Parul® University

Data Compression

Mr. Prashant Sahatiya, Assistant Professor
Information Technology Engineering

Parul®
University

DIGITAL LEARNING CONTENT

The Course Outline

Chapter 1 : Compression Techniques

Chapter 2: Huffman coding algorithm

Chapter 3: Arithmetic Coding

Chapter 4: Scalar Quantization

Chapter 5: Vector Quantization

Parul®
University

DIGITAL LEARNING CONTENT

CHAPTER-1

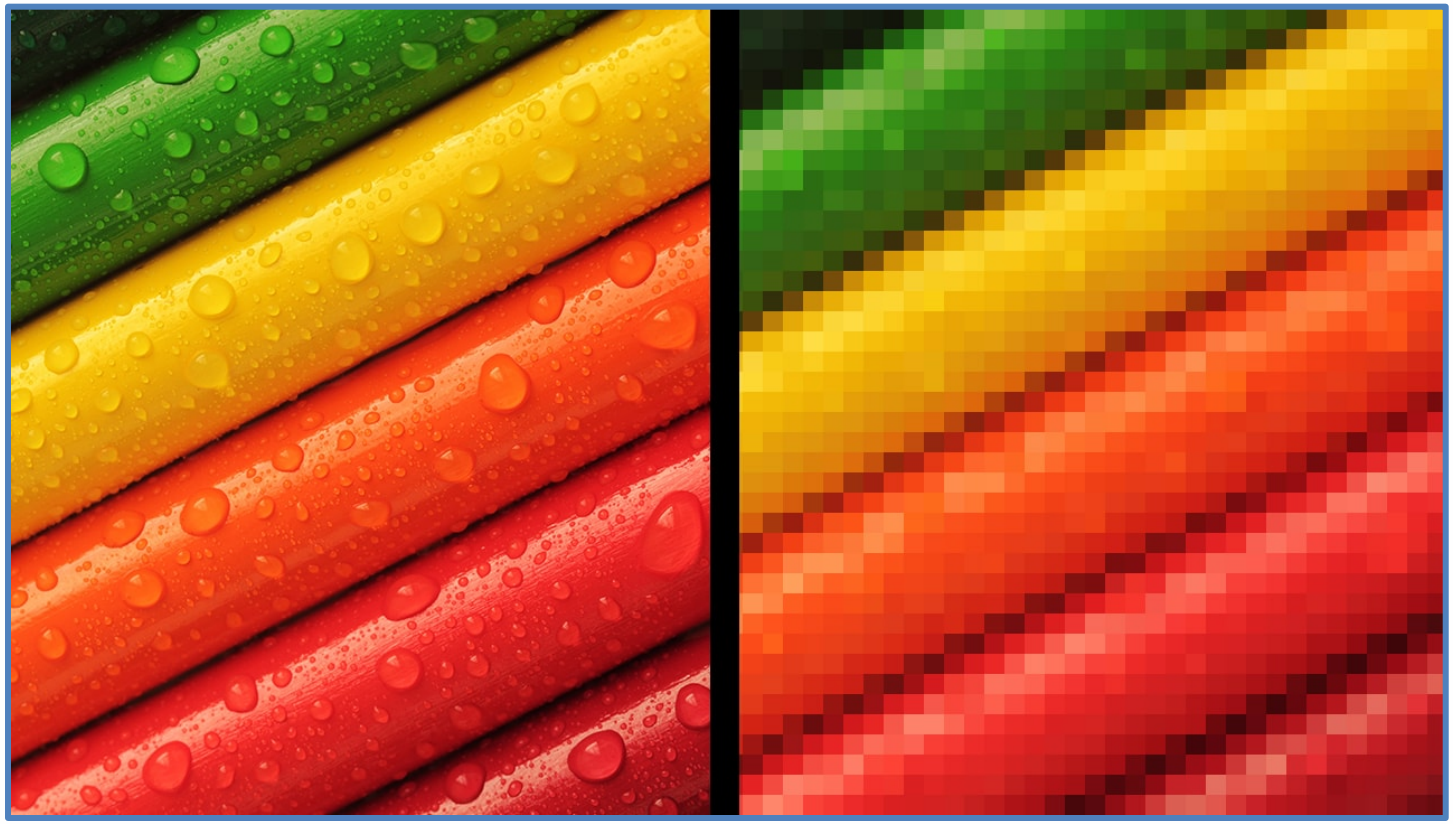
Compression Techniques

What is Data Compression?

- Data compression is used everywhere. Many different file types use compressed data. Without data compression a 3-minute song would be over 100Mb in size, while a 10-minute video would be over 1Gb in size. Data compression shrinks big files into much smaller ones. It does this by getting rid of unnecessary data while retaining the information in the file.

- Data compression can be expressed as a decrease in the number of bits required to illustrate data. Compressing data can conserve storage capacity, accelerate file transfer, and minimise costs for hardware storage and network capacity.

What is Data Compression?



How Compression Works?

- Compression is executed by a program that uses a procedure to identify how to reduce the data size.

