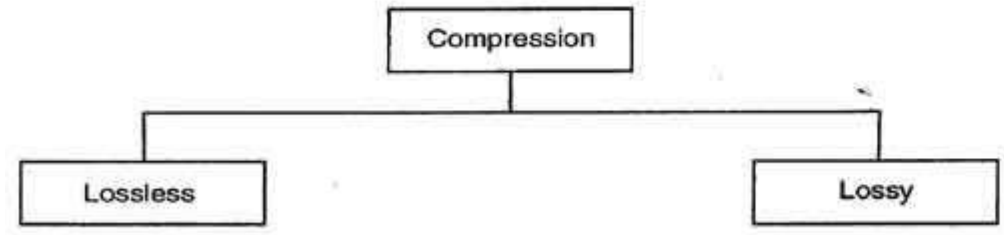
Types of compression: The two types of compression are:



Types of Compression

Lossy Compression - where data bytes are removed from the file. This results in a smaller file, but also lower quality. It makes use of data redundancies and human perception – for example, removing data that cannot be perceived by humans. So whilst quality might be affected, the substance of the file is still present. Lossy compression would be commonly used over the internet, where large files present a problem. An example of lossy compression is "mp3" compression, which removes wavelength extremes which are out of the hearing range of normal people. MP3 has a compression ratio of 11:1. Another example would be JPEG (Joint Photographics Expert Group), which is used to compress images. JPEG works by grouping pixels of an image which have similar colour or brightness, and changing them all to a uniform, "average" colour, and then replaces the similar pixels with codes.

The **Lossy compression** method eliminates some amount of data that is not noticeable. This technique does not allow a file to restore in its original form but significantly reduces the size. The lossy compression technique is beneficial if the quality of the data is not your priority. It slightly degrades the quality of the file or data but is convenient when one wants to send or store the data. This type of data compression is used for organic data like audio signals and images.

Lossy Compression Technique

- **Transform coding:** This method transforms the pixels which are correlated in a representation into disassociated pixels. The new size is usually lesser than the original size and reduces the redundancy of the representation.
- **Discrete Cosine Transform (DCT):** This is the most vastly used image compression technique. JPEG process is centers around DCT. DCT process divides the images into distinct parts of frequencies. In the quantization step, where compression basically occurs least important frequencies are rejected. And the critical frequencies are retained so that the image can be obtained in the decompression process. The reconstructed image could contain some distortion.
- **Discrete Wavelet Transform (DWT):** It provides a location of time and frequencies simultaneously and can be utilized in decomposing a signal into component wavelets.

Lossless Compression - Lossless compression is compression where no data is lost and the quality of the original file remains. This works by using an algorithm that exploits data redundancies. These achieve lower compression ratio than lossy compression, but no data is lost and the quality stays the same. In a text file, for example, common words or other data might be replaced by preset symbols so that when the file is decompressed, it can be 'expanded' out, retaining the original data. Common uses would be computer programs where lossy compression could result in bugs and legal documents where no loss is allowed. A common lossless compression file format is .GIF. It achieves compression ratios of about 2:1. It is best used for drawing and line art, as it works by replacing patterns.

The **Lossless compression** method is capable of reconstituting the original form of the data. The quality of the data is not compromised. This technique allows a file to restore its original form. Lossless compression can be applied to any file format can improve the performance of the compression ratio.

Lossless Compression Technique

- **Run Length Encoding (RLE):** This technique reduces the frequency of repeating symbols in a string by using a special marker at the beginning of the symbol.
- **Lempel-Ziv-Welch (LZW):** This technique also works similar to RLE technique and searches for the repeating strings or words and stores them in variables. It then uses a pointer at the place of the string, and the pointer points the variable in which string is stored.
- **Huffman Coding:** This technique handles data compression of ASCII characters. It constructs a full binary tree for various symbols after computing the probability of each symbol and place it in descending order.

Data Compression refers to a technique where a large file to reduced to smaller sized file and can be decompressed again to the large file. Lossy compression restores the large file to its original form with loss of some data which can be considered as not-noticeable while lossless compression restores the large file to its original form without any loss of data.

