"Analysis on Lossless Image Compression in Image Processing"

Ajay Babu Sriramoju

Programmer Analyst, Everest Consultancy Group Inc., USA, ajay.ramoju@gmail.com, 14129980025

Abstract – Image restitution is a very dynamic area of research. The main objective is to improve the general quality of an image or remove errors/defects. Different methods from linear algebra or partial differential equations are used for this rationale. Image restoration is the process of taking a corrupted/noisy image and preparing the clean original image. Corruption may come in various forms such as motion, blur and camera resolution problem. In addition to these blurring effects, noise always corrupts any recorded image.

Keywords - Effect, Image, Motion

---♦---

1. INTRODUCTION

The use of Geographical Information Systems (GIS) to provide users with digital navigation information is widespread and becoming more and more popular. Examples of this are the personal car navigator and PDA-based digital topographic maps for foresters, geologists and engineers. Usually the architecture of such systems does not depend on the application area. A typical example is a system where the user's coordinates are obtained via a satellite positioning service, such as Global Positioning System (GPS), and geographical information about the current location is obtained from a local or remote map database. Map images in a database can be stored in two principally different formats: vector and raster. The vector format assumes that the map is stored as a set of geometrical primitives (lines, symbols, curves, textures) describing the image content. Each primitive is described by a set of required parameters. For example, a straight line segment is described by four real numbers defining the end points. In order to be displayed, the data must be projected on a plane with the desired scale and rotation, and then drawn on the screen of the client device. However, some geographical data is still unavailable in vector form, and the only sources are traditional maps printed on paper sheets. Although vectorization is an actively developing technology (Ablameyko, Pridmore, 2000). (Chhabra, Dori (eds.) 2000). a universal non-supervised vectorization algorithm still does not exist. It is often too expensive to manually convert such data into vector format, and therefore storage in raster format can be a better solution. Raster format assumes that the image is stored as an array of values that represents the rectangular matrix of pixels forming the picture. Depending on the application, the storage of one pixel requires one bit for binary images, one byte for gray-scale or indexed images, three bytes for true color, and even more in the case of a multi-spectral image. A natural advantage of raster format is that it does not require any additional processing for displaying the image. The image can be represented immediately after the data is received. A typical way of combining vector and raster format in the same system is to use the global database stored data as vectors and provide the user with a raster image converted from the vector original to represent the area needed. The main drawback raster format is that it is not flexible when some transformation of the image is needed. For example, zooming, rotation and projection of the image are all impossible without degradation. The storage size needed is also a problem. In contrast to vectors, raster images store all pixels of the line instead of coordinates of the corresponding segment. In the case of geographical 2 maps, the size of digitized images can be huge. For example, in maps from the National Land Survey of Finland (NLS) topographic database (NLS), a single map sheet of 10×10 km2 1:20,000 scale is represented by a single image of 5000×5000 pixels, which requires about 70 Mbytes of memory to be stored. Another example is a map of A4 size scanned with 300 dpi in true color, which results in a 2500×3500 pixel image requiring about 25 Mbytes. The necessity for image compression is obvious since more efficient storage space utilization as well as faster map transmission is needed to make digital navigation services more usable, reliable and cheaper. Features that are distinctive for map images can be characterized as follows. A map image contains only a few unique colors; in cases where the image is converted from a vector source the number of colors rarely grows higher than several tens. A map image also contains a lot of uniform areas representing particular regions like water, forests or background. The areas of the map are usually distinctly separated from each other. This makes a map image which contains sharp and easily localized edges. Smooth gradation is rarely present in map images.