- Text compression can be done by eliminating unnecessary characters, embedding a repeat character to specify repeated characters, and substituting a smaller bit string for a commonly occurring bit string. Data compression can cut a text file to 50%, or to a percentage still smaller of its original size.

- For data transmission, compression can be done on the data content or on the transmission unit as a whole. When data needs to be transferred over the internet, larger files can be sent in a ZIP, GZIP or other compressed format.

What is the Purpose of Compression?

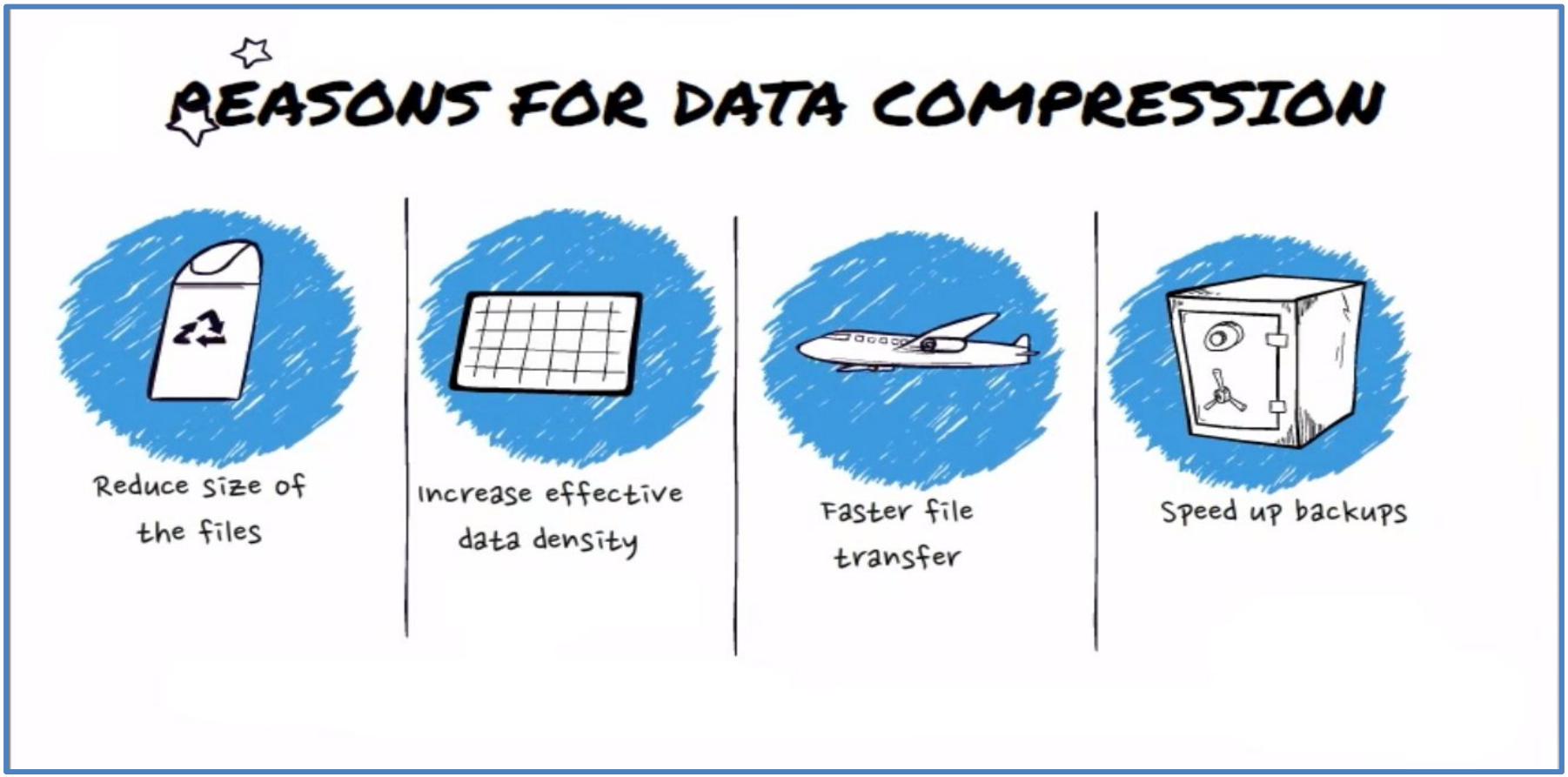
- The purpose of compression is to make a file, message, or any other chunk of data smaller. Data compression can significantly decrease the amount of storage space a file takes up.
- If we had a 10Mb file and could shrink it down to 5Mb, we have compressed it with a compression ratio of 2, since it is half the size of the original file. If we compressed the 10Mb file to 1Mb it would have a compression ratio of 10 because the new file is a 10th the size of the original.
- The higher the compression ratio the better the compression. Because of compression, administrators save money and time that would otherwise be spent on storage.

What is the Purpose of Compression?

