Univ-MLV

Laszlo Marak

On image compression

Contents

1 Introduction

Image compression is the application of data compression on digital images. In effect, the objective is to reduce redundancy of the image data in order to be able to store or transmit data in an efficient form.

Image compression can be lossy or lossless. Lossless compression is sometimes preferred for artificial images such as technical drawings, icons or comics. This is because lossy compression methods, especially when used at low bit rates, introduce compression artifacts. Lossless compression methods may also be preferred for high value content, such as medical imagery or image scans made for archival purposes. Lossy methods are especially suitable for natural images such as photos in applications where minor (sometimes imperceptible) loss of fidelity is acceptable to achieve a substantial reduction in bit rate.

A lossy compression in itself is a very general term. It only refers to the fact, that the uncompressed image will not be equal with the original. The type of the information "deleted" from the original image can be different from application to application. In this report we will try to discuss some basic ideas on lossy image compression methods applicable on a sequence of images. The report is divided into three main parts. Firstly we try to look at compression functions, then the human visual system, and finally we try to explain some basic ideas and theories. In this report we will focus on the compression theory and less attention is given to the programmability or the optimization of the compression.

2 Compression

The theoretical background of compression is provided by information theory (which is closely related to algorithmic information theory) and by rate-distortion theory. These fields of study were essentially created by Claude Shannon, who published fundamental papers on the topic in the late 1940s and early 1950s. Cryptography and coding theory are also closely related. The idea of data compression is deeply connected with statistical inference.

By definition every invertible function is a lossless compression algorithm. The objective of compression is to reduce the file size. However there is no invertible function which compresses every file with at least a bit. So no matter which compression function or algorithm we choose there will always be some files which are not compressible. That's why designing a compression algorithm based on global properties of the data is very difficult. Most of the compression functions and algorithms are based on the local properties.

2.1 Lossless compression

Many lossless data compression systems can be viewed in terms of a four-stage model. Lossy data compression systems typically include even more stages, including, for example, prediction, frequency transformation, and quantization.

The **Lempel-Ziv** (LZ) compression methods are among the most popular algorithms for lossless storage. DEFLATE is a variation on LZ which is optimized for decompression speed and compression ratio, although compression can be slow. DEFLATE is used in PKZIP, gzip and PNG. LZW (**Lempel-Ziv-Welch**) is used in GIF images. Also noteworthy are the LZR (**LZ-Renau**) methods, which serve as the basis of the Zip method.

The very best compressors use probabilistic models whose predictions are coupled to an algorithm called **arithmetic coding**. Arithmetic coding, invented by Jorma Rissanen, and turned into a practical method by Witten, Neal, and Cleary, achieves superior compression to the better-known **Huffman algorithm**, and lends itself especially well to adaptive data compression tasks where the predictions are strongly context-dependent. Arithmetic coding is used in the bilevel image-compression standard JBIG, and the document-compression standard DjVu. The text entry system, Dasher, is an inverse-arithmetic-coder.

2.2 Lossy compression

The general idea of lossy image compression is that a binary image has some information which the human eye can't see (or don't care for). The main requirement of a lossy compression is that it shall keep the most useful information possible. There is no generally correct measure for the quality of a lossy compression (other than the human eye) but there are several distance functions defined like "Peak signal-to-noise ratio" (PSNR) or "Structural SIMilarity" (SSIM). Both have been designed to create an objective measure of compression quality. SSIM for example is a full featured distance function used in measure theory. The distance between two images is a decimal number between 0 and 1. With these distance we can estimate the quality of the compression function.

3 The human visual system

Vision and hearing are the two most important means by which humans perceive the outside world. Human vision can be divided into two sub-categories. The low-level vision is the physical way our eye processes the light. It is easier to model. The high-level vision is the theoretical **method, algorithm or function** how our eye interpolates, filters, segments and understands the images. Description of the high-level vision is among the opened problems of the computer science.

