CMPSCI 754 Multimedia Systems **Spring 2001**

Lecture 2: February 1

Lecturer: Prashant Shenoy *Scribe: Bhuvan Urgaonkar*

2.1 Need for Compression

Digitization of video streams yields a sequence of discrete samples called frames or images. The storage of uncompressed images or video requires considerable capacity. Consider the following examples:

- An image of a 180x180 sq. km area taken from a satellite with each pixel representing an area of 30 sq. meters takes 600 MB/image.
- NTSC video has a data rate of 30 frames/sec, each frame is 640x480 pixels and each pixel is represented using 3 bytes of data — a total of 30 MB/sec.

In order to provide feasible and cost effective solutions, most multimedia systems handle compressed images/digital video or audio streams.

2.2 Compression Fundamentals

The following properties are desired of a good compression algorithm:

- The encoding and decoding processes should be fast.
- It should enable us to achieve high compression ratios (the ratio of the size of the uncompressed multimedia object to the compressed object).
- The object obtained after decoding should be close enough to the original object.
- The algorithm should handle different sizes and frame rates equally well.

Compression Algorithms can be classified into two types - Lossless and Lossy. Clearly, Lossy algorithms can provide much higher compression ratios.

Entropy encoding ignores the semantics of input data and compresses media streams by regarding them as sequence of bits. *Source encoding* optimizes the compression ratio by considering media specific characteristics. Described below are some fundamental compression techniques. All commercial compression tools use one or more of these.

- **Run-length Encoding** Replace successive occurrences of a symbol by the number of times the symbol occurred and the symbol, e.g aaaaa is replaced by 5a.
- **Huffman Encoding** Use variable length codes with more frequently occurring symbols represented by fewer bits. The important thing to keep in mind is that the bit string assigned to any symbol shouldn't be the prefix of the bit string assigned to another symbol.

- **Predictive Encoding** Instead of encoding the values of the pixels, this type of encoding employs a predictor for the value of the pixel and encodes the difference between the actual value and the predicted value. The idea is to have a predictor such that the differences are much smaller than the actual values and hence can be encoded using fewer bits.

- **Transform Encoding** The basic idea is this: An NxN image is subdivided into several nxn *blocks*. Each block undergoes a transform that concentrates most of the information in the block into a few of the transform coefficients. Many of the coefficients contain little information and may be dropped.

2.3 JPEG Compression Standard

JPEG (Joint Photographic Experts Group) is a joint project of ISO/IEC JTC1/SC2/WG10 and the commission Q.16 of CCITT SGVIII. In 1992 JPEG became an ISO International Standard. JPEG is now the de facto standard for compression and decompression of still images. It applies to color and gray-scaled still images. This standard fulfills the following requirements in order to guarantee a further distribution and application of JPEG:

- The JPEG implementation should be independent of image size.

- Should be applicable to any image and pixel aspect ratio.

- The color representation itself should be independent of the special implementation.

- The image content may be of any complexity and with any statistical characteristics.

- It should be the state of art regarding the compression factor and the image quality achieved.

- The processing complexity must permit a software solution to run on as many as possible available standard processors.

- Sequential decoding(line by line) and progressive decoding(refining the whole image) should be possible. A lossless, hierarchical coding of the same image with different resolutions similar to the photo CD images should be supported.

The basic steps in JPEG encoding are the following:

- **Image transformation** JPEG employs the Forward Discrete Cosine Transformation (FDCT), which is one of the best known transforms. The original image is partitioned into 8x8 pixel blocks and FDCT is applied to each of these. FDCT enhances compression by concentrating most of the energy in the signal in the lower spatial frequencies. Though this transform is lossless in theory, in practice the computation of cosine functions leads to the loss of some accuracy. Note that JPEG does not specify a unique algorithm — any algorithm for FDCT may be used.

- **Quantization** This is the lossy step. Each of the 64 DCT coefficients is quantized using a 64-element quantization table. This table is carefully determined based on source image characteristics, display characteristics, viewing distance, etc.

- **Encoding Quantized Images** Finally, the quantized image is further compressed using one of the encoding techniques described earlier such as run-length coding or Huffman coding.

References

- R. STEINMETZ, Data Compression in multimedia computing - standards and systems, *Multimedia Systems*, 1994, pp. 187–204.