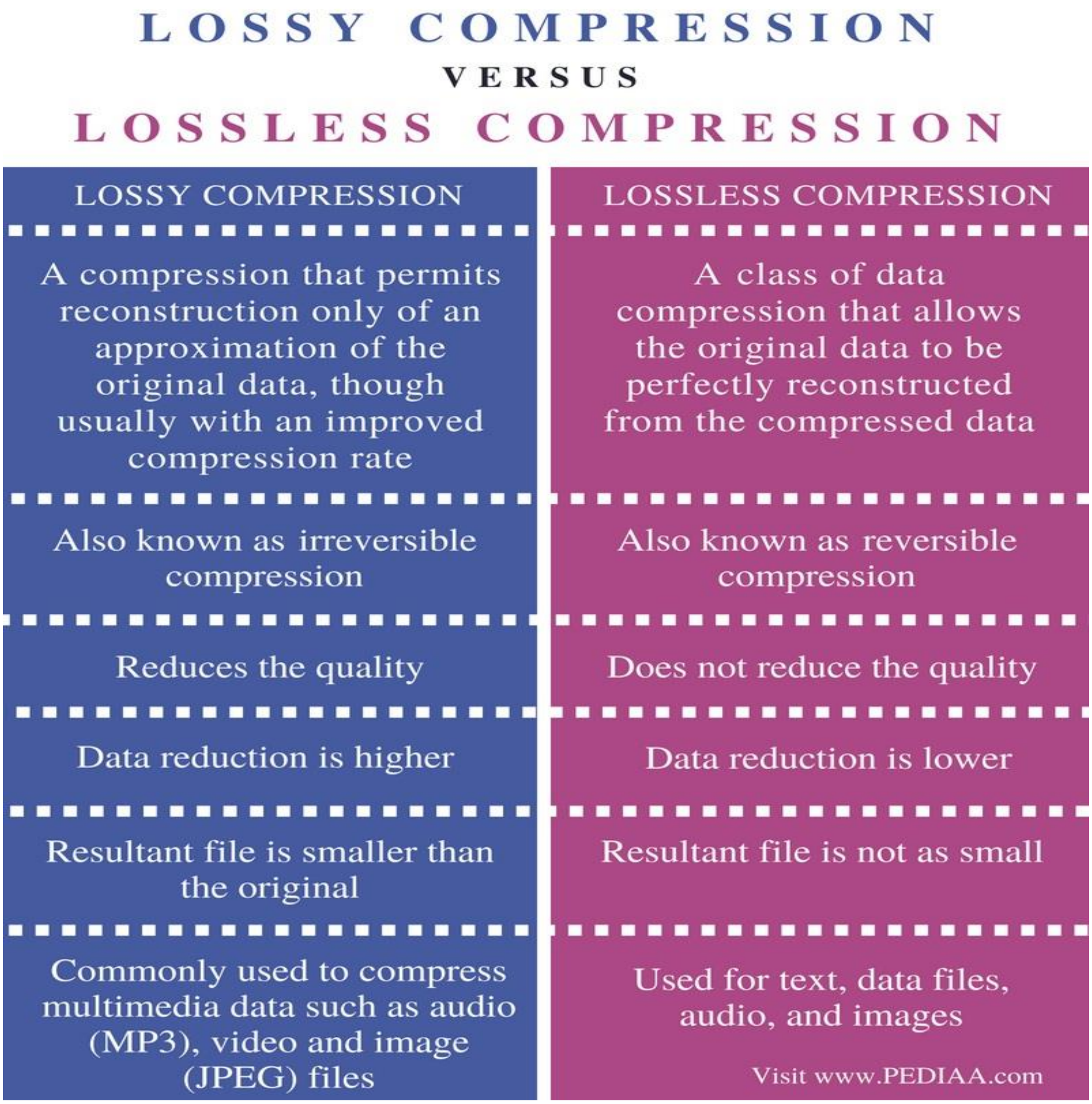
Decompression

In order to use a compressed file, you must first decompress it. The software used to decompress depends on how the file was compressed in the first place. To decompress a .zip file you need software, such as WinZip. To decompress a .sit file, you need the Stuff it Expander program. WinZip does not decompress .sit files, but one version of Stuff It Expander can decompress both .zip and .sit files. Files ending in .sea or .exe are called self-extracting files. These are compressed files that do not require any special software to decompress. Just click on the file and it will automatically decompress and open.