3.1 Low-level vision

Light is the electromagnetic radiation that stimulates our visual response. It is expressed as a spectral energy distribution $L(\lambda)$, where λ is the wavelength that

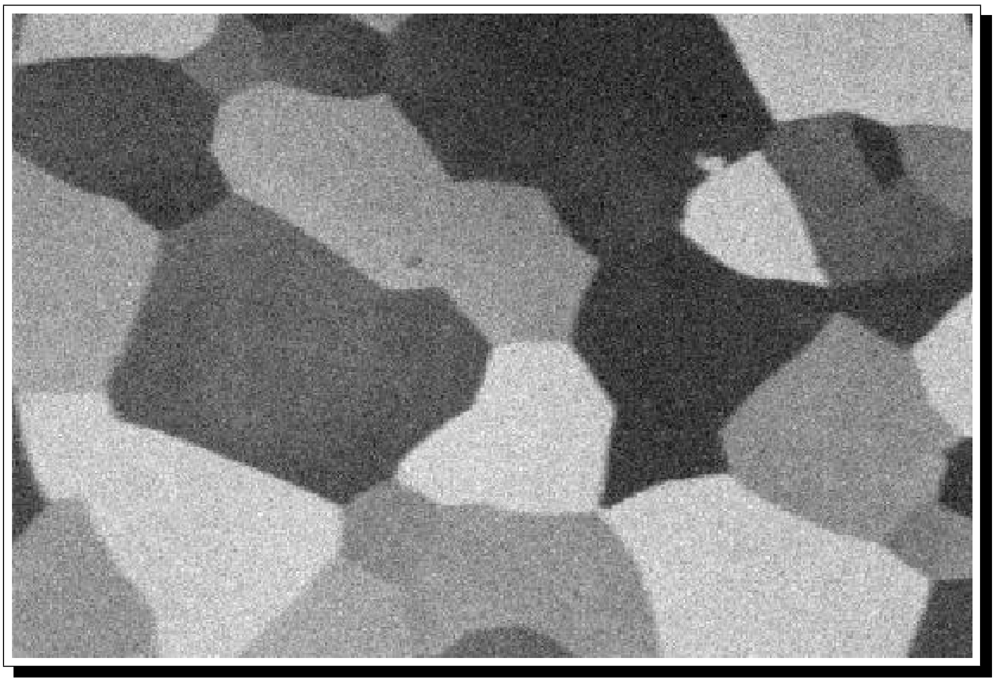

Figure 3.1: The grainy appearance due to noise is more pronounced in the darker uniform regions than in lighter regions. This proves that eye is also sensitive to variation of intensity, not just the power of it.

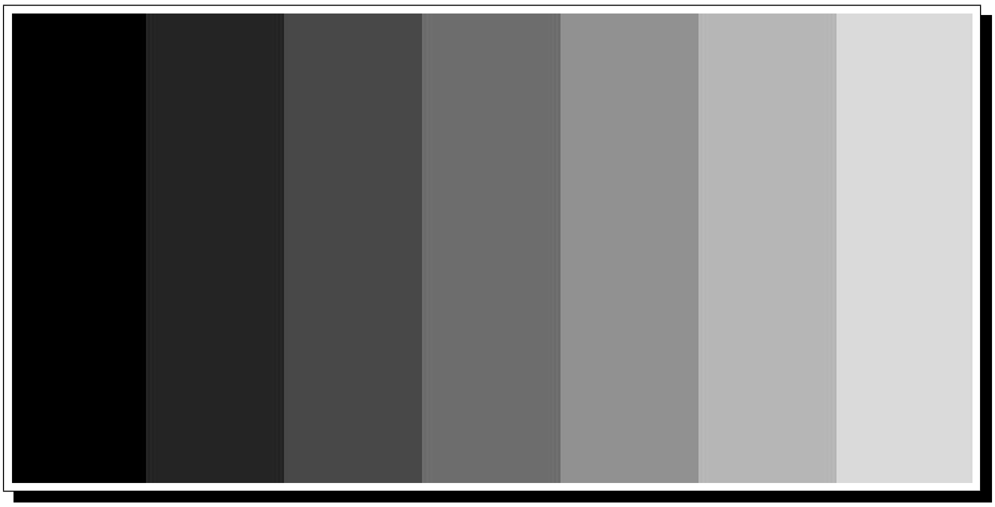

Figure 3.2: The spatial interaction of luminance from an object and its surrounds creates a phenomenon called the **Mach band** effect, which shows that brightness is not a monotonic function of luminance. In the bar chart, each bar has constant intensity. However, for a given bar the region looks brighter towards the left and darker towards the right.

lies in the visible region, 350nm to 780nm. Light received from an object can be written as

$$I(\lambda) = \rho(\lambda)L(\lambda)$$

where $\rho(\lambda)$ represents the reflectivity or transmissivity of the object and $L(\lambda)$ is the incident energy distribution. The illumination range over which the human eye can operate is roughly 1 to 1010, or ten orders of magnitude.

