Outline

- Introduction
  - What is Image ?
  - Digital Image Processing .
  - Advantage & Disadvantages .
  - Fundamental Steps in DIP .
- Fundamentals
  - Data Compression & Data Redundancy.
  - Types of Data Redundancy .
  - Fidelity Criteria .
- Image Compression Models .
- Error Free Compression .
  - Variable Length Coding
  - LZW

Introduction

- **What is an Image ?**

  The world around us is 3D while the image obtain through a camera is 2D,Hence an image can be defined as "A 2D representation of 3D world. "

- **Types of Images**

  1. **Analog :** An analog image can be represented as continuous range of value representing position and intensity.

  2. **Digital :** It is composed of picture element called pixel.

Digital Image Processing

- **Digital Image**

A Digital Image is a representation of a two-dimensional image as a finite set of digital values, called **picture elements** or **pixels**.

- **Digital Image Processing**

Digital image processing focuses on two major tasks.

- Improvement of **pictorial information** for human interpretation.
- Processing of image data for **storage**, **transmission** and representation for autonomous machine perception.

Advantages of Digital Image

- Processing of digital image is **faster** and **cost effective**.
- It can be **effectively stored** and **transmitted** from one place to another.
- When suiting a digital image on can immediately see the **image is good or not.**
- Digital technique offers a plenty of store for versatile **image manipulation** .

Disadvantages of Digital Image

- Misused of **copyright** has become easier because images can be copy from the internet by just clicking on the mouse.

- Digital images **can't be enlarge** on certain size without compromising on quality .

- The **memory required** for store and processing a good quality of digital image is **very high**.

Types of Digital Image

1. Bit map Image /Monochrome/Binary

Each pixel consist of **single bit** either **0** or **1**,where 0 is black and 1 is white.

2. Gray-Scale Image

Each pixel consist of **eight bit** ,each bit range from **0** to **255**.

3.Colored Image

It consist of **3-primitive** colour **Red**, **Green** and **Blue** ,colored image consist of **24-bit** where 8-bit for red, 8-bit for green and 8-bit for blue.

