IMAGE COMPRESSION-II

Week IX

IMAGE COMPRESSION

- Data redundancy
- Self-information and Entropy
- Error-free and lossy compression
- Huffman coding
- Predictive coding
- Transform coding

Data Redundancy

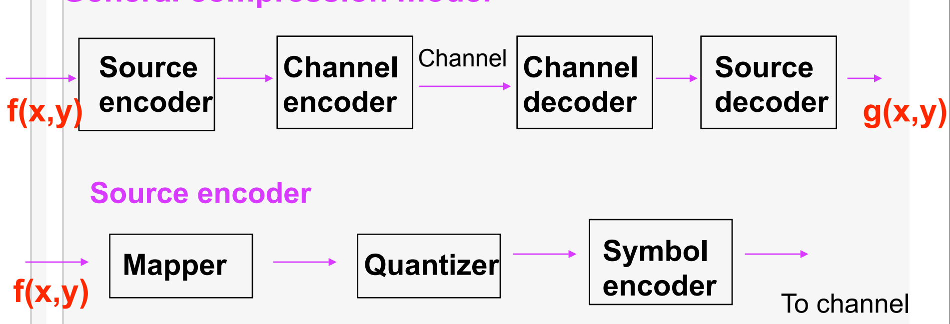
- **CODING**: Fewer bits to represent frequent symbols.
- **INTERPIXEL / INTERFRAME**: Neighboring pixels have similar values.
- **PSYCHOVISUAL**: Human visual system can not simultaneously distinguish all colors.

Coding Redundancy (contd.)

- Consider equation (A): It makes sense to assign fewer bits to those r_k for which $p_r(r_k)$ are large in order to reduce the sum.
- this achieves data compression and results in a variable length code.
- More probable gray levels will have fewer # of bits.

$$L_{avg} = \sum_{k=0}^{L-1} l(r_k)\, p_r(r_k) \quad \rightarrow (A)$$

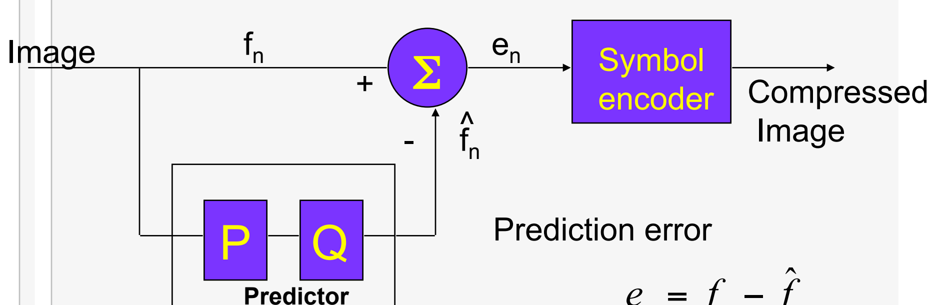
General Model



8.2.9: Predictive coding

To reduce / eliminate interpixel redundancies

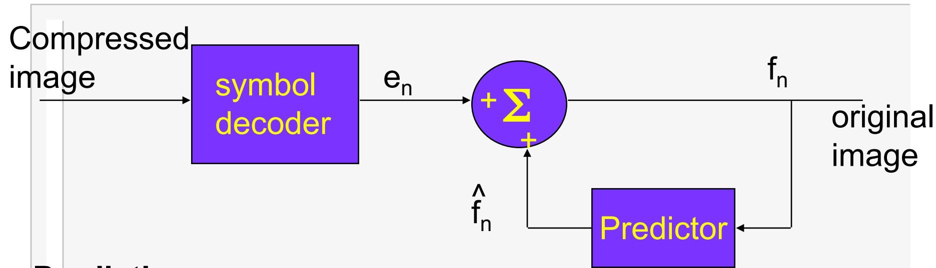
Lossless predictive coding: ENCODER



Prediction error

$$e_n = f_n - \hat{f}_n$$

Decoder



Prediction error:

