
Pr{W}	0.887	0.935
Pr{B}	0.113	0.065
p_{WB}	0.027	0.024
p_{BW}	0.214	0.347
H(S)	0.241 bpp	0.215 bpp

[Kunt, 1974]

Facsimile compression standards

- Standards by the ITU-T
  - T4 (Group 3)
    - Used by all fax machines over PSTN
    - 1-D modified Huffman code (MH) or 2-D MMR code
  - T6 (Group 4)
    - Fax over digital networks
    - Always uses 2-D MMR
- Format
  - Horizontal resolution: 1728 pixels/line
  - Vertical resolution: 3.85 lines/mm (standard)
    7.70 lines/mm (fine)

Group 3 fax: modified Huffman code

- Lengths of white pixels and black pixels encoded within a scan line
- Each run represented as

  white runs: $r_w = 64 \times r_{w/make\text{-}up} + r_{w/term}$ similar for black

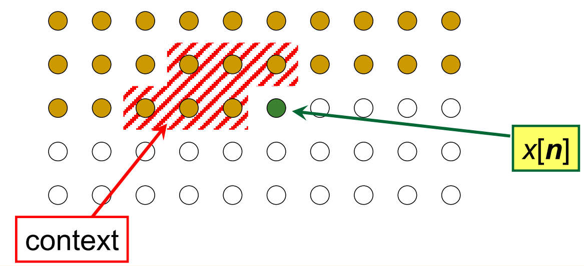
- Two separate Huffman code tables for white and black runs based on the statistics of 8 representative documents.
- Shortest code words (2 bits) for black runs of length 2 and 3
- Shortest code words (4 bits) for white runs of lengths 2…7.
- Special EOL codework for each line, 6x EOL as end of page.

Compression efficiency of T4 fax standard (1-d MHC)

CCITT Document Number	Standard resolution (2,052,864 pixels)	Fine resolution (4,105,728 pixels)
1	133,095	266,283
2	123,930	247,443
3	244,028	487,485
4	436,450	871,983
5	253,509	506,283
6	191,347	381,905
7	428,028	855,841
8	238,221	476,624
average	256,076	511,731

An Alternative: Context adaptive coding

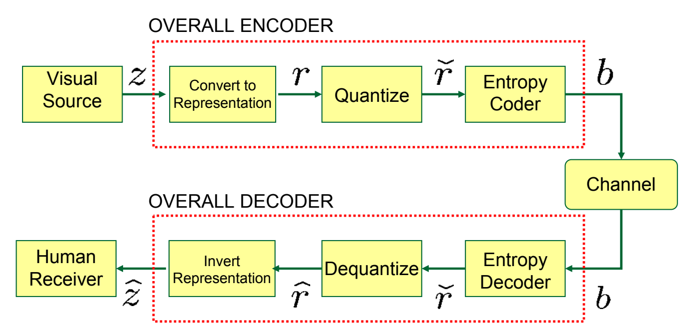
- Use values of previously encoded pixels to define state for Markov model.



Context Adaptive Coding

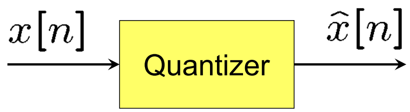
- Context vector defines the Markov state
  - Feasible for binary images
  - JBIG (Joint Binary Image Group) standard used 10 binary pixels as context.
    - 1024 states
    - Pixels encoded using adaptive implementation of an arithmetic coder

A General Coding Structure



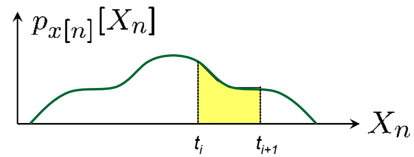
$$\text{error} = \mathcal{F}(z - \hat{z})$$

Quantization



- The output of the quantizer is a sequence of symbols that is fed to an entropy encoder.
- If the pdf (probability density function) of *x*[*n*] is known, the probabilities of the quantized values can be calculated.