Fundamental steps in DIP

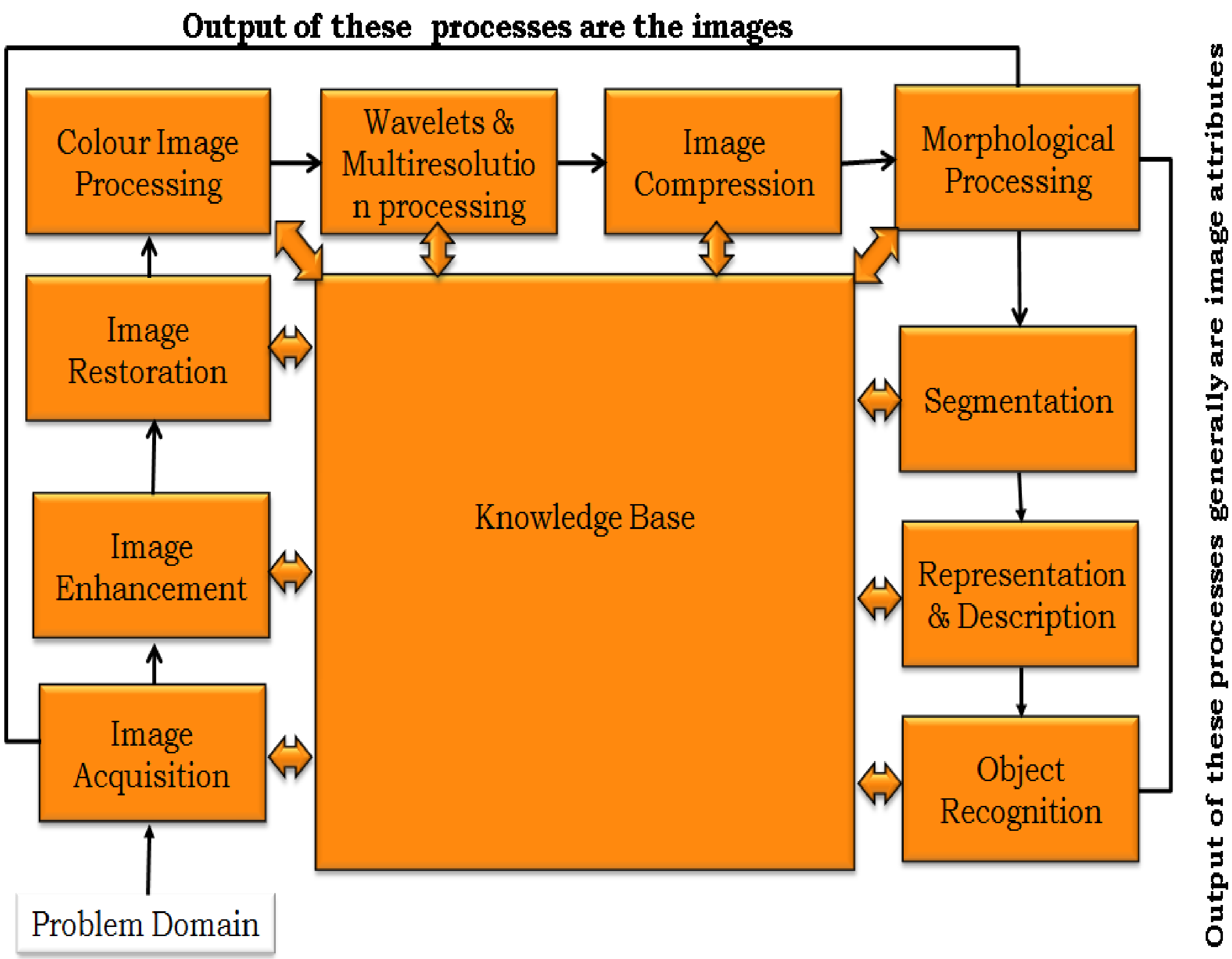


Image Compression

- Techniques for reducing the storage required to save an image or the bandwidth required to transmit it.

- Image compression address the problem of **reducing the amount of data** required to represent a digital image with **no significant loss** of information.

- **Image** = **Information** + **Redundant Data**

- Image compression techniques fall into Two categories:

  Lossless : Information preserving ,Low compression ratios .
  Lossy : Not information preserving ,High compression ratios .

Fundamentals

- **Data compression:** Data compression refers to the process of reducing the amount of data required to represent a given quantity of information.

- Data and Information are not the same thing ,**Data** is the means by which **information** is conveyed.

- Data compression aims to **reduce** the amount of data required to represent a given quantity of information while **preserving** as much information as possible.

Fundamentals

- The same amount of **information** can be represented by various amount of **data.**

Example:

✓ **Ex.1:** Your Brother, Ashish, will meet you at santacruz Airport in mumbai at 5 minutes past 6:00 pm tomorrow night.

✓ **Ex.2:** Your Brother will meet you at santacruz Airport at 5 minutes past 6:00 pm tomorrow night.

✓ **Ex.3:** Ashish will meet you at santacruz at 6:00 pm tomorrow night .

