Scalable Visual Information in Multimedia*

Andrew Dorrell and David Lowe
School of Elec. Eng., University of Technology, Sydney
P.O. Box 123 Broadway 2007 Sydney Australia
email:{andrew,dwl}@ee.uts.edu.au

Abstract

In this paper we present a useful generalisation of the concept of data format for image and video. Whereas image and video formats for multimedia are usually motivated by the need for compression or interfacing with display hardware, we propose they be motivated by the operations to be performed on the data. By first defining a set of operations required for interactive multimedia we are able to determine a set of key properties required of an image or video format for use in multimedia. We focus on one of these properties—scalability—which has the potential to greatly assist in the better integration of image and video data in interactive multimedia applications.

1 Introduction

The inclusion (in multimedia applications) of visual information in the form of still images and video has potential to significantly increase their communicative power. However, providing a similar level of integration and interactivity as can be achieved with other media (such as text and graphics) still remains an obstacle. One reason for this is the failure of image and video formats to support anything more than the most basic operations. To clarify this we note that most of the processing and interaction currently provided by multimedia applications is based on the “raw” format data. Although the provision of operations for other formats such as JPEG has been studied to some degree (Smith & Rowe 1993, Arman, Hsu & Chiu 1993) it is still fair to say that in most systems, the only operation available for JPEG format data is “decode”. Obviously, if operations could be performed on compressed format data directly, many computational savings could be made. But we may take this idea a step further by observing that the image or video format defines the representation of the data which is most easily accessible to an application. To understand the importance of this observation we need to first understand the relationship between the data, its representation and the operations performed on that data by application software.

In computer systems, there is an inextricable relationship between algorithms (corresponding to processing tasks) and the data they act on. It is both efficient and appropriate to select a format (or representation) for data which is suited to the algorithms most commonly performed on the data. As an example, it we were required to uniformly brighten an image by adding a constant value to all of its pixels. If we were to perform this operation on a raw format image of dimension $256 \times 256$ it would take 65536 additions. If on the other hand,

the a Fourier domain format were used (in which a single stored value specifies the average intensity of the data) the same operation may be performed with a single addition. We refer to this optimisation by saying that the data format supports a specified set of operations. Of importance to image and video information are the operations “read from” and “write to” a storage device. These operations are made more efficient if a compressed format is chosen. On the other hand if the most important operation is display, a format which reflects the layout of the display memory or the primitives used to communicate with display hardware will be most efficient. To date, these two issues have existed independently and driven image and video formats down divergent paths. Recently however, dedicated hardware for decoding standard image and video formats have become available. In the foreseeable future display hardware will be capable of accepting image and video information in a range of different formats. This provides an opportunity for the development of data formats which support more of the operations performed by application software—leaving the mapping of this format to the display memory up to the system hardware.

In this paper we consider image and video format in the context of the operations commonly found in multimedia computing. In section 2 we propose a set of requirements based on our own research into multimedia authoring and presentation systems. In section 3 we show how many of these requirements are linked to an ability to access (image and video) data at a range of resolutions. The concept of scalability is presented as a solution to this in section section 4. The scalability of current standards as well as emerging formats is considered in section 4.1 from which some broader implications and concluding remarks (section 5) are drawn.

2 Requirements for Multimedia

2.1 Compression

Compression is considered a fundamental requirement of any image or video format used for multimedia due to the significant storage and bandwidth requirements associated with raw image and video data. As these issues are well understood and discussed in a number of papers (eg. Hilton, Jawerth & Sengupta (1994)) we will not elaborate on them here.

2.2 Supporting a Range of Architectures

An equally significant problem associated with the use of image and video data on computers is coping with the great diversity of computer architectures. Any multimedia computer may contain an arbitrary mix of graphics hardware, CPU and bus architectures and all of these factors will effect the ability of the machine to meet the real-time requirements of video presentation, and to a lesser extent, user interaction. Although there are currently standardisation efforts underway, it is likely that there will always be a mix of machines on the market and it is simply not viable for producers of multimedia applications software and on-line database systems to restrict their market to top-end machines.

This particular problem has a second, more complex manifestation in distributed multimedia systems. In such systems, the network resources not only vary according to their physical properties (architecture of the host machine, bandwidth of network cables etc.) but may also vary over time with fluctuations in network activity. In such situations it is important that client and server applications can dynamically adjust certain quality of service parameters in
order to meet any real-time constraints—such as maintaining a relatively constant frame rate for video data.

In fact both of these problems may be addressed by allowing a trade-off between data volume and certain quality parameters. In order to cope with variations in system capabilities during presentation, this trade-off must be available at presentation time rather than (as currently predominates) at he time the data is encoded. Even in the case where system parameters are constant with time (as can be assumed for many single user environments), this approach carries the added advantage that a single set of distribution media may be “pruned” at presentation time to suit the architecture of the host machine. This would represent a significant improvement in the portability of multimedia (and in particular image and video) information.