$e_n = f_n - \hat{f}_n$

e_n is coded using a variable length code (symbol encoder)

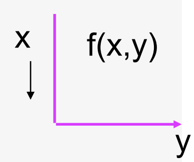
$f_n = e_n + \hat{f}_n$

Example

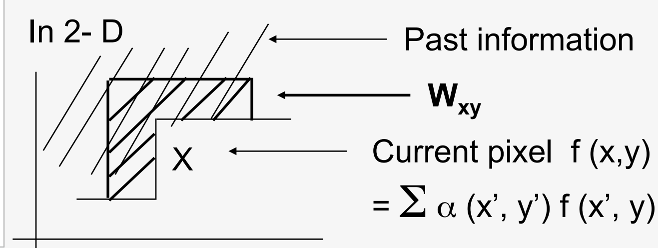
Example 1: $\hat{f}_n = \text{Int}\left(\sum_{i=1}^{m} \alpha_i f_{n-i}\right)$

→ Linear predictor ; m = order of predictor

Example 2: $\hat{f}(x,y) = \text{Int}\left(\sum_{i=1}^{m} \alpha_i f(x, y-i)\right)$



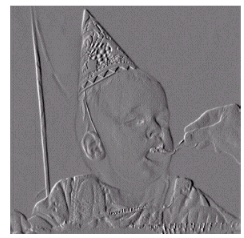
In 2- D



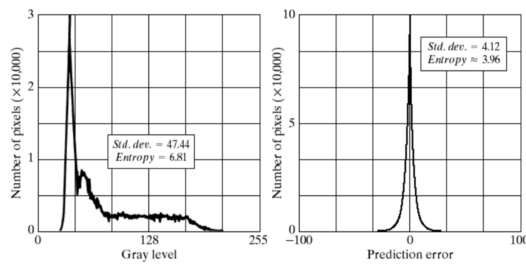
$= \sum \alpha(x', y') f(x', y')$

$(x', y') \in W_{xy}$

An example of first order predictor



Prediction error image

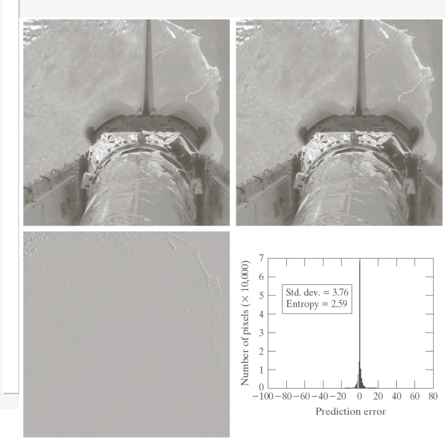


Histograms of original and error images.

Predictive coding and temporal redundancy

$$\hat{f}(x,y,t) = round[\alpha f(x,y,t-1)]$$

$$e(x,y,t) = f(x,y,t) - \hat{f}(x,y,t)$$

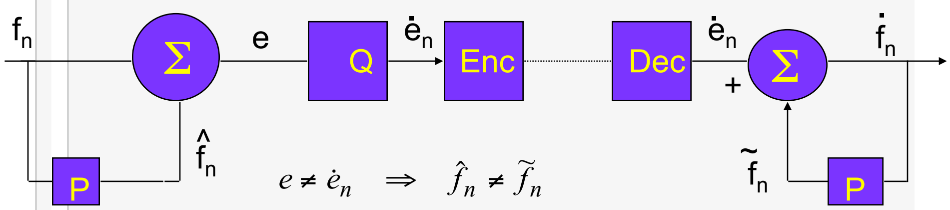


a b
c d

FIGURE 8.35 (a) and (b) Two views of Earth from an orbiting space shuttle video. (c) The prediction error image resulting from Eq. (8.2-36). (d) A histogram of the prediction error. (Original images courtesy of NASA.)