Decompression is the process of restoring compressed data to its original form. Data decompression is required in almost all cases of compressed data, including lossy and lossless compression. Similar to compression of data, decompression of data is also based on different algorithms. Decompression is considered important, as compressed data needs to be restored back to standard state for usage. Decompression is widely used in data communications, multimedia, audio, video and file transmissions. Decompression is also known as un-compression.

Differences between Lossy Compression and Lossless Compression:

Sr. No.	Key	Lossy Compression	Lossless Compression
1	Data Elimination	Lossy compression eliminates those bytes which are considered as not-noticable.	Lossless compression keeps even those bytes which are not-noticable.
2	Restoration	After lossy compression, a file cannot be restored to its original form.	After lossless compression, a file can be restored to its original form.
3	Quality	Lossy compression leads to compromise with quality.	No quality degradation happens in lossless compression.
4	Size	Lossy compression reduces the size of file to large extent.	Lossless compression reduces the size but less as compared to lossy compression.
5	Algorithm used	Transform coding, Discrete Cosine Transform, Discrete Wavelet transform, fractal compression etc.	Run length encoding, Lempel-Ziv-Welch, Huffman Coding, Arithmetic encoding etc.
6	Uses	Lossy compression is used to compress audio, video and images.	Lossless compression is used to compress text, images and sound.
7	Capacity	Lossy compression technique has high data holding capacity.	Lossless compression has low data holding capacity as compared to lossy compression.



LOSSY COMPRESSION VERSUS LOSSLESS COMPRESSION

LOSSY COMPRESSION	LOSSLESS COMPRESSION
A compression that permits reconstruction only of an approximation of the original data, though usually with an improved compression rate	A class of data compression that allows the original data to be perfectly reconstructed from the compressed data
Also known as irreversible compression	Also known as reversible compression
Reduces the quality	Does not reduce the quality
Data reduction is higher	Data reduction is lower
Resultant file is smaller than the original	Resultant file is not as small
Commonly used to compress multimedia data such as audio (MP3), video and image (JPEG) files	Used for text, data files, audio, and images

Visit www.PEDIAA.com

<u>Evaluating and Visibility:</u>

When an application programmer is deciding which compression algorithm to use, there are a number of factors that must be balanced:

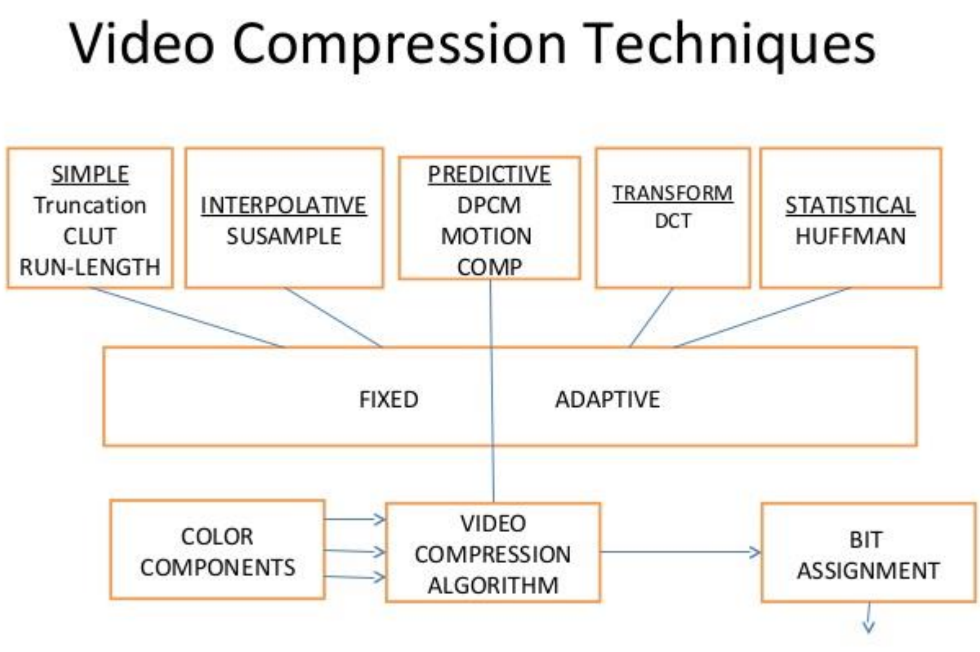
- Speed: how fast is it?
- Compression ratio: There's really no point if it doesn't make our data smaller than the uncompressed size.
- Complexity: How big is the executable? (This has both hard costs -- how much ROM or disk space does the software use? - as well as soft costs - if I need to change it, how long does it take to go through all the source code to find the right thing to change, and once changed, how long does it take to test every part of the code to make sure the change doesn't break something else?)