2.3 Interaction

Interaction with different media is becoming one of the most important features of multimedia systems. An implicit requirement of interaction is that the data be structured. In the MATILDA framework (Lowe & Ginue 1995) for example, data is considered to be structured at a number of conceptual levels. Each of these levels may be identified with a set of operations. At the lowest level is the raw data, the unit of structuring for which is the file and typical operations include read, write and display. At higher levels, lexical, syntactic and semantic structuring introduces progressively more knowledge about the nature of the data and, in so doing, supports more complex operations and forms of interaction.

Currently the provision of this structuring information is difficult. Invariably this information is entered entirely manually as with the “image maps” used in the mark-up of images for HTML documents. This, in turn represents a significant cost in the production of visual information systems and severely restricts the level of interactivity that can be practically provided by applications—especially for video information which may require separate mark-up for each frame.

Possibly the most attractive solution to this problem is through application of image analysis techniques. If the computer is able to refine a user’s rough input based on an analysis of the image data, the manual tracing of region boundaries is made considerably more efficient. The natural extension of such a system to video would see the machine able to both assist with the initial definition of regions, and then be able to track these regions through successive frames. A number of systems which make use of such techniques have been described recently (Ueda, Miyatake & Yoshizawa 1991, Flickner, Sawhney, Niblack, Ashley, Huang, Dom, Gorkani, Hafner, Lee, Petkovic, Steele & Yanker n.d.). Image analysis can also play an important role in the extraction of syntactic structure from image data. In the QBIC system (Flickner et al. n.d.) for example, successive frames of a video sequence are analyzed for the purpose of region extraction based on motion information. After this segmentation, each region is assigned a depth based on its interaction with other regions over time. The resulting structure can support queries which include simple 3D predicates such as “find an image of A in front of B”. For complex scenes with many regions, the ability to automate this process represents a significant saving over manual processing.

A problem which will need to be addressed if analysis is to find more widespread use in multimedia however, is that most analysis techniques are computationally expensive. Where analysis is combined with interaction, response time may become a problem. We can however view this problem as a generalisation of the “multiple architecture” problem of section 2.2. In
particular, where the data volume has been reduced by a trade in perceptual quality, analysis may yield a useful but more approximate result. Due to the reduced data volume, processing time will be reduced. Given appropriately designed algorithms it is conceivably possible to support the interactive use of analysis by this means.

A second approach to this problem, which should be examined in parallel with the above considerations is the selection of a data format which makes explicit information which is more directly useful in analysis (Lowe 1993, Burt 1984). Using the terminology of section 1 we note that such formats support certain analysis operations.

2.4 Presentation Requirements

There are two principle concerns of presentation which we identify. The first is the need to have fine control over the size (and rate for video) of presentation. This is required for two reasons. It is required to make information “fit” (where fit is used in a general sense) into mixed media presentations—examples of which include spatial placement within text and synchronisation with sound. But it is also required because the information contained in image and video data occurs over a range of scales. This can be clearly seen in the case of video where we might consider the content of individual frames, the short collection of frames bounded by camera changes, scenes etc. Correspondingly, an application for editing video will use a differently sized presentation of the data depending on whether whole scenes are being spliced or single frames are being “touched-up”. Similarly, applications for image viewing and editing will usually provide small displays for browsing collections of images and may also support magnified presentation of regions of interest from a “normal” image.

The second issue is ensuring that the required presentation is achieved consistently across a range of output devices. A problem here is that the display resolution of different output devices may vary widely. For example, screen resolution may vary in the range of 70–120ppi (pixel per inch) while hard-copy devices such as dye sublimation printers and slide recorders usually have resolution in excess of 300ppi. If presentation size is determined in terms of pixels (as it typically is), display size may vary dramatically between devices and machines. This problem is exacerbated by the fact that the resizing of data is (currently) handled in an ad-hoc fashion according to the particular application being used and the particular class of device it is communicating with.

Thus the ability to finely control the size and resolution of image and video data is the key requirement for presentation. To make this a more precise requirement (as resizing of image and video data is obviously possible and performed all the time) we additionally require that this process incorporate an efficiency trade-off. That is, there should be a proportional relationship between the resolution at which data is accessed and the processing (bandwidth etc.) required to perform the access. Only in this way can the full range of presentation modes (icon, normal, magnified, fast forward and rewind etc.) be readily supported on a single data set.

A third issue which will not pursue in any depth is that of providing perceptually correct colours on different output devices. This particular problem is well understood (Trussell 1993) and colour matching software, commercially available.