The retina of the human eye contains about 100 million **rods** (100 mega-pixels) and 6.5 million **cones**. The rods are sensitive, and provide vision the lower several orders of magnitude of illumination. The cones are less sensitive, and provide the visual response at the higher 5 to 6 orders of magnitude of illumination. The cones are also responsible for **color vision**. They are densely packed in the center of

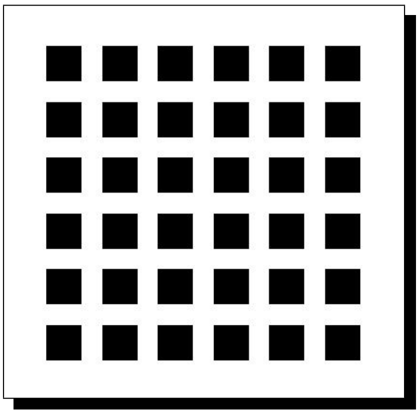

Figure 3.3: Similar effect can be observed as on image 3.2. Although we know that it is not the case, the white areas between the black squares appear with varying shades of grey.

the retina (the fovea), at a density of approximately 120 cones per degree of arc subtended in the field of vision. The density of cones falls off rapidly outside a circle of 1 degree radius from the fovea. The pupil of the eye acts is an aperture, and therefore acts as a **low-pass filter**. In bright light the passband is about 60 cycles per degree.

An effective model of the human eye must take into account lot of factors and can clearly become quite complex.

Aside from the response to grey levels, of which we can observe a few dozen, we have the ability to distinguish between thousands of col ours. The perceptual attributes of color are brightness, hue, and saturation:

- Brightness represents the perceived luminance, as mentioned before
- The hue of a color refers to its "redness", "greenness", and so on. For monochromatic light, differences in hue are manifested by the differences in wavelength
- Saturation is that aspect of perception that varies as more white light is added

to monochromatic light.

Colors are sensed by the eye by three different types of cones, which are sensitive to different wavelengths.

It is therefore possible in our perception of light to describe any color by three values. Note however that the light itself forms a continuous spectrum of wavelengths. The RGB color space expresses a color by its red, green, and blue components, so any color can be represented by a point within a 3-D cube. However, although the RGB space is good for acquisition or display of color information, it is not particularly good for explaining the perception of colors. Alternative color spaces more suited to this purpose are HSV (hue, saturation, value), YUV (chrominance, luminance), and Lab, although numerous others exist. It is generally possible to convert from one color space to another by applying a **nonlinear transformation** to the color values.

3.2 High-level vision

Aside from the low-level "hardware" aspects of the imaging process, the human visual system exhibits a considerable **cognitive component**. Vision is therefore also influenced by memory, context, and intention. There is evidence that these high-level components feed back to the low-level vision system, further complicating the issue.

It is also apparent that a considerable component of the human visual system relies on us having a certain knowledge regarding the appearance of objects in our environment.

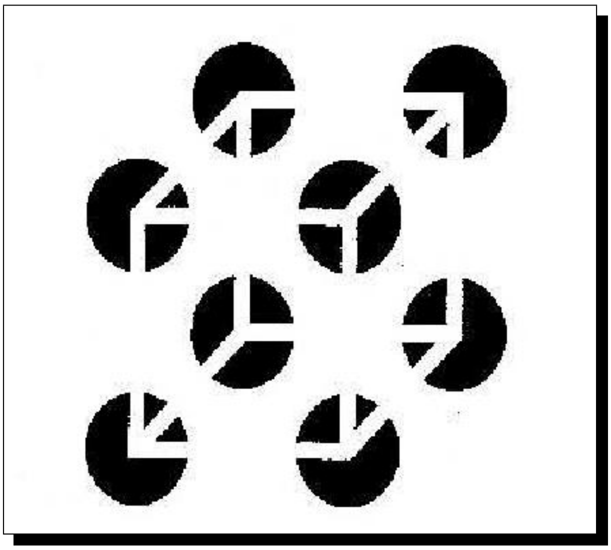

Figure 3.4: Subjective contours are an interesting aspect of the human visual system