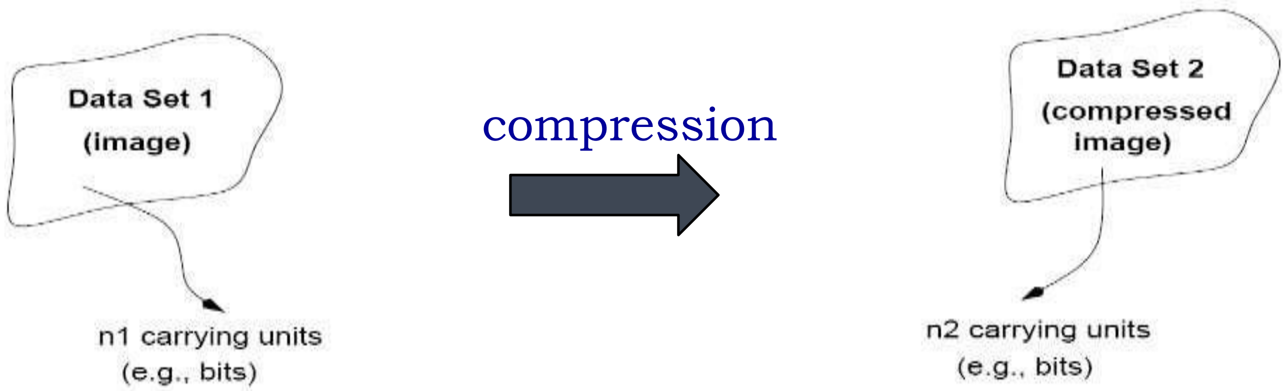
Fundamentals

- **Data Redundancy**
  It contains data (or words) that either provide no relevant information or simply restate that which is already known.

- The Relative **data redundancy R_D** of the first data set ,is defined by :

$$R_D = 1 - \frac{1}{C_R}$$

Where, **R_D** is **Relative Data Redundancy** ,**C_R** is **Compression ratio**.



Fundamentals

- C_R refers to the compression ratio and is defined by: $$C_R = \frac{n_1}{n_2}$$
- where ,**n1** and **n2** denoting the **information-carrying units** in two data sets that represent the same information/image.
- If **n1 = n2** , then **C_R =1** and **R_D=0**, indicating that the first representation of the information contains no redundant data.
- When **n2<<n1** , **$C_R \rightarrow$ large value(∞)** and **$R_D \rightarrow$ 1**. Larger values of C indicate better compression .

Fundamentals

- Three Principal type of Data Redundancies used in Image Compression :

  (1) Coding Redundancy
  (2) Interpixel Redundancy
  (3) Psychovisual Redundancy

Coding Redundancy

- **Code:** A list of symbols (letters, numbers, bits etc) .
- **Code word:** A sequence of symbols used to represent a piece of information or an event (e.g., gray levels).
- **Code word length:** Number of symbols in each code word

Example: (binary code, symbols: 0,1, length: 3)

0: 000	4: 100
1: 001	5: 101
2: 010	6: 110
3: 011	7: 111

Coding Redundancy

- The gray level histogram of an image can be used in construction of codes to reduce the data used to represent it. Given the normalized histogram of a gray level image where,

$$p_r(r_k) = n_k / n \qquad k = 0, 1, 2, \ldots\ldots\ldots\ldots L-1$$

- r_k is the **discrete random variable** defined in the interval [0,1]
- $p_r(k)$ is the **probability of occurrence** of r_k , **L** is the number of **gray levels**
- n_k is the **number of times** that k^{th} **gray level appears** , **n** is the **total number of pixels** in image.

Coding Redundancy

- Average number of bits required to represent each pixel is given by:

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

- Where, **$l(r_k)$** is the **number of bits** used to represent each value of **r_k**.

Interpixel Redundancy

- It is also called **Spatial & Temporal redundancy.**
- Because the pixels of most 2D intensity arrays are correlated spatially(i.e. **Each pixel is similar to or dependent on neighbor pixel**), information is replicated unnecessarily .
- This type of redundancy is related with the Interpixel correlations within an image.
- In video sequence, temporally correlated pixels also duplicate information. The codes used to represent the gray levels of each image have nothing to do with correlation between pixel.

Psychovisual Redundancy

- Certain information has relatively less importance for the quality of **image perception**. This information is said to be **psychovisually redundant.**

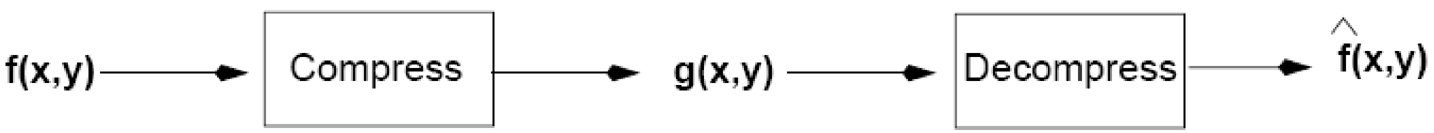
- Unlike **coding** and **interpixel redundancies**, the Psychovisual redundancy is related with the **real/quantifiable** visual information. Its elimination results a loss of quantitative information. However psychovisually the loss is negligible.