Quantization (cont'd)



- If $t_i \leq x[n] \leq t_{i+1} \quad \hat{x}[n] = s_i$

$$P[s_i] = \int_{t_i}^{t_{i+1}} p_{x[n]}[X_n]\, dX_n$$

Common pdf models for Images

- Gaussian

$$p_x(X) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(X-m)^2/(2\sigma^2)}$$

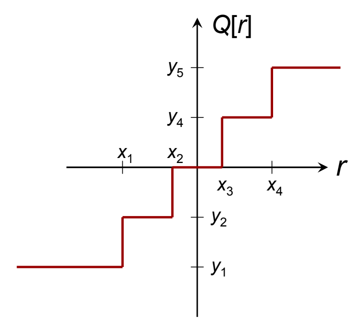
- Laplacian

$$p_x(X) = \frac{\lambda}{2} e^{-\lambda|X|} \qquad X \geq 0$$

- Generalized Gaussian

$$p_x(X) = ae^{-b|X-m|^c}$$

Quantization Intervals



- Interval boundaries

  $-\infty, x_1, x_2, x_3, x_4, \infty$

- Representation values

  y_1, y_2, y_3, y_4, y_5

Formalizing

- An N-point scalar quantizer Q is a mapping $Q: R \rightarrow C$, where R is the real line and $C=\{y_1,y_2,\ldots,y_N\} \subset R$ is the output set or **codebook** with size N.
- Associated with every N-point quantizer is a partition of the real line R into N cells or atoms. The i^{th} cell is given by

  $R_i = \{x \in R\text{: } Q(x)=y_i\} = Q^{-1}(y_i)$

Quantization Error

- There are a number of measures of the distortion introduced by a quantizer.
  - $d(x,y_i) = |x-y_i|^2$ (squared error)
  - $d(x,y_i) = |x-y_i|$ (absolute error)
  - $d(x,y_i) = |x-y_i|^m$ (m^{th} power)
- Mean-Squared Error

$$D = E[(X-Q(X))^2] = \sum_{i=1}^{N} \int_{R_i} (x-y_1)^2 p_X(x)\,dx$$

Quantizer Design

- Design Problem #1

  Given $p_X(x)$, N, and distortion criterion, find $\{y_i\}$, $\{x_i\}$ to minimize D.

- Design Problem #2

  Assume knowledge of interval can be encoded using l_i bits.

$$R = \sum_{i=1}^{N} \ell_i \int_{x_i}^{x_{i+1}} p_X(x)\,dx$$

  Subject to $R \le R^*$, find $\{l_i\}$, $\{x_i\}$, $\{y_i\}$, N that minimize D.

Uniform Quantizers

- Regular quantizers
- Partitions of same size
- Reconstruction levels are midpoints of intervals.
- Implemented by most A/D converters
- Uniform quantizers minimize the maximum error

Optimum Quantizers

- Two conditions must be true for a compressor consisting of an encoder followed by a decoder to be optimal
  - The **encoder** must be optimal for the given decoder.
  - The **decoder** must be optimal for the given encoder.
- Necessary and sufficient conditions can be found for these two constraints separately.

The optimal encoder for a given decoder

- i.e. given $\{y_1, y_2, \ldots, y_N\}$ find $\{x_0, \ldots, x_N\}$
- The best encoder maps input values into the output reproduction level having the minimum distortion with respect to the input.
  - ➔ encoder uses a nearest neighbor rule

$$R_i \subset \{x : d(x, y_i) \leq d(x, y_j); \forall j \neq i\}$$

- For both squared and absolute error distortions

$$x_{i-1} = \frac{y_{i-1} + y_i}{2}$$

The optimal decoder for a given encoder

- Given a nondegenerate partition $\{R_i\}$, the unique optimal codebook for the mean squared error is given by $y_i = E[X | X \in R_i]$

$$y_i = \frac{\int_{R_i} x p_X(x)\, dx}{\int_{R_i} p_X(x)\, dx}$$

Centroid condition

The Lloyd iteration for codebook improvement

1. Given a codebook $C_m = \{y_i\}$, find the optimal partition into quantization cells using the nearest neighbor condition.
2. Using the centroid condition, find C_{m+1}

The Lloyd algorithm

1. Begin with an initial codebook, C_1. Set m=1.
2. Given codebook C_m, perform a Lloyd iteration to generate an improved codebook C_{m+1}.
3. Compute the average distortion for C_{m+1}. If it has changed by a small enough amount since the last iteration, stop. Else $m+1 \rightarrow m$ and go to Step 2.