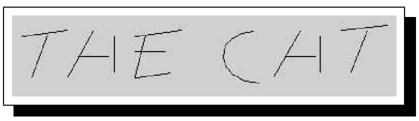

Figure 3.5: In this image the **H** of the article is identical with the **A** of the noun. However our brain distinghues them according to they neighborhood.

Figure 3.6: We know that edges are very important in vision, but in this image we recognize the curves of a dog even without a single edge.

4 Compression of image sequences or similar images

In section 3. we have seen, that from the images we can extract different types of information. These informations can also be classified according to their importance, furthermore **the image can be recostructed** from these informations. We assume that the amount of information, the image is reconstructed from is a continous function of the image quality. That is to say if we remove a "little" information from the image, the image looses a "little" of its quality.

We can image the lossy compression of still images as following: We **create a lossless representation of the image** with the information that the human visual system is sensitive to. We **sort this information** according to relevance and we **delete it from the least important** to the most inportant until we achieve the desired ratio.

If have a sequence of images or many images of the **same character**, before deleting we can compare this information. If the images are similar (for example

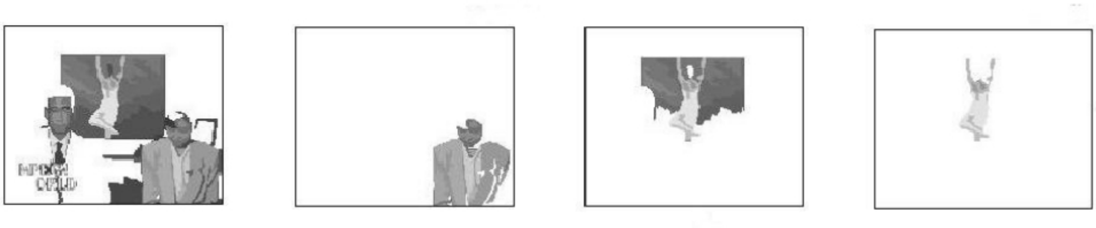

Figure 4.1: Image decomposed into main regions of interest. The background can be compressed with a more destructive function. Also the regions can be compared in the neighbor images and can be compressed according to the difference.

a person is speaking on the sequence), some information appears in every image (for example the background is not changing). In this case we can cummulate the information on all the images, sort it according to relevance, than proceed the same way as at the still images. In this case every similar information can be considered equal, appears only once, so we achieve higher compression ratios.

This method moves the problem of compression from problem of redundance to the problem of extracting information from the images. The literature of lossy image compression is too complex to represent it in whole, so we have choosen a few ideas, which is a great example, how understanding the images can help us design good compression functions.

4.1 Hierarchical region based processing

A possible approche of converting images into information is to **divide the images into regions**. Region processing is based on decomposing the images into subsets. With this approach we assume that different regions can have different importance for the human eye. Unimportant regions can than be downsampled or compressed with a more destructive function while important regions can be preserved.