- Removing this type of redundancy is **a lossy** process and the lost information cannot be **recovered.**

- The method used to remove this type of redundancy is called **quantization** which means the mapping of a broad range of input values to a limited number of output values.

Fidelity criteria

- When lossy compression techniques are employed, the decompressed image will not be identical to the original image. In such cases , we can define **fidelity criteria** that measure the difference between this two images.



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

How close is $f(x,y)$ **to** $\hat{f}(x,y)$ **?**

- Two general classes of criteria are used :

  (1) Objective fidelity criteria

  (2) Subjective fidelity criteria

Objective Fidelity Criteria

- When information loss can be expressed as a **mathematical function** of input & output of a compression process.

- E.g.. **RMS error between 2 images.**
- Error between two images

$$e(x, y) = f'(x, y) - f(x, y)$$

So, total error between two images

$$\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [f'(x, y) - f(x, y)]$$

Where image are size of M x N

Subjective Fidelity Criteria

- A Decompressed image is presented to a cross section of viewers and averaging their evaluations.

- It can be done by using an **absolute rating scale .**

Or

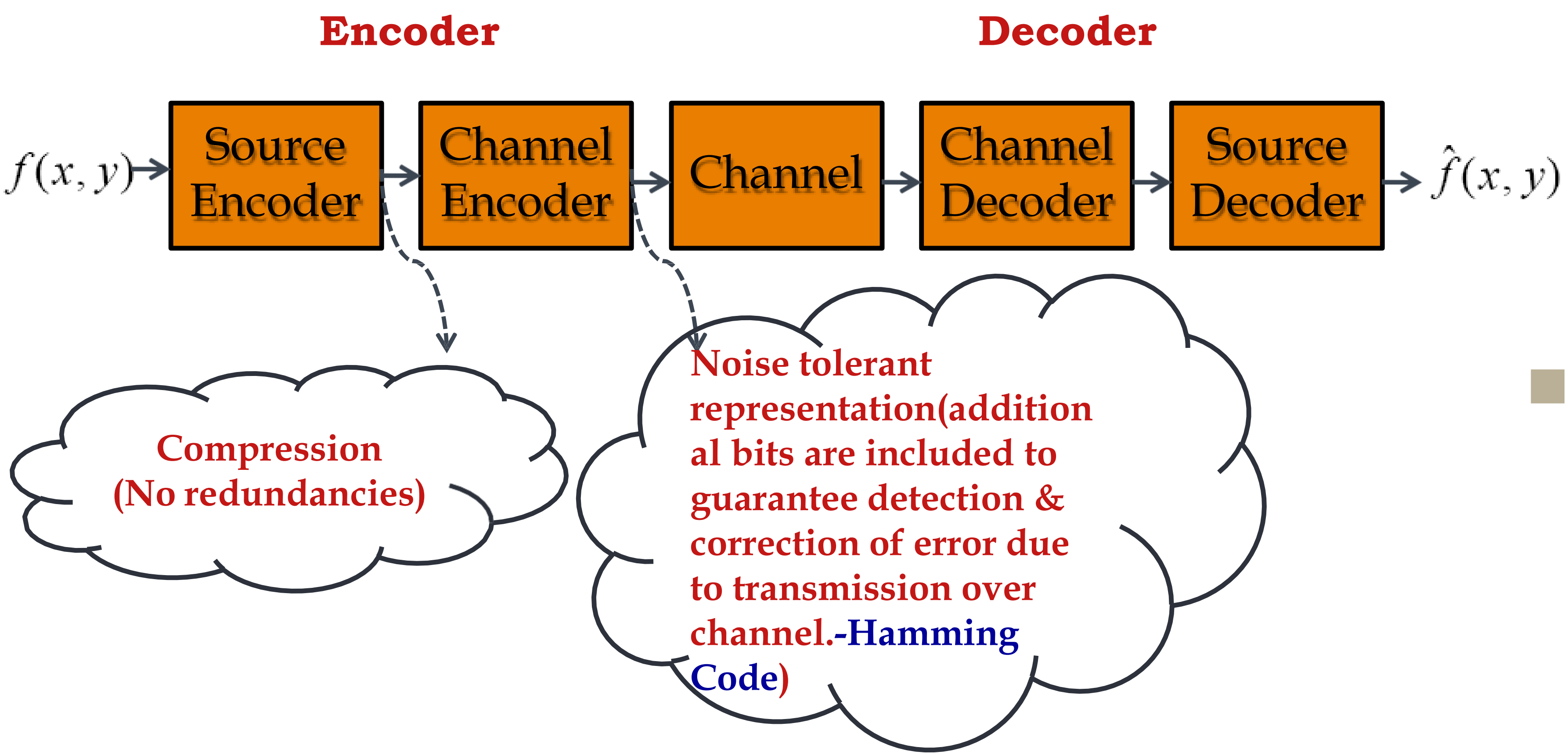
- By means of side by side comparisons of **f(x, y)** & **f′(x, y).**

