Feature Extraction

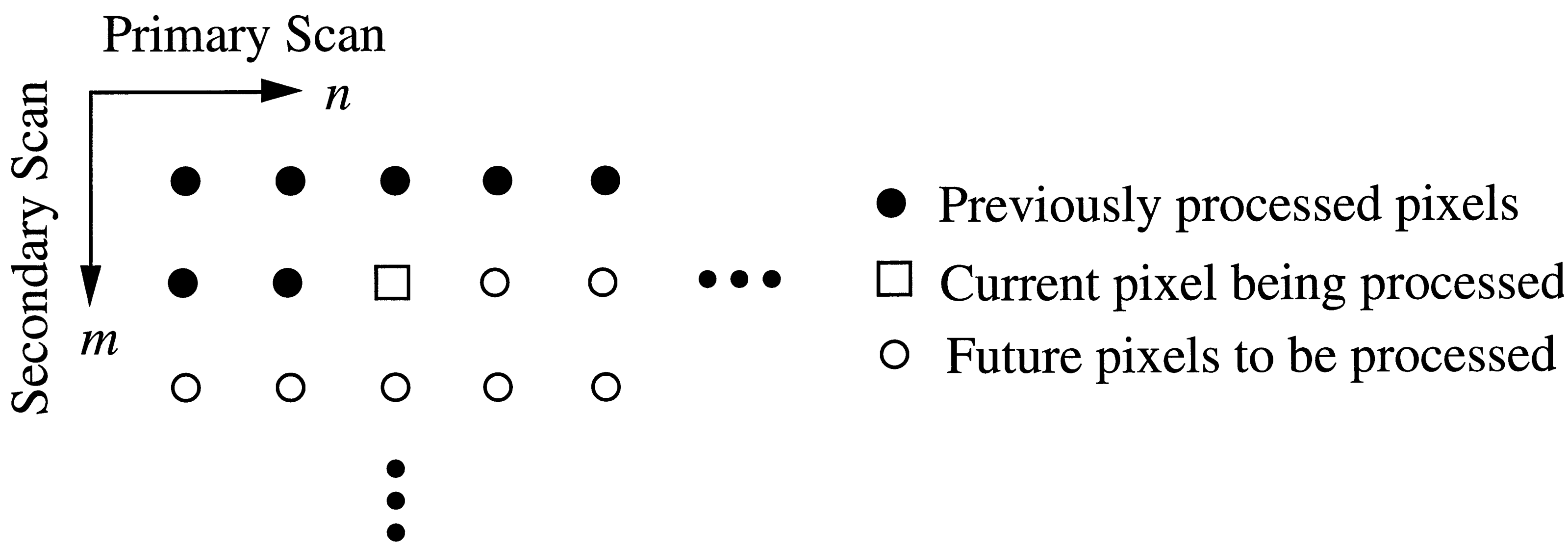
- Partition image into N x N blocks of pixels
- For each block, compute a feature vector to represent all pixels contained within that block
- If feature vector provides a complete description of the block, it can be used as part of a lossless algorithm; otherwise algorithm will be lossy

Example Features

- Pixel values
  - N = 1
  - lossless

Example Features (cont.)

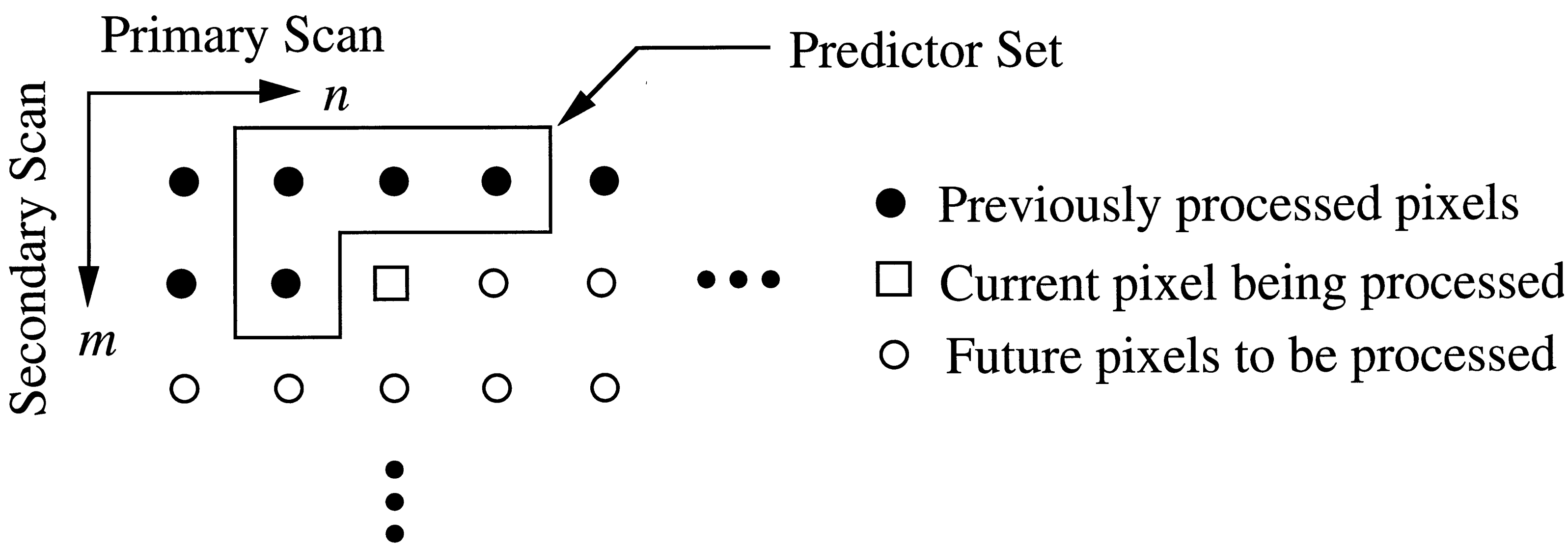
- **Difference between current pixel value and that of previous pixel on line (N = 1)**