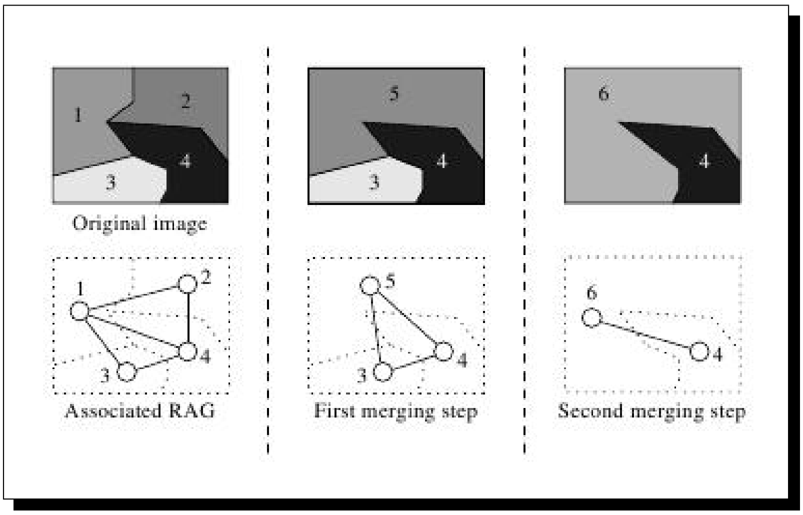

Figure 4.2: Region merging on an adjacency graph

4.1.1 Region adjacency graph

The region adjacency graph is the most well known region oriented structure. The Region Adjacency Graph is a graph which is constituted by a set of nodes representing regions of the space and a set of links connecting two spatially neighboring nodes. The region adjacency graph is usually used to represent a partition of the image. Note that a node of the graph can represent a region, a flat zone (a connected component of the space where the signal is constant) or even a single pixel.

Processing techniques relying on region adjacency graphs have mainly focussed on merging techniques. The graph is constructed based on an initial partition: each region of the partition image is associated to a node in the graph, and two nodes are connected if their associated regions are neighbors in the partition image. A merging algorithm on such a graph is simply an iterative process that removes some of the links and merges the corresponding nodes. The merging order, that is, the order in which the links are processed is usually based on a **similarity criterion between regions**. Such criterion may be based on color, frequency, SSIM, PSNR, etc. Each time a link is processed its associated nodes (i.e. regions) are merged together. After each merging the algorithm has to look for the links whose distance has to be recomputed. The merging ends once a termination criterion is reached. Most commonly used termination criterion is the number of nodes associated to the graph.

4.1.2 Region representation with trees

Another suitable structure for representing the images using regions relies on trees. **A tree is an acyclic graph**. Tree representations are well suited for representing data in a hierarchical way. The links of the tree indicate "containment" or "is composed of". One may take, for example, the table of contents of a book. The table of contents is usually divided in chapters, sections, subsections and so on. Assume we want to represent the table of contents using a tree structure. The title of the book can be considered as being associated to the root node of such tree. The chapters are associated to the children of the root node, whereas the sections are the children of the associated chapters nodes. Other possible examples of tree structures of every day use are, for example, the directory structures that are used to organize files on a hard disc. In a similar way, tree structures may be used in image analysis to represent the structure of the image in a hierarchical and compact way.

Assume that the images shown on top correspond to a decomposition of the original image into a set of regions.

Note that the region adjacency graph structure is oriented towards coding the neighborhood relationships between regions, whereas a tree structure is oriented towards the coding of inclusion relationship. As such, the tree is not able to code neighborhood relationship.

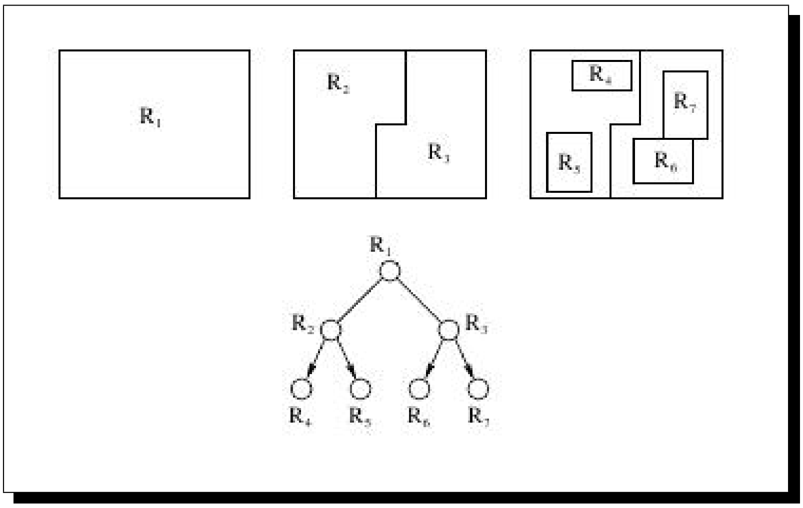

Figure 4.3: Example of tree structure. On top there is a possible decomposition of an image. The associated tree representation is shown on the bottom.