- Side by Side comparison can be done with a scale such as **{-3, -2, -1, 0, 1, 2, 3}** to represent the subjective valuations **{much worse, worse, slightly worse, the same, slightly better, better, much better}** respectively.

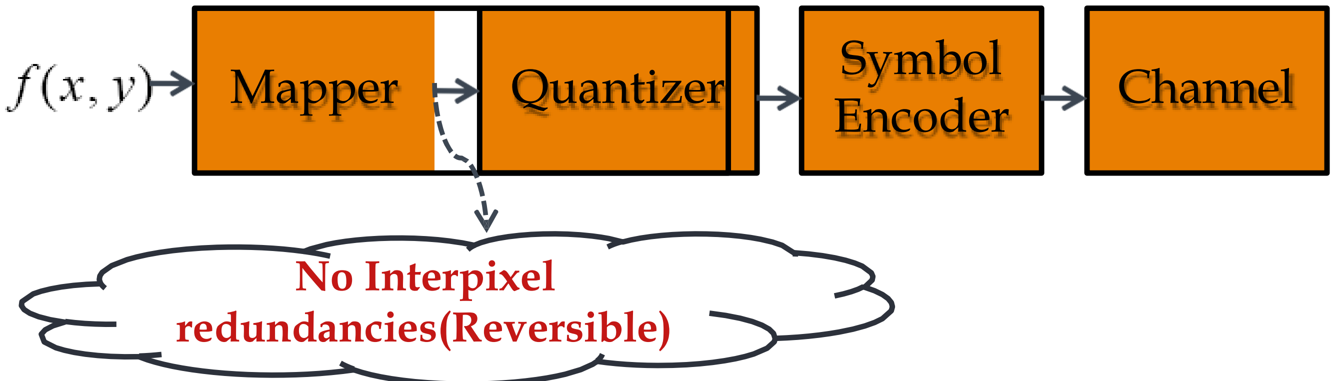
Image Compression Model

- The image compression system is composed of 2 distinct functional component: an **encoder** & a **decoder**.
- Encoder performs **Compression** while Decoder performs **Decompression**. Encoder is used to remove the redundancies through a **series of 3 independent operations**.
- Both operations can be performed in Software, as in case of Web browsers & many commercial image editing programs. Or in a combination of hardware & firmware, as in DVD Players.
- A **codec** is a device which performs coding & decoding.