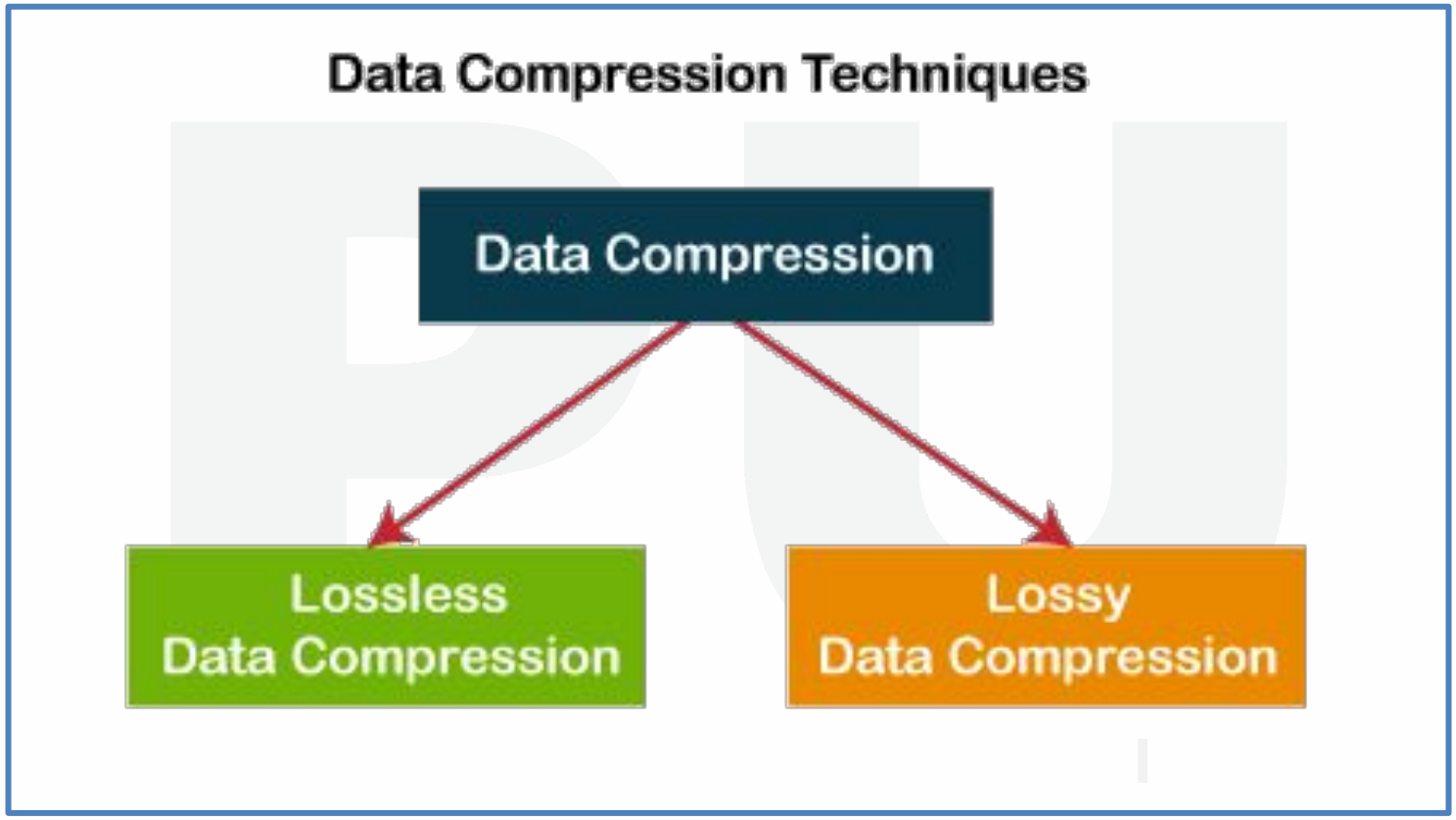
- Compression enhances backup storage operation and has also affected primary storage data reduction. Compression will continue to play a significant role in data reduction as data continues its own exponential growth.

- Almost any type of file can be compressed, but it's imperative to follow best practices when selecting files to compress. For example, some files are already compressed, so compressing them would not have a substantial impact.

What is the Purpose of Compression?