- Space: how much RAM does it need to run?
- Latency: how long do I have to wait before I hear the first note of the song?
- Interoperability: Will the files I generate be readable by any standard archive utility?

When a compression library programmer has tweaked a compression algorithm and trying to decide if it is really better or if he should revert to the previous version, he uses the same criteria.

VIDEO COMPRESSION/DECOMPRESSION TECHNIQUES

When used to convey multimedia transmissions, video streams contain a huge amount of data that requires a large bandwidth and subsequent storage space. As a result of the huge bandwidth and storage requirements, digital video is compressed in order to reduce its storage or transmitting capacity. This technology (video compression) reduces redundancies in spatial and temporal directions. Spatial reduction physically reduces the size of the video data by selectively discarding up to a fourth or more of unneeded parts of the original data in a frame. Temporal reduction, Inter-frame delta compression or motion compression, significantly reduces the amount of data needed to store a video frame by encoding only the pixels that change between consecutive frames in a sequence. Several important standards like Moving Picture Experts Group (MPEG) standard, H.261, 263 and 264 standards are the most commonly used techniques for video compression.



H.261: It was developed in 1990 by the International Telecommunication Union (ITU) developed the H.261 standard for data rates that are multiples of 64 Kbps. H.261 standard uses motion compensated temporal prediction. It supports two resolutions, namely, Common Interface Format (CIF) with a frame size of 352x288 and quarter CIF (QCIF) with a frame size of 172x144. The coding algorithm is a hybrid of the following:

Inter-picture prediction: It removes temporal redundancy transform coding, removes spatial redundancy motion compensation and uses motion vectors to compensate.

A macro block, the basic unit of temporal coding, is used to represent a 16x16 pixel region. Each macro block is encoded using intra (I-coding) or predictive) P-coding. Motion prediction uses only the previous picture to minimize delay. H.261 is intended for carrying video over ISDN in teleconferencing applications such as videoconferencing and videophone conversations. H.261 is not suitable for usage in general digital video coding.

H.263: It was developed by the International Telecommunication Union (ITU) in 1996. It uses an encoding algorithm called test model (TMN), which is similar to that used by H.261 but with improved performance and error recovery leading to higher efficiency. It is optimized for coding at low bit rates. H.263 provides the same quality as H.261 but with half the number of bits. A block motion-compensated structure is used for encoding each picture into macro blocks. The functionality of H.263 is enhanced by features like: bi-directionally encoded B-frames, overlapped-block motion compensation on 8x8 blocks instead of 16x16 macroblocks, unrestricted motion vector range outside the picture boundary, arithmetic encoding and fractional-pixel motion-vector accuracy.

H.263 is like H.261, is not suitable for usage in general digital video coding. However, H.261 and 263 are a bit contradictory since they both lack some of the more advanced techniques to really provide efficient bandwidth use.