Image Compression Model

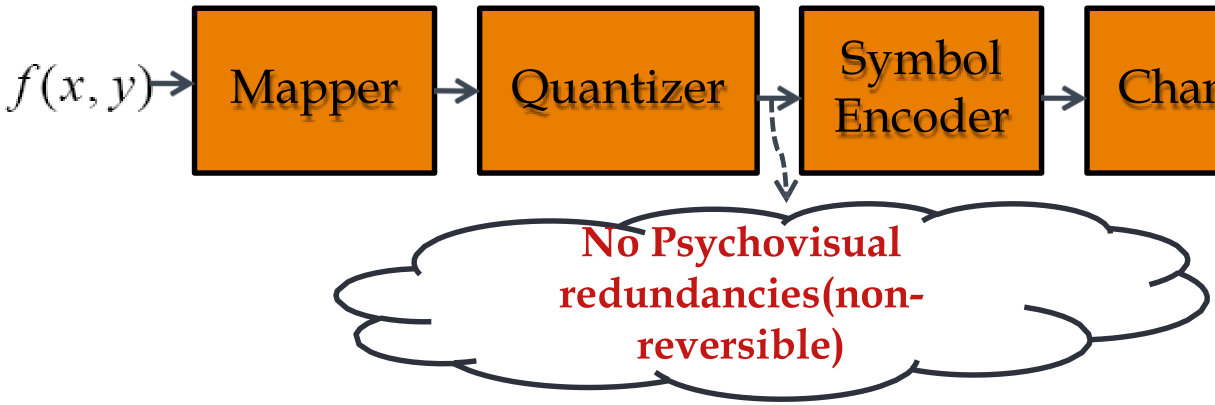


Mapper



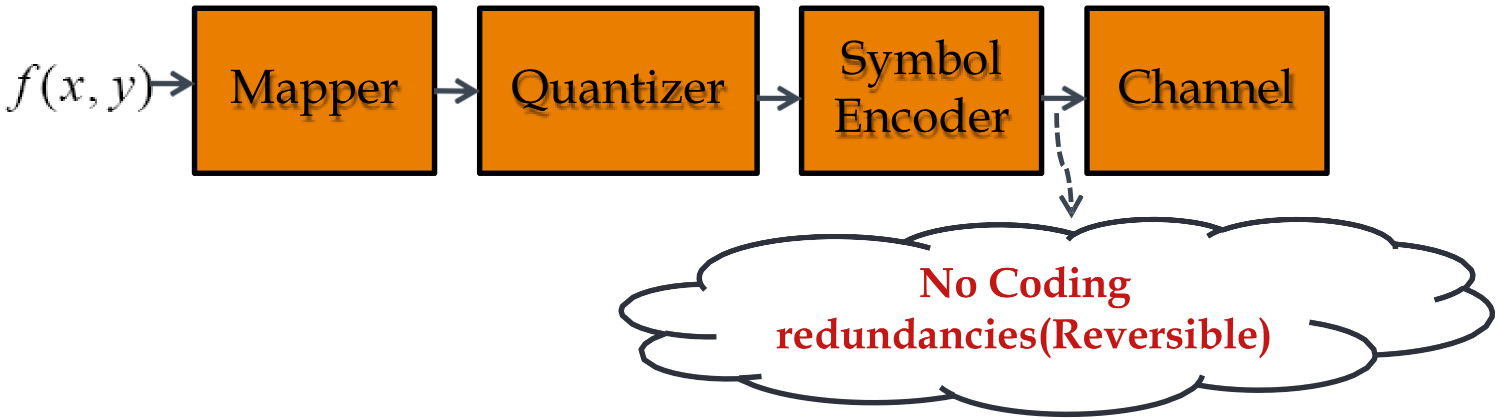
- It transforms input data in a way that facilitates **reduction of interpixel redundancies.**
- It is **reversible .**
- It may / may not reduce the amount of data to represent image.
- Ex. Run Length coding
- In video applications, mapper uses previous frames to remove temporal redundancies.

Quantizer



- It keeps irrelevant information out of compressed representations.
- This operation is **irreversible.**
- It must be **omitted when error free compression** is desired.
- The visual quality of the output can vary from frame to frame as a function of image content.