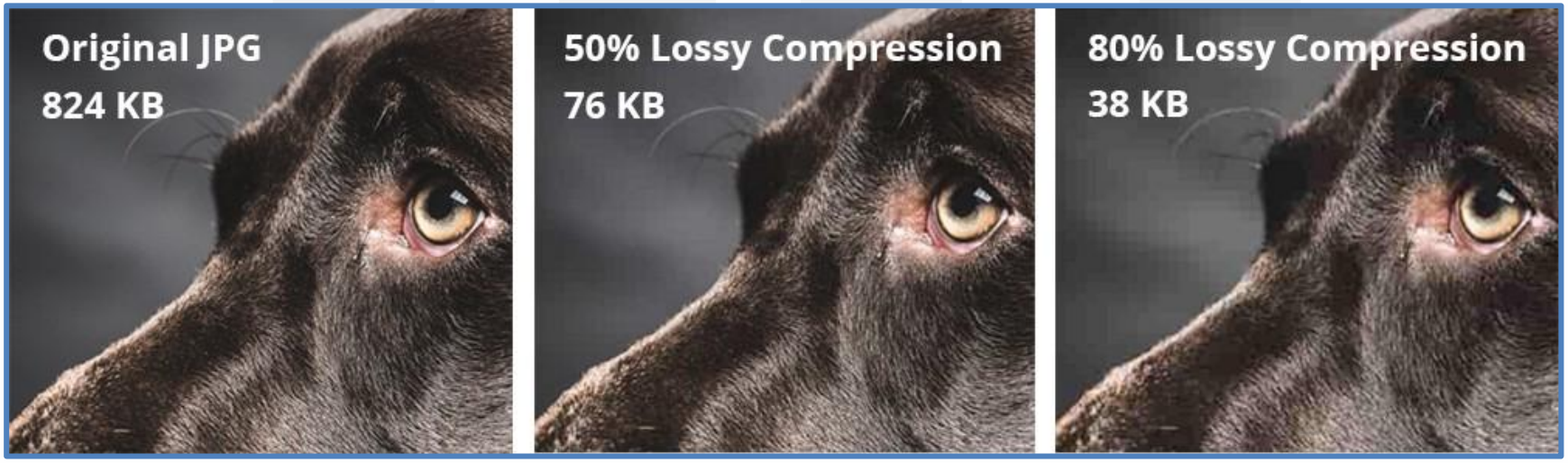


Data Compression Methods



Lossy Compression

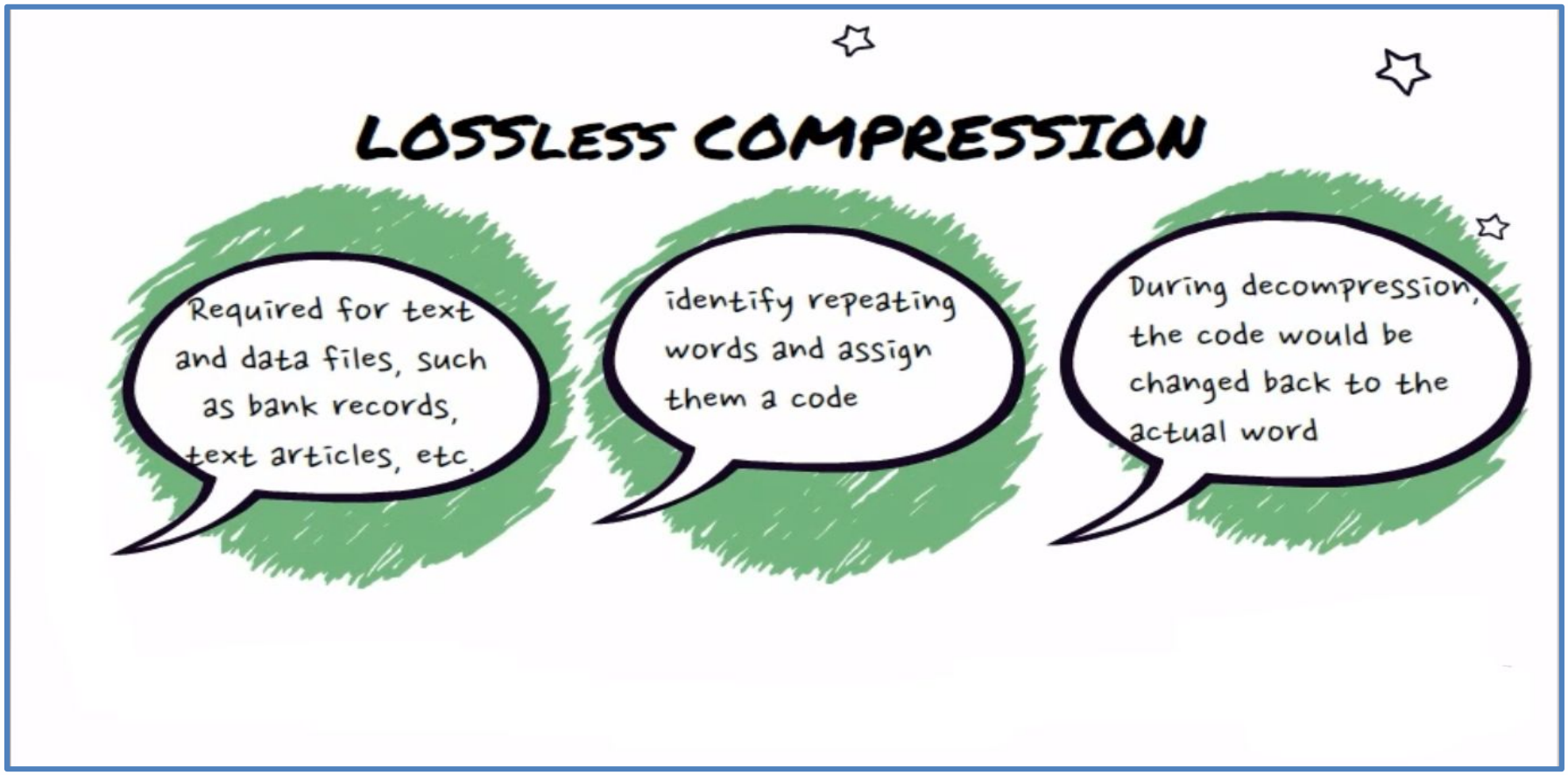
- **Lossy compression** loses data, while lossless compression keeps all the data. With lossless compression we don't get rid of any data. Instead, the technique is based on finding smarter ways to encode the data. With lossy compression we get rid of data, which is why we need to distinguish data from information.