H.263+: It is an extension of H.263 with higher efficiency, improved error resilience and reduced delay. It allows negotiable additional modes, spatial and temporal scalability. H.263+ has enhanced features like:

- Reference picture re-sampling motion compensation and picture prediction
- Reduced resolution update mode that permits a high frame rate during rapid motion
- Independent segment decoding mode that prevents the propagation of errors from corrupt frames

- Modified quantization mode improves bit rate control by controlling step size to detect errors and reduce decoding complexity

MPEG-1: The first public standard for the Moving Picture Experts Group (MPEG) committee was the MPEG-1. MPEG-1 was approved in November 1991 and its first parts were released in 1993. It has no direct provision for interlaced video applications. MPEG frames are encoded in three different ways:

- **Intra-coded (I-frames):** Encoded as discrete frames (still frames), independent of adjacent frames
- **Predictive-coded (P-frames):** Encoded by prediction from a past I-frame or P-frame, resulting in a better compression ratio (smaller frame)
- **Bi-directional-predictive-coded (B-frame):** Encoded by prediction using a previous and a future frame of either I-frames or P-frames; offer the highest degree of compression

MPEG-1 decoding can be done in real time using a 350 MHz Pentium processor. It is also suitable for playback from CD-ROM.

MPEG-2: The MPEG-2 project was approved in November 1994, focused on extending the compression technique of MPEG-1 to cover larger pictures and higher quality at the expense of higher bandwidth usage. MPEG-2 is designed for digital television broadcasting applications that require a bit rate typically between 4 and 15 Mbps (up to 100 Mbps), such as Digital high definition TV (HDTV), Interactive Storage Media (ISM) and cable TV (CATV) Profiles and levels were introduced in MPEG-2. The profile defines the bit-stream scalability and the color space resolution. With scalability, it is possible to extract a lower bit stream to get a lower resolution or frame rate. The level defines the image resolution, the Y (Luminance) samples/sec, the number of video and audio layers for scalable profiles and the maximum bit-rate per profile. The MPEG compatibilities include upward (decode from lower resolution), downward (decode from higher resolution), forward (decode from previous generation encoding) and backward (decode from new generation encoding). The

MPEG-2 input data is interlaced making it compatible with the television scanning pattern that is interlaced.

The MPEG-2 is suitable for TV broadcast applications and high-quality archiving applications. It is not however designed for the internet, as it requires too much bandwidth.

MPEG-4: It was approved in October 1998 and it enables multimedia in low bit-rate networks and allows the user to interact with the objects. The objects represent aural, visual or audiovisual content that can be synthetic like interactive graphics applications or natural like in digital television. These objects can then be combined to form compound objects and multiplexed and synchronized to provide QoS during transmission. Media objects can be in any place in the coordinate system. Streamed data can be applied to media objects to change their attributes.

The MPEG-4 compression methods are used for texture mapping of 2-D and 3-D meshes, compression of time-varying streams and algorithms for spatial, temporal and quality scalability, images and video. Scalability is required for video transmission over heterogeneous networks so that the receiver obtains a full resolution display. The MPEG-4 provides a high coding efficiency for storage and transmission of audio-visual data at very low bit-rates About 5-64 Kbps is used for mobile or PSTN video applications and up to 2 Mbps for TV/film applications.

MPEG-7: It was approved in July 2001 to standardize a language to specify description schemes. The MPEG-7 is a different kind of standard as it is a multimedia content description standard and does not deal with the actual encoding of moving pictures and audio. With MPEG- 7, the content of the video is described and associated with the content itself, for example to allow fast and efficient searching in the material.

The MPEG-7 uses XML to store metadata and it can be attached to a time code in order to tag particular events in a stream. Although, MPEG-7 is independent of the actual encoding technique of the multimedia, the representation that is defined within MPEG-4, i.e., the representation of audio-visual data in terms of objects, is very well suited to the MPEG-7 standard. The MPEG-7 is relevant for video surveillance since it could be used for example to tag the contents and events of video streams for more intelligent processing in video management software or video analytics applications.