Symbol Encoder

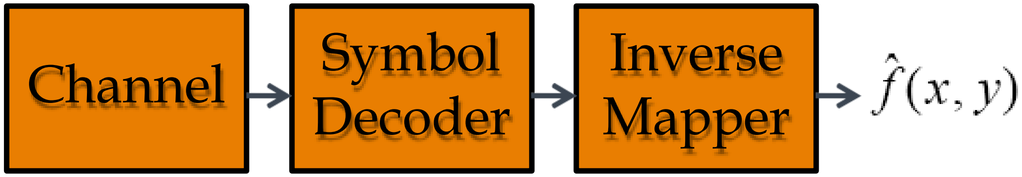


- Generates a fixed or variable length code to represent the Quantizer output and maps the output in accordance with the code.
- Shortest code words are assigned to the most frequently occurring Quantizer output values. Thus **minimizing coding redundancy**.
- It is reversible. Upon its completion, the input image has been processed for the **removal of all 3 redundancies.**

Decoding or Decompression Process

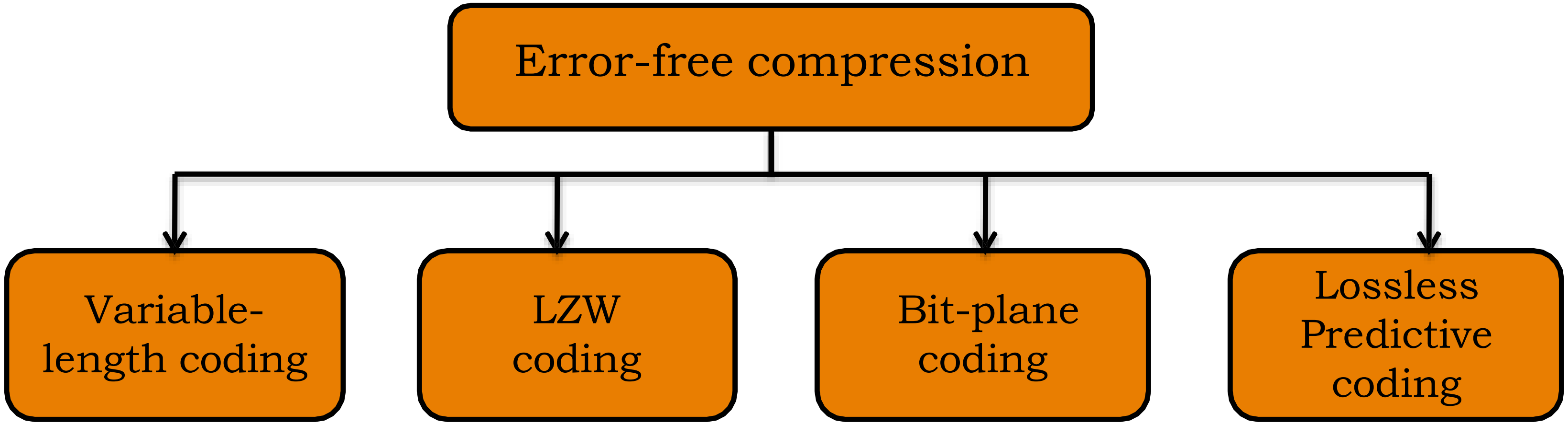
- **Inverse steps** are performed .
- Quantization results in irreversible loss, an **inverse Quantizer block is not included** in the decoder block.

Decoder



Error-free compression

- **Error-free compression** is generally composed of two relatively independent operations: (1) reduce the interpixel redundancies and (2) introduce a coding method to reduce the coding redundancies.



Variable-length Coding

- The coding redundancy can be minimized by using a variable-length coding method where the shortest codes are assigned to most probable gray levels.

- The most popular variable-length coding method is the **Huffman Coding**. **Huffman Coding:** The Huffman coding involves the following 2 steps.
- 1)Create a series of source reductions by ordering the probabilities of the symbols and combining the lowest probability symbols into a single symbol and replace in the next source reduction.

- 2)each Code reduced source starting with the smallest source and working back to the original source.

Variable-length Coding

1)Huffman source reductions:

ai's corresponds to the available gray levels in a given image.