- $e[m,n] = x[m,n-1] - x[m,n]$
- **lossless, since** $x[m,n] = x[m,n-1] - e[m,n]$
- **basis for Differential Pulse Code Modulation (DPCM)**

Example Features (cont.)

- **Error in prediction of current pixel based on values of previously processed neighboring pixels**



  - $e[m,n] = \hat{x}[m,n] - x[m,n]$
  - **lossless, since** $x[m,n] = \hat{x}[m,n] - e[m,n-1]$
  - **basis for predictive encoders**

Example Predictors

- **Linear**

$$\hat{x}[m,n] = \sum_{(k,l)\in\Omega} a_{kl}\, x[m-k, n-l]$$

  - **minimum mean-squared error predictor**
    - » **coefficients are solution to**

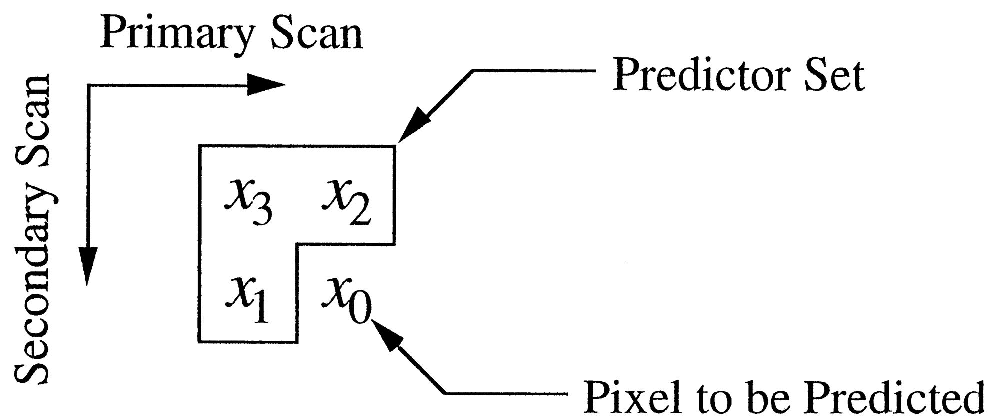
$$\sum_{(k,l)\in\Omega} a_{kl}\, r[k'-k, l'-l] = r[k',l'], \quad (k',l') \in \Omega$$

    - » **where**

$$r[k,l] = \sum_{(m,n)} x[m,n]\, x[m+k, n+l]$$

Example Predictors (cont.)