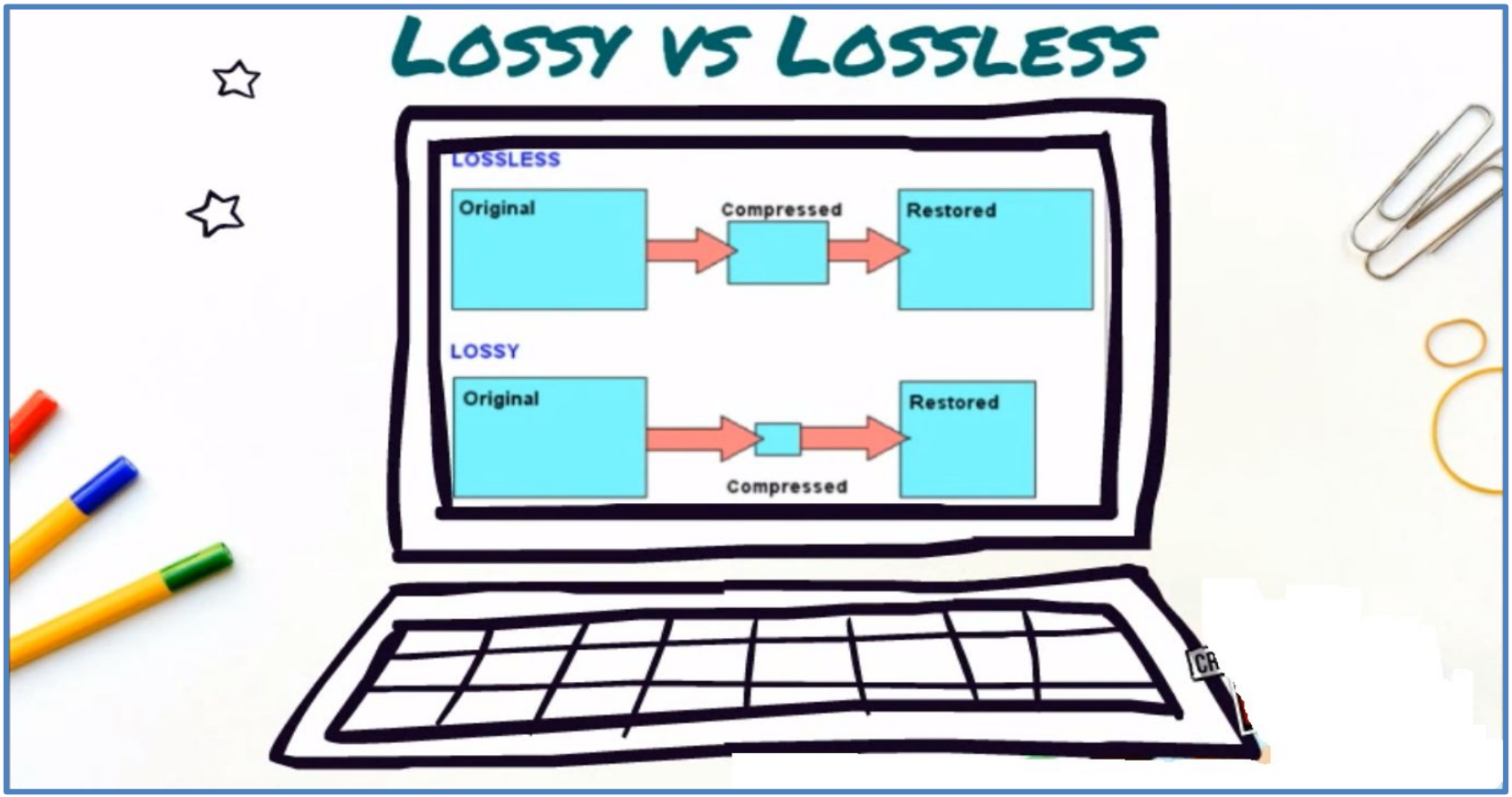
Lossless Compression

- **Lossless compression** allows the potential for a file to return to its original size, without the loss of a single bit of data, when the file is uncompressed. Lossless compression is the usual approach taken with executables, as well as with text and spreadsheet files, where the loss of words or numbers would change the information.
- Lossless compression can compress the data whenever redundancy is present. Therefore, lossless compression takes advantage of data redundancy.

Lossless Compression



Difference between



Difference between

Lossy Compression	Lossless Compression
Lossy compression is the method which eliminate the data which is not noticeable.	While Lossless Compression does not eliminate the data which is not noticeable.
In Lossy compression, A file does not restore or rebuilt in its original form.	While in Lossless Compression, A file can be restored in its original form.
In Lossy compression, Data's quality is compromised.	But Lossless Compression does not compromise the data's quality.
Lossy compression reduces the size of data.	But Lossless Compression does not reduce the size of data.