2. REVIEW OF LITERATURE

Recent advances in camera technology, computing hardware, and optical fabrication have led to the emergence of computational photography, a field exploring the joint design of optical light modulation and computational processing. While conventional cameras record two-dimensional images, the ultimate" computational camera would capture all visual information with a single shot. The amount of control over such high-dimensional data is unprecedented: dynamic range, depth of field, focus, colour gamut, and time scale of a photograph can be interactively controlled in post-processing. Required visual properties include the colour spectrum as well as spatial, temporal, and directional light variation the plenoptic function. Unfortunately, digital sensors optically integrate over the plenoptic dimensions; most of the desired information is irreversibly lost in the process. We explore the multiplexed acquisition of the plenoptic function in this study. For this purpose, we introduce a mathematical framework that models plenoptic light modulation and corresponding computational processing. This framework not only allows us to evaluate and optimize the optical components of computational cameras, but also subsequent reconstructions. The combined design of optical modulation and computational processing is not only useful for photography, displays benefit from similar ideas. Within this scope, we propose multi-layer architectures and corresponding optimization schemes for glasses-free 3D display. Compared to conventional Automultiscopic displays, our devices optimize brightness, resolution, and depth of field while preserving thin form factors. In a different application, adaptive coded apertures are introduced to projection displays as next-generation auto-iris systems. Combined with computational processing that exploits limitations of human perception, these systems increase the depth of field and temporal contrast of conventional projectors. With computational optics, integrated into sunglasses or car windshields, the capabilities of the human visual system can be extended. By optically modulating perceived intensities and colours, we demonstrate applications to contrast manipulation, preattentive object detection, and visual aids for the colour blind. Finally, we introduce computational probes as high-dimensional displays designed for computer vision applications, rather than for direct view. These probes optically encode refraction caused by transparent phenomena into observable changes in colour and intensity. Novel coding schemes enable single-shot reconstructions of transparent, refractive objects.

The rapid evolution of advanced wireless communication systems has been driven in recent years by the growth of wireless packet data applications such as mobile multimedia and wireless streaming services. In particular, image and video applications are among those services that facilitate communication beyond the traditional voice services. The large amount of data that is needed to represent the visual content and the scarce bandwidth of the wire- less channel constitute a difficult system design framework.

3. LOSSLESS IMAGE COMPRESSION

Images as a class can be of very different natures, structures and contents. Therefore, any successful compression technique is usually adapted to be applied on a particular type of images. Lossless image compression algorithms can be organized into three groups: continuous-tone, discrete-tone and universal algorithms. The compression algorithms referred as continuous-tone are optimized to perform on natural imagery, usually photographic or other types of images obtained with a physical sensor. Discrete-tone algorithms are designed to perform on other types of images that contain fewer colors and less gradation, with more sharp edges and uniform areas. Images of this type are mostly of an artificial nature such as web graphics, engineering drawings, maps and circuits. Universal compression algorithms are usually applied when the type of the data is not predefined. Popular universal compression techniques are based on various adaptations of a classical dictionary-based LZ77 or LZ78 algorithm. For example, CompuServe Graphics Interchange Format (GIF, 1990) which is widely used for the compression of palette images, uses LZC (Thomas, *et. al.,* 1985). improvement of LZW (Welch, 1984). The Portable Network Graphics (PNG) algorithm (Boutell), which was proposed as the replacement for the relatively old GIF, uses DEFLATE (Deutsch, 1991) algorithm. It uses a combination of LZ77 and Huffman coding. The ITU Group 4 algorithm incorporated in Tagged Image File Format (TIFF) uses simple data compression techniques based on run-length coding, prefix coding and differential relative address designate (READ) coding to utilize line to line correlations. 6 However, universal compression algorithms suffer from the one-dimensional nature of the method, and thus present relatively low compression efficiency.

4. LOSSY COMPRESSION

Most popular lossy algorithms are used for compression of photographic imagery since the nature of the human eye's perception allows significant reduction of information in the image without any subjective loss of quality. However, in some applications the properties of the input imagery can differ significantly from natural photography, thus requiring different compression principles to be applied.