H.264/AVC: In early 1998, the Video Coding Experts Group (VCEG) ITU-T issued a call for proposals on a project called H.26L, with a target of doubling the coding efficiency in comparison to any other existing video coding standards for various applications. The Moving Picture Expert Group (MPEG) and the Video Coding Expert Group (VCEG) have developed a new and outstanding standard that promises to outperform the earlier MPEG-4 and H.263 standard. Even though the first draft design for the new standard was adopted in October 1999, it provides the most current balance between the coding efficiency, cost and implementation complexity. It has been finalized by the Joint Video Team (JVT) as the draft of the new coding standard for formal approval submission referred to as H.264/AVC and was approved by ITU-T in March 2003 (known also as MPEG-4 part 10). The standard is further designed to give lower latency as well as better quality for higher latency. In addition, all these improvements compared to previous standards were to come without increasing the complexity of design so much that it would be impractical or expensive to build applications and systems. An additional goal was to provide enough flexibility to allow the standard to be applied to a wide variety of applications: for both low and high bit rates, for low and high resolution video and with high and low demands on latency. The main features that improve coding efficiency are the following :

- Variable block-size motion compensation with the block size as small as 4x4 pixels
- Quarter-sample motion vector accuracy
- Motion vectors over picture boundaries
- Multiple reference picture motion compensation
- In-the-loop deblocking filtering
- Small block-size transformation (4x4 block transform)

- Enhanced entropy coding methods (Context- Adaptive Variable-Length Coding (CAVLC) and Context Adaptive Binary Arithmetic Coding (CABAC))

COMPARISON OF VIDEO COMPRESSION METHODS

Video compression standards provide a number of benefits, of which the foremost is ensuring interoperability, or communication between encoders and decoders made by different people or different companies. In these way standards lower the risk for both consumer and manufacturer and this can lead to quicker acceptance and widespread use. In addition, these standards are designed for a large variety of applications and the resulting economies of scale lead to reduced cost and further widespread use. The well known families of video compression standards, performed under the auspices of the International Telecommunications Union-Telecommunications (ITU-T, formerly the International Telegraph and Telephone Consultative Committee, CCITT), the International Organization for Standardization (ISO) and the Moving Pictures Expert Group (MPEG) which was established by the ISO in 1988 to develop a standard for compressing moving pictures (video) and associated audio on digital storage media. The first video compression standard to gain widespread acceptance was the H.261 standard. The H.261 and 263 standards are suitable for carrying video over ISDN. They are used for video delivery over low bandwidths. The MPEG standards provide a range of compression formats that are suitable for applications that require higher bit rates. The MPEG-1 provides compression for standard VHS quality video compression. The MPEG-2 meets the requirements of applications with bit rates up to 100 Mbps and can easily cater for digital television broadcasting applications.

MPEG-1 and 2 are used for broadcast and CD-ROM applications, but unsuitable for the Internet. The MPEG-4 is suitable for low bit-rate applications such as video conferencing as it provides a high coding efficiency for storage and transmission. The MPEG-4 applications include Internet multimedia, interactive video, video conferencing, videophone, wireless multimedia and database services over ATM networks. H.263 and MPEG-4 are used for video delivery over low bandwidths. To cater for the high bandwidth requirements for the Internet, codes must have high bandwidth scalability, lower complexity and tolerance to losses, as well as lower latency for interactive applications.

MPEG-7 addresses this problem as it caters for both real-time and non-real time applications and enables retrieval of multimedia data files from the Internet. If the available network bandwidth is limited, or if a video is to be recorded at a high frame rate and there are storage space restraints, MPEG may be the preferred option. It provides a relatively high image quality at a lower bit-rate (bandwidth usage). Still, the lower bandwidth demands come at the cost of higher complexity in encoding and decoding, which in turn contributes to a higher latency when compared to motion.

H.264/AVC is now a widely adopted standard and represents the first time that the ITU, ISO and IEC have come together on a common, international standard for video compression. H.264 entails significant improvements in coding efficiency, latency, complexity and robustness. It provides new possibilities for creating better video encoders and decoders that provide higher quality video streams at maintained bit-rates (compared to previous standards), or, conversely, the same quality video at a lower bit-rate.

REETA

24/03/2020