Difference between

Lossy Compression	Lossless Compression
Algorithms used in **Lossy** compression are: Transform coding, Discrete Cosine Transform, Discrete Wavelet Transform, fractal compression etc.	Algorithms used in **Lossless** compression are: Run Length Encoding, Lempel-Ziv-Welch, Huffman Coding, Arithmetic encoding etc.
Lossy compression is used in Images, audio, video.	Lossless Compression is used in Text, images, sound.
Lossy compression has more data-holding capacity.	Lossless Compression has less data-holding capacity than Lossy compression technique.
Lossy compression is also termed as irreversible compression.	Lossless Compression is also termed as reversible compression.

Measures of Performance of Data Compression Techniques

- Following are the measure of performance of Data Compression:
  (i) Compression ratio
  (ii) Distortion
  (iii) Compression rate
  (iv) Fidelity and Quality
  (v) Self Information

Measures of Performance of Data Compression Techniques

- **Compression ratio:** It is very logical way of measuring how well a compression algorithm compresses a given set of data is to look at the ratio of the number of bits required to represent the data before compression to the number of bits required to represent the data after compression.

- This ratio is called ‘Compression ratio’. Ex. Suppose storing n image requires 65536 bytes, this image is compressed and the compressed version requires 16384 bytes. So the compression ratio is 4:1. It can be also represented in terms of reduction in the amount of data required as a percentage i.e 75%

Measures of Performance of Data Compression Techniques

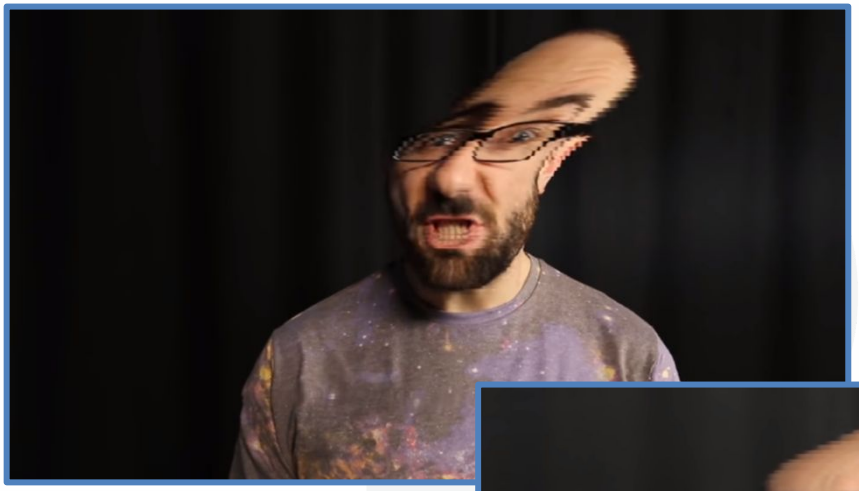
- **Distortion:** In order to determine the efficiency of a compression algorithm, we have to have same way of quantifying the difference. The difference between the original and the reconstruction is called as 'Distortion'. Lossy techniques are generally used for the compression of data that originate as analog signals, such as speech and video.

- In compression of speech and video, the final arbiter of quality is human. Because human responses are difficult to model mathematically, many approximate measures of distortion are used to determine the quality of the reconstructed waveforms.

Parul®
University

DIGITAL LEARNING CONTENT

Measures of Performance of Data Compression Techniques



Measures of Performance of Data Compression Techniques

- **Compression rate:** It is the average number of bits required to represent a single sample. Ex. In the case of the compressed image if we assume 8 bits per byte (or pixel) the average number of bits per pixel in the compressed representation is 2.

- Thus we would say that the compression rate is 2 bits/ pixel.

Measures of Performance of Data Compression Techniques

- **Fidelity and Quality:** The difference between the reconstruction and the original are fidelity and quality. When we say that the fidelity or quality of a reconstruction is high, we mean that the difference between the reconstruction and the original is small.

- Whether the difference is a mathematical or a perceptual difference should be evident from the context.

Measures of Performance of Data Compression Techniques

- **Self Information:**
- Event 1: The sun will rise from the east tomorrow. (Obvious)
- Event 2: The electricity will go after one hour. (Unsure)
- Event 3: There will be heavy snowfall at Dessert in 2021. (Unsure)

- If the probability of an event is low, the amount of self-information associated with it is high. If the probability of an event is high, the information associated with it is low.

Measures of Performance of Data Compression Techniques

- **Self Information:** Shannon defined a quantity called Self – Information. Suppose we have an event A, which is set of outcomes of some random experiment. If P(A) is the probability that event A will occur, then the self-information associated with A is given by:

$$i(A) = log_b = -log_b P(A) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

- Ex. The barking of a dog during a burglary is a high probability event and therefore does not contain too much information theory. However if the dog did not bark furring a burglary, this is a low-probability event and contains a lot of information.