- **Nonlinear (Graham Predictor)**



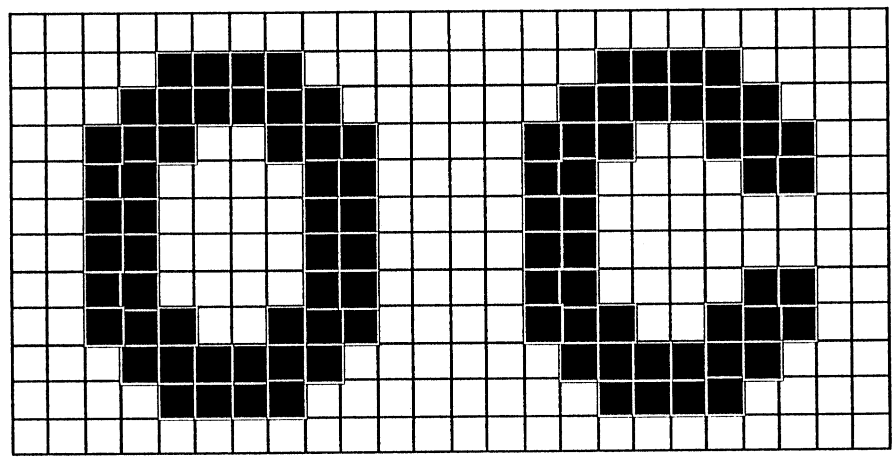
$$d_{13} = |x_1 - x_3|$$

$$d_{23} = |x_2 - x_3|$$

$$\hat{x}_0 = \begin{cases} x_1, & d_{13} > d_{23}, \\ x_2, & d_{13} < d_{23} \end{cases}$$

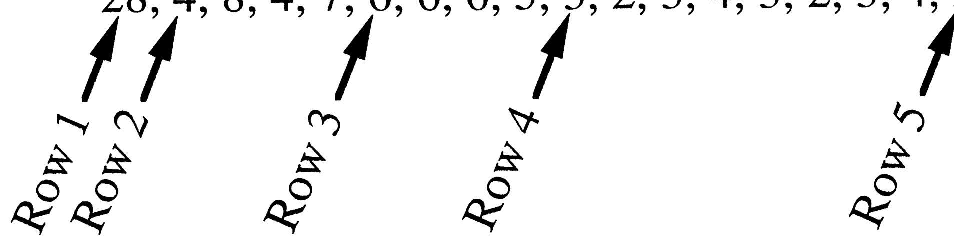
Example Features (cont.)

- **Lengths of runs of 0's and 1's in a black/white image**



Run-length code

28, 4, 8, 4, 7, 6, 6, 6, 5, 3, 2, 3, 4, 3, 2, 3, 4, 2, 4, 2, 4, 2, ...



– **variable block length, lossless**

Example Features (cont.)

- **Discrete cosine transform (DCT) (N = 8)**

$$X[k,l] = \frac{1}{4} C[k] C[l] \sum_{k=0}^{7} \sum_{l=0}^{7} x[m,n] \cos\left[\frac{(2m+1)k\pi}{16}\right] \cos\left[\frac{(2n+1)l\pi}{16}\right],$$

$$C[k] = \begin{cases} 1/\sqrt{2}, & k = 0, \\ 1, & \text{else} \end{cases}$$

  - **inverse transform exists with similar structure (⇒lossless)**
  - **closely related to Discrete Fourier Transform (DFT)**
  - **compared to DFT, DCT is real-valued, and yields better energy compaction**
  - **basis for Joint Photographic Experts Group (JPEG) standard**

Example Features (cont.)

- **Block truncation statistics (N > 1)**
  - **block structure**

$$\begin{array}{ccc} x_1 & \cdots & x_N \\ \vdots & \ddots & \vdots \\ x_{N^2-N+1} & \cdots & x_{N^2} \end{array}$$

  - **first and second moments**

$$\bar{x} = \sum_{i=1}^{N^2} x_i \qquad \overline{x^2} = \sum_{i=1}^{N^2} {x_i}^2$$

Example Features (cont.)

- **Block truncation statistics (cont.)**
  - **binary mask**

$$\tilde{x}_i = \begin{cases} a, & x_i > \bar{x}, \\ b, & \text{else} \end{cases}$$

  - a **and** b **are chosen to preserve first and second moments, i.e.**

$$\bar{\tilde{x}} = \bar{x} \quad \text{and} \quad \overline{\tilde{x}^2} = \overline{x^2}$$

Example Features (cont.)

- **Block truncation statistics (cont.)**
  - **example**

Original Image Block

1	1	2	2
1	2	7	7
2	7	8	8
7	8	9	9

Reconstructed Image Block

2.3	2.3	2.3	2.3
2.3	2.3	8.6	8.6
2.3	8.6	8.6	8.6
8.6	8.6	8.6	8.6

  - **feature is lossy**
  - **mean and coarse structure of block are preserved**

Bit Rate for Block Truncation Code

- **Bit rate is number of binary digits required per image pixel**
- **To encode one $N \times N$ block, must transmit:**
  - **values of parameters *a* and *b* - 8 bits each**
  - **structure of binary mask - N^2 bits**
- **Overall bit rate**

$$B = \frac{N^2 + 16 \text{ bits / block}}{N^2 \text{ pixels / block}} = 1 + \frac{16}{N^2} \text{ bits / pixel}$$