4.1.3 Partition tree

The Partition Tree is a structured representation of a set of hierarchical partitions obtained usually by means of a segmentation procedure. The partition tree is created from an initial partition of an image, plus a set of hierarchical partitions that are "above" and "below" the initial partition. The latter are created by using, respectively, merging and region splitting algorithms by means of a homogeneity criterion. The regions can be homogeneous either spatially (lower levels of the tree) or in motion (upper levels of tree). Thus, the partition tree defines a universe of possible regions the image is made of.

The Partition Tree is analyzed in order to select the best coding strategy among the set of regions that are represented by the Partition Tree and a list of coding techniques It defines the set of regions out of which the algorithm should create the final partition. The list of coding techniques deals with the coding of these regions. In practice, each region of the Partition Tree is coded by all coding techniques

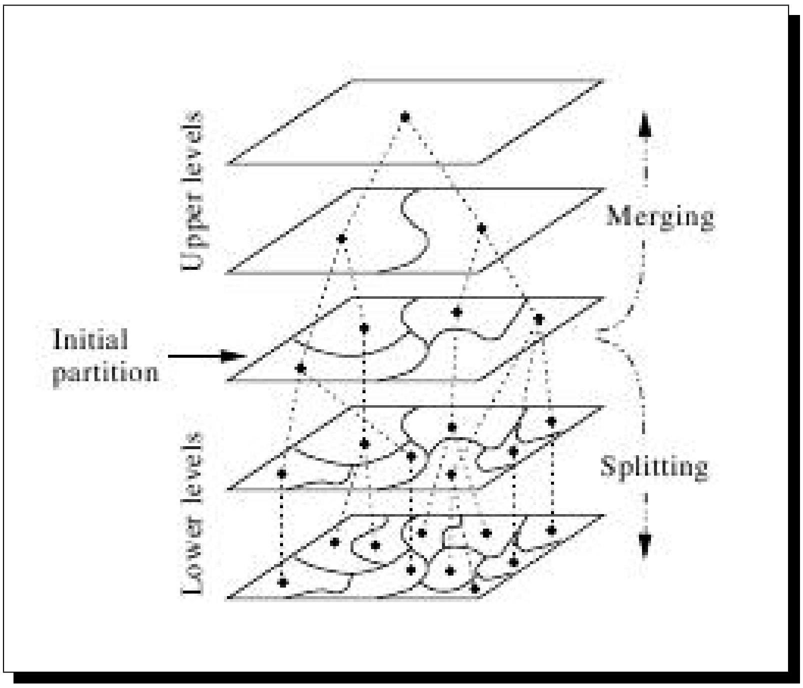

Figure 4.4: Example of a partition tree

and the corresponding rate distortion values are stored in a Decision Tree. In the formulation of the Rate-Distortion problem, it is assumed that the budget (in bits) is given for each frame. Based on this budged, the algorithm finds the coding strategy that minimizes the distortion. Once the Partition Tree has been constructed, several region coding techniques (defined by the set of coding techniques CT1 , . . . , CTn) are considered for each region. By means of the rate-distortion approach the best partition is defined. The present approach is used for a generic video coding algorithm.

5 Conclusion

We saw in the previous sections, that the images **can be represented by the information** they contain, and that this information **can be deleted separately**. According to the relevance of the information we can define predictable lossy compression functions.

These functions hold a huge potencial in video-search-engines, classifing and storing high amount of images or videos. They can also simplify transfer or protection from loss hence for the same cost of hardware the videos can be backed up several times.

The trade-off for this compression is that their usage is limited. To achieve their goal, they use many properties of the human visual system. If we would try to compress with them other type of data (text, voice, etc.) they would not be able to preserve their compression ratio and their compression quality would decrease.

References

[1] J. Serra J. Crespo and R.W. Schafer. Theoretical aspects of morphological filters by reconstruction. *Signal Processing*, 47(2):201–225, 1995.

[2] J. Serra C. Gratin J. Crespo, R.W. Shafer and F. Meyer. A flat zone approach: a general low-level region merging segmentation method. *EURASIP Signal Processing*, 62.

[3] Fred Nicolls. Human visual system. 2008.

[4] Luis Garrido Ostermann. *Hierarchical region based processing of images and video sequences*. PhD thesis, 2002.