Lossy Compression (pp. 596)

Lossy compression: uses a quantizer to compress further the number of bits required to encode the 'error'.
First consider this:



Notice that, unlike in the case of loss-less prediction, in lossy prediction the predictors P "see" different inputs at the encoder and decoder

Quantization error

This results in a gradual buildup of error which is due to the quantization error at the encoder site.
In order to minimize this buildup of error due to quantization we should ensure that 'Ps' have the same input in both the cases.

f_n 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

$\hat{f}_n = f_{n-1};$ Quantizer: e=0 if less than1; e=2 if >= 1.

e_n = 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1

$\dot{e}_n$ = 0 2 2 2 2 . . . . . .

$\dot{f}_n$ = 0 2 4 6 8 10 . . . .

Example

Example:

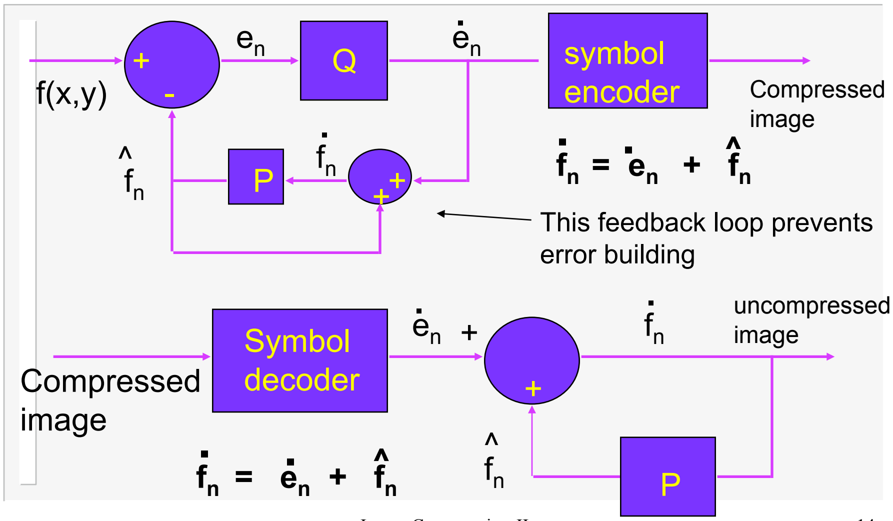
$$\hat{f}_n = \alpha \, \dot{f}_{n-1}$$

and $\dot{e}_n = \begin{cases} +\xi & e_n > 0 \\ -\xi & e_n < 0 \end{cases}$

$0 < \alpha < 1$
prediction coefficient

$$\dot{f}_n = \dot{e}_n + \hat{f}_n$$
$$= \dot{e}_n + \alpha \, \dot{f}_{n-1}$$

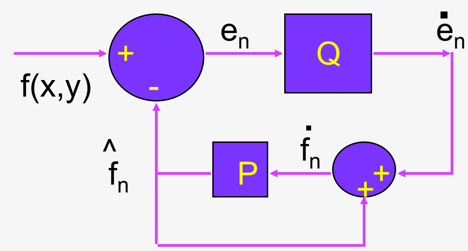
Predictive Coding With Feedback



Example

with feedback

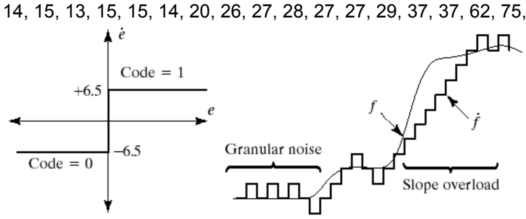
f_n =	0	1	2	3	4
e_n =	_	1	2	1	2
$\dot{e}_n$ =	_	0	2	0	2
$\dot{f}_n$ =	0	0	2	2	4
$\hat{f}_n$ =	_	0	0	2	2



Note: The quantizer used here is-- floor $(e_n/2)*2$. This is different from the one used in the earlier example. Note that this would result in a worse response if used without Feedback (output will be flat at "0").

Another example

{14, 15, 14, 15, 13, 15, 15, 14, 20, 26, 27, 28, 27, 27, 29, 37, 37, 62, 75, 77, 78, 79, 80, 81, 81, 82, 82}



a b
c

FIGURE 8.22 An example of delta modulation.