The classical examples of popular lossy compression algorithms are Joint Photographic Expert Group (JPEG) and a more recent standard JPEG2000. These algorithms are based on image transforms: discrete cosine transform (DCT) for JPEG and wavelet transform for JPEG2000. The transform coefficients are rounded and quantized causing partial loss of information. These algorithms are optimized for compression of photographic images, which are mostly used in computer industry. There are also transform-based algorithms optimized for different tasks such as Enhanced Compression Wavelets (ECW) and Multiresolution Seamless Image Database (MrSID) . These are commercial solutions for the compression of aerial and satellite photos. DCT and especially wavelet based algorithms present excellent compression efficiency in terms of compression vs. degradation tradeoff for the class of images to which they were optimized. The DjVu algorithm was proposed for lossy compression of scanned imagery containing text and line drawings, especially scanned books. This algorithm utilizes the fact that scanned images of that type contain a lot of sharp edges and details, which are difficult to represent by DCT or wavelets. The algorithm therefore 11 separates the image into two parts: text and background, and applies different compressors for each. The binary context-based algorithm JB2 is a variant of JBIG2 standard and is applied for text. The low resolution wavelet-based IW44 is proposed to compress the background. Lossy predictive coding is also used for the near-lossless compression when the degree of imposed degradation is limited. Lossy predictive coding assumes that the prediction error is not encoded precisely but quantized, thereby causing minor errors when the image sample is reconstructed. This technique is used in JPEG-LS near-lossless mode, for example. Quantization of signal can also be seen as an approach of lossy compression. Reducing the number of unique colors (or gray scale gradations) in the image imposes distortion, and at the same time, reduces the informational content of the image, thus improving its compressibility. For example, the GIF standard operates only on indexed palette images requiring quantizing colors to a predefined palette (typically 256-color) before the compression. The impact of quantization on compression efficiency has been studied in several papers.

CONCLUSION

Image quality improvement/enhancement has been a concern throughout the area of image processing. Images are affected undesirable because it degrades image quality. An application of reduction in an image processing is a promising research fields. The accurate prediction of quality from an end-user perspective has received increased attention with the growing demand for compression and communication of digital image and video services over wired and wireless networks. The existing quality assessment methods and metrics have a vast reach from computational and memory numerical methods to highly complex models incorporating aspects of the human visual system. It is hence crucial to classify these methods in order the favorable approach for an intended application.

REFERENCE

A. K. Chhabra, D. Dori (eds.) (2000). Graphics Recognition – Recent Advances, Lecture Notes on Computer Science, vol. 1941, Springer, Verlag.

CIE (1986). Colorimetry, CIE Pub. No. 15.2, Centr.Bureau CIE, Vienna, Austria.

D. Salomon (2004). Data Compression: The complete reference, 3rd Edition, Springer Verlag.

Graphics Interchange Format (sm) (1990). Version 89a, http://www.dcs.ed.ac.uk/home/mxr/gfx/2d/GIF89a.txt.

I.H. Witten, A. Moffat, T.C. Bell (1999). Managing Gigabytes: Compressing and Indexing Documents and Images. 2nd Edition, Morgan Kaufmann.

K. Sayood (2000). Introduction to Data Compression, 2nd edition, Academic Press.

NLS: National Land Survey of Finland, Opastinsilta 12 C, P.O.Box 84, 00521 Helsinki, Finland. http://www.nls.fi/indexfie.html.

P. Deutsch (1991). "DEFLATE compressed data format specification", RFC1951, http://www.faqs.org/rfcs/rfc1951.html.

S. Ablameyko, T. Pridmore (2000). Machine Interpretation of Line Drawing Images, Springer.

S. W. Thomas, J. Mckie, S. Davies, K. Turkowski, J. A. Woods, and J. W. Orost (1985). Compress (version 4.0) program and documentation.

T. Boutell, “PNG (Portable Network Graphics) specification”, ftp://ftp.uu.net/graphics/png/documents/

T. Welch (1984). “A technique for high-performance data compression”, Computer Magazine, vol.17(6), pp. 8-19.