Parul®
University

Information Theory

Information theory studies the quantification, storage, and communication of **information**. It was originally proposed by Claude Shannon in 1948 to find fundamental limits on signal processing and communication operations such as data compression, in a landmark paper titled "A Mathematical **Theory** of Communication“.

Models in Data Compression

1.Physical Models: If we know something about the physics of the data generation process, we can use that information to construct a model.

For Ex. In speech- related applications, knowledge about the physics of speech production can be used to construct a mathematical model for the sampled speech process. Sampled speech can be encoded using this model.

Real life Application: Residential electrical meter readings

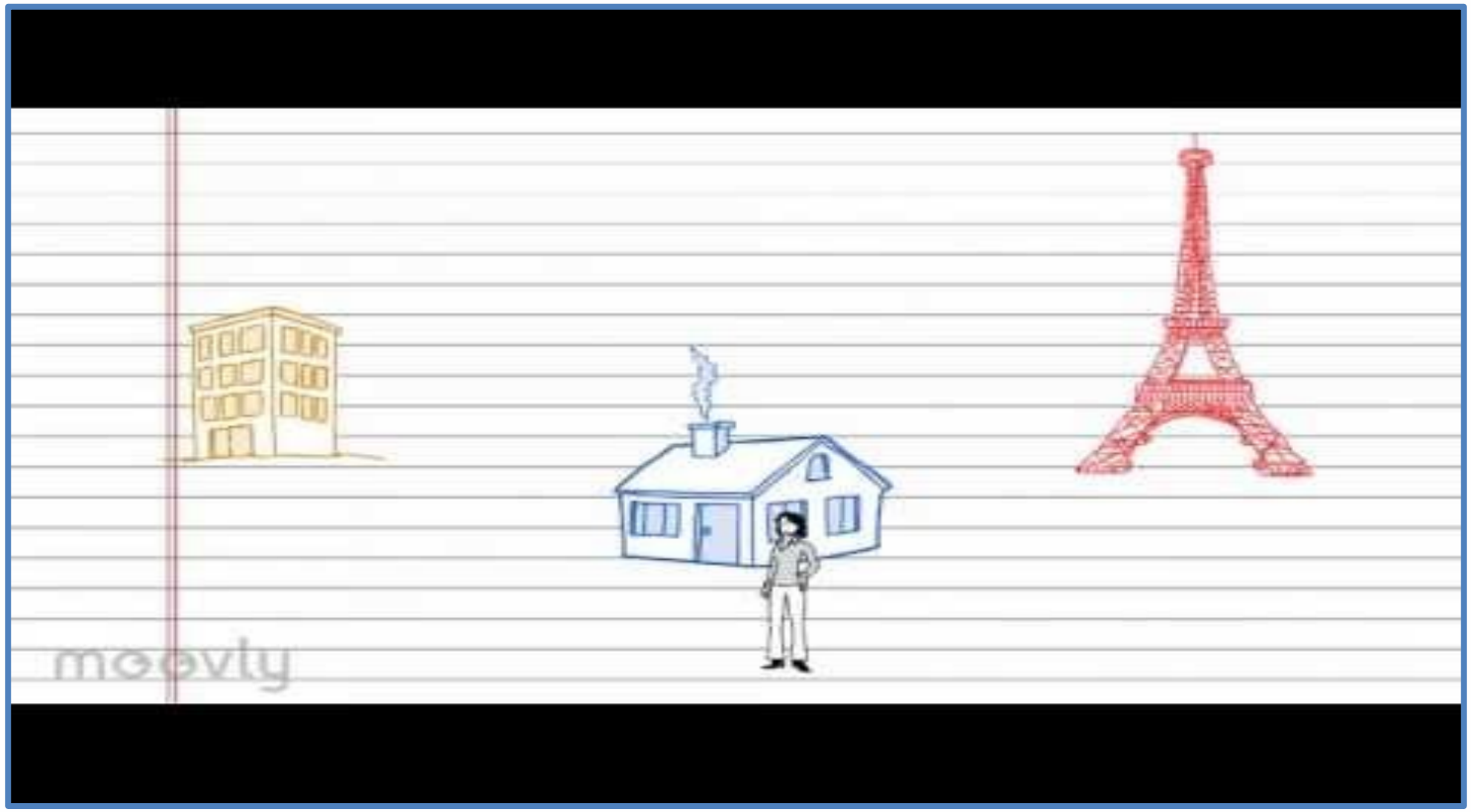
Models in Data Compression

2. Probability Models: The simplest statistical model for the source is to assume that each letter that is generated by the source is independent of every other letter, and each occurs with the same probability. We could call this the ignorance model as it would generation be useful only when we know nothing about the source.

The next step up in complexity is to keep the independence assumption but remove the equal probability assumption and assign a probability of occurrence to each letter in the alphabet.