Original source		Source reduction			
Symbol	Probability	1	2	3	4
a_2	0.4	0.4	0.4	0.4	0.6
a_6	0.3	0.3	0.3	0.3	0.4
a_1	0.1	0.1	0.2	0.3	
a_4	0.1	0.1	0.1		
a_3	0.06	0.1			
a_5	0.04				

Variable-length Coding

2) Huffman code assignments:

The first code assignment is done for **a2** with the **highest probability** and the last assignments are done for **a3** and **a5** with the **lowest probabilities.**

Original source			Source reduction							
Sym.	Prob.	Code	1		2		3		4	
a_2	0.4	1	0.4	1	0.4	1	0.4	1	0.6	0
a_6	0.3	00	0.3	00	0.3	00	0.3	00	0.4	1
a_1	0.1	011	0.1	011	0.2	010	0.3	01		
a_4	0.1	0100	0.1	0100	0.1	011				
a_3	0.06	01010	0.1	0101						
a_5	0.04	01011								

Variable-length Coding

- The shortest codeword (1) is given for the symbol/pixel with the highest probability (a2). The longest codeword (01011) is given for the symbol/pixel with the lowest probability (a5).

- The **average length of the code** is given by:

$$L_{avg} = (0.4)(1) + (0.3)(2) + (0.1)(3) + (0.1)(4) + (0.06)(5) + (0.04)(5)$$
$$= 2.2 \; bits/symbol$$

Variable-length Coding

- It is **uniquely decodable. Because** any string of code symbols can be decoded by examining individual symbols of string from left to right.
- Ex. 01010 011 1 1 00
- First valid code: 01010 – a3, 011 – a1,
- Thus, completely decoding the message, we get, a3a1a2a2a6
- Slower than Huffman coding but typically **achieves better compression.**

Lempel–Ziv–Welch (LZW)

- **Lempel–Ziv–Welch (LZW)** is a universal lossless data compression algorithm created by **Abraham Lempel, Jacob Ziv, and Terry Welch.**
- The key to LZW is building a dictionary of sequences of symbols (strings) as the data is read and compressed.
- Whenever a string is repeated, it is replaced with a single code word in the output.
- At **decompression time**, the same dictionary is created and used to replace code words with the corresponding strings.

Lempel-Ziv-Welch (LZW)

- A **codebook** (or **dictionary)** needs to be constructed. LZW compression has been integrated into a several images file formats, such as **GIF** and TIFF and **PDF**.

- Initially, the **first 256** entries of the dictionary are assigned to the **gray levels 0,1,2,..,255 (i.e., assuming 8 bits/pixel)**

Consider a 4x4, 8 bit image

39	39	126	126
39	39	126	126
39	39	126	126
39	39	126	126

Initial Dictionary

Dictionary Location	Entry
0	0
1	1
.	.
255	255
256	-
511	-

Lempel-Ziv-Welch (LZW)

- As the encoder examines image pixels, gray level sequences (i.e., blocks) that are not in the dictionary are assigned to a new entry.

39	39	126	126
39	39	126	126
39	39	126	126
39	39	126	126

Dictionary Location	Entry
0	0
1	1
.	.
255	255
256	39-39
511	-

- Is 39 in the dictionary……..Yes
- What about 39-39………….No
  * Add 39-39 at location 256

Lempel–Ziv–Welch (LZW)

39 39 126 126
39 39 126 126
39 39 126 126
39 39 126 126

CR = empty

If CS is found:
(1) No Output
(2) CR=CS

else:
(1) Output D(CR)
(2) Add CS to D
(3) CR=P

Concatenated Sequence: CS = CR + P

Currently (CR) Recognized Sequence	(P) Pixel Being Processed	Encoded Output	Dictionary Location (Code Word)	Dictionary Entry
	39			
39	39	39	256	39-39
39	126	39	257	39-126
126	126	126	258	126-126
126	39	126	259	126-39
39	39			
39-39	126	256	260	39-39-126
126	126			
126-126	39	258	261	126-126-39
39	39			
39-39	126			
39-39-126	126	260	262	39-39-126-126
126	39			
126-39	39	259	263	126-39-39
39	126			
39-126	126	257	264	39-126-126
126		126		