Input		Encoder				Decoder		Error
n	f	$\hat{f}$	e	$\dot{e}$	$\dot{f}$	$\hat{f}$	$\dot{f}$	$[f - \dot{f}]$
0	14	—	—	—	14.0	—	14.0	0.0
1	15	14.0	1.0	6.5	20.5	14.0	20.5	–5.5
2	14	20.5	–6.5	–6.5	14.0	20.5	14.0	0.0
3	15	14.0	1.0	6.5	20.5	14.0	20.5	–5.5
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
14	29	20.5	8.5	6.5	27.0	20.5	27.0	2.0
15	37	27.0	10.0	6.5	33.5	27.0	33.5	3.5
16	47	33.5	13.5	6.5	40.0	33.5	40.0	7.0
17	62	40.0	22.0	6.5	46.5	40.0	46.5	15.5
18	75	46.5	28.5	6.5	53.0	46.5	53.0	22.0
19	77	53.0	24.0	6.5	59.6	53.0	59.6	17.5
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.

A comparison (Fig 8.23)

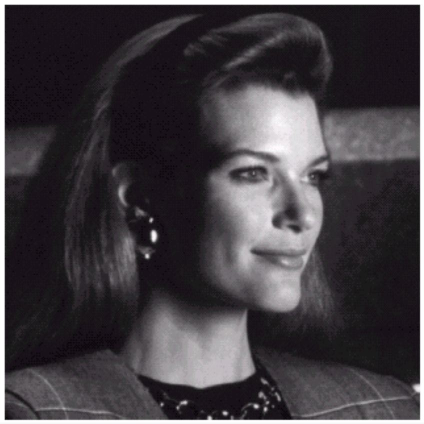
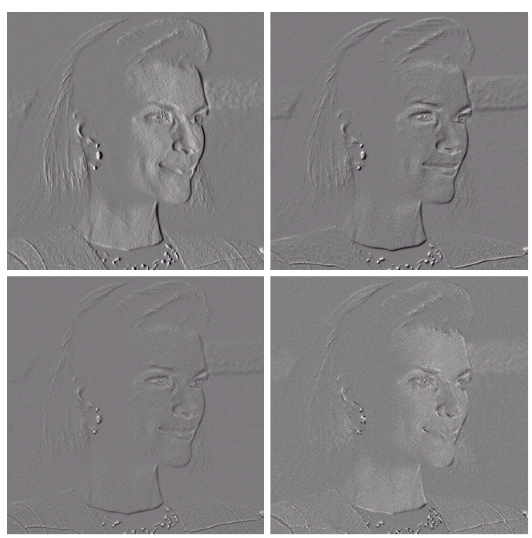


FIGURE 8.23 A 512 × 512 8-bit monochrome image.

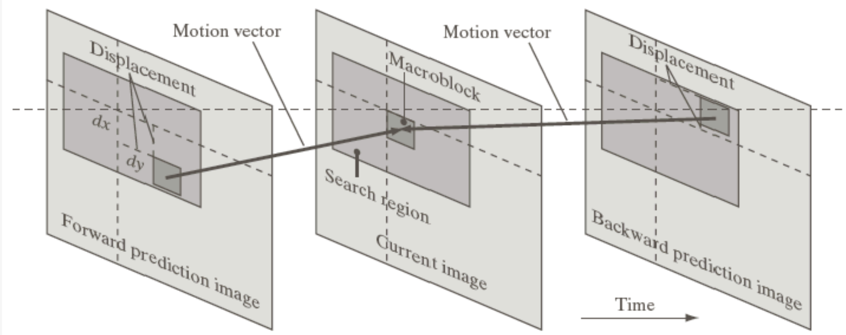
Four linear predictors

a b
c d

FIGURE 8.24 A comparison of four linear prediction techniques.



Motion compensation in Video



Coding sequence: e.g., IPPBPPI……