Models in Data Compression

3. Markov Models:



Models in Data Compression

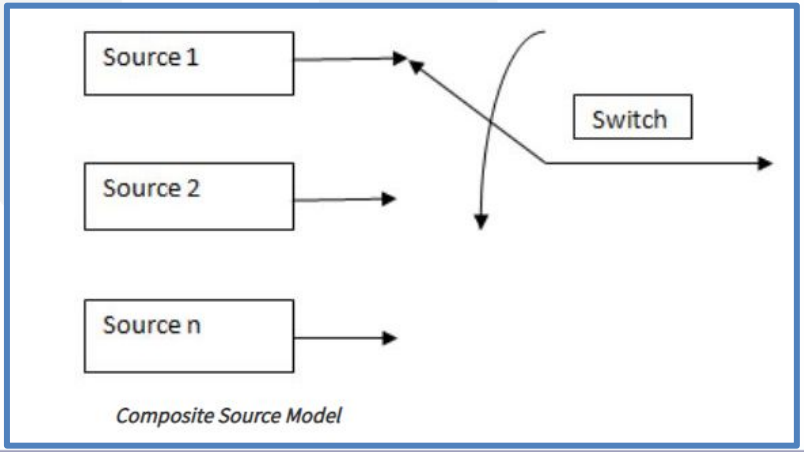
3. Markov Models: Markov models are particularly useful in text compression, where the probability of the next letter is heavily influenced by the preceding letters. In current text compression, the kth order Markov Models are more widely known as finite context models, with the word context being used for what we have earlier defined as state. Consider the word ‘preceding’. Suppose we have already processed ‘preceding’ and we are going to encode the next ladder.

If we take no account of the context and treat each letter a surprise, the probability of letter ‘g’ occurring is relatively low. If we use a 1st order Markov Model or single letter context we can see that the probability of g would increase substantially. As we increase the context size (go from n to in to din and so on), the probability of the alphabet becomes more and more skewed which results in lower entropy.

Models in Data Compression

4. Composite Source Model: In many applications it is not easy to use a single model to describe the source. In such cases, we can define a composite source, which can be viewed as a combination or composition of several sources, with only one source being active at any given time.

A composite source can be represented as a number of individual sources S_i , each with its own model M_i and a switch that selects a source S_i with probability P_i. This is an exceptionally rich model and can be used to describe some very complicated processes.



Composite Source Model

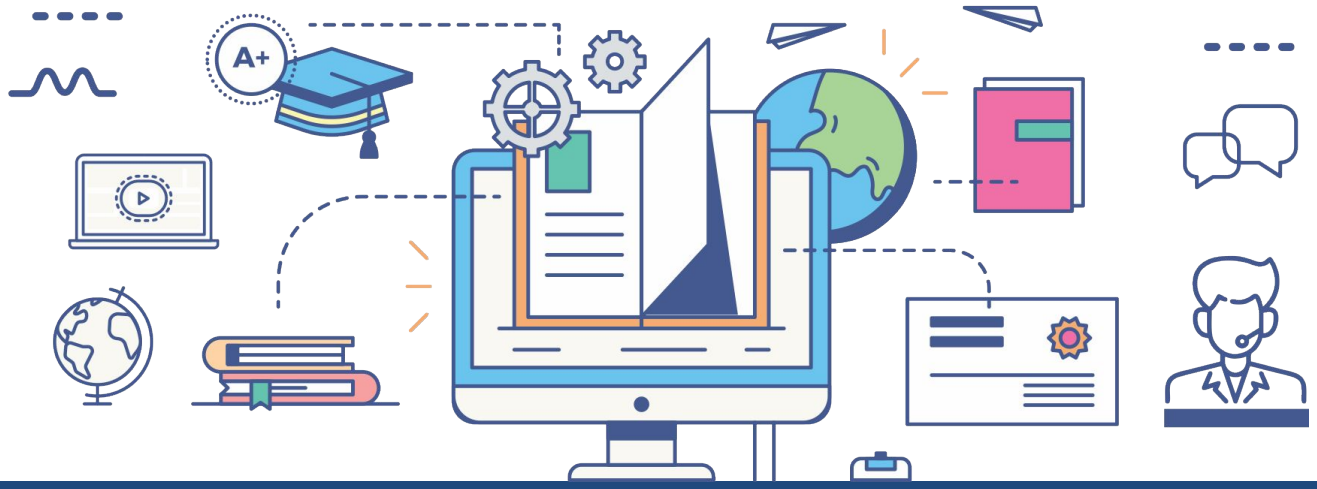
Coding in Data Compression

- i. Coding is an assignment of binary sequences to elements of an alphabet.
- ii. The set of binary sequences is called a code, and the individual members of the set are called codewords.
- iii. An alphabet is a collection of symbols called letters. Ex. The ASCII code for letter ‘a’ is 1000011, the letter ‘A’ is coded as 1000001, and the letter ‘b’ is coded as 0011010.

Coding in Data Compression

- iv. The ASCII code uses the same number of bits to represent different symbols. If we use fewer bits to represent symbols that occur more often, on the average we would use fewer bits per symbol.
- v. The average number of bits per symbol is often called the rate of the code. The idea of using fewer bits to represent symbols that occur more often is the same idea that is used in Morse Code.
- vi. The codewords for letters that occur more frequently are shorter than for letters that occur less frequently

DIGITAL LEARNING CONTENT



Parul® University

www.paruluniversity.ac.in