3 Requirements for a Data Format

We can extract three requirements from the above discussion. These are:
Figure 1: Trading image resolution for data size. The full resolution image (a) contains 65536 pixels; the same image at half resolution (b) contains 16384 pixels; and quarter resolution, 4096 pixels. The effect on perceived quality is notably less significant.

- Compression,
- The ability to perform at presentation time a perceptually smooth trade-off between presentation quality and data volume,
- The ability to perform a range of analysis tasks
- The ability to trade access resolution against the processing requirements of presentation

We can reduce this list further by noting that resolution is ideally suited to quality trade-offs as it has the power to dramatically effect the quantity of data a machine needs to handle whilst providing a smooth degradation in perceived quality. For example, halving the resolution of an image will quarter the volume of data (an inverse square relationship). The saving is even more dramatic for video as the relationship is inverse cubed. The effect on the perceived quality of the data (with change in resolution) would appear to be less significant than this (as demonstrated in figure 1).

Postponing a consideration of compression (about which much is already known) we observe that the problem of efficiently providing access to data at a range of resolutions is central to the requirements of multimedia. Specifically, a solution to this requirement provides a means by which processing and bandwidth demands may be traded (smoothly) against perceptual quality factors. It also provides a means of controlling the presentation of the data across a wide range of output (and input) devices by supporting access to the data at a resolution appropriate to any device. Due to the close relationship between resolution and size, the size of presentation may also be controlled by this mechanism.

A similar rational may be applied to the use of computer analysis of visual data. As was noted in section 2.3, most analysis techniques are computationally expensive. However, good approximate results may be achieved by processing data at lower resolution. In addition, both the efficiency and reliability of certain types of analysis may be improved by use a so called “multiresolution” techniques (Rozenfeld 1984) in which an initial low resolution analysis is progressively refined.

In summary we note that compression and multiresolution access are key requirements for image and video formats used in multimedia. One way of providing efficient multiresolution...
access which has received attention in recent image coding literature, is through the use of a 
scalable code.

4 Scalability

We consider scalability of an image or video code to mean that a file (or bit-stream) may be parsed and decoded in part or in full to produce versions of the data at differing resolutions. This differs slightly from the definition proposed by Bove, Jr. (1993) which trades “quality” of the decoded image as opposed to resolution. Our definition is however both less ambiguous and more consistent with the published implementations of scalable codes (Taubman & Zakhor 1994, Bove, Jr. 1993). Thus scalable codes not only provide a mechanism for the decoupling of stored resolution and the resolution at which the data is accessed by an application, but they achieve it through a simple and efficient means.

The smallest element in most scalable (image and video) codes is a significantly reduced version of the original data. This provides a floor to the resolution at which data may be accessed (without the express need for resizing by an application). The remainder of the file is composed of “detail elements”. The resolution of the decoded image may be increased by the inclusion of more of these detail elements up to the full resolution of the data (see figure 2).

4.1 Achieving Scalability

Most scalable coders for image and video data are based on the subband coding paradigm (Woods & O’Neil 1986). The mathematics of this paradigm (which is closely related to the wavelet transform (Daubechies 1992)) is particularly elegant and has received enormous amounts of attention in recent mathematical and signal processing literature. Recently, wavelet coding techniques were also introduced to the multimedia community by (Hilton et al. 1994) and currently there are attempts to produce a wavelet based image coding standard (Zandi, Allen, Schwartz & Boliek 1995).

In Taubman & Zakhor (1994), a scalable (multi-rate) video codec is described which is based on the wavelet transform. A software implementation of the codec has also been released (Taubman 1995). This supports a range of playback rates and resolutions (an example set of rates drawn from Taubman & Zakhor (1994) is given in table 1) and provides an
<table>
<thead>
<tr>
<th>Frame format</th>
<th>Available frame rates (fps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>352 x 240 colour</td>
<td>30, 15, 7.5, 3.75</td>
</tr>
<tr>
<td>352 x 240 monochrome</td>
<td>30, 15, 7.5, 3.75</td>
</tr>
<tr>
<td>176 x 120 colour</td>
<td>30, 15, 7.5, 3.75</td>
</tr>
<tr>
<td>176 x 120 monochrome</td>
<td>30, 15, 7.5, 3.75</td>
</tr>
<tr>
<td>88 x 60 colour</td>
<td>30, 15, 7.5, 3.75</td>
</tr>
<tr>
<td>88 x 60 monochrome</td>
<td>30, 15, 7.5, 3.75</td>
</tr>
<tr>
<td>44 x 30 colour</td>
<td>15, 7.5, 3.75</td>
</tr>
<tr>
<td>44 x 30 monochrome</td>
<td>30, 15, 7.5, 3.75</td>
</tr>
<tr>
<td>22 x 15 monochrome</td>
<td>15, 7.5, 3.75</td>
</tr>
</tbody>
</table>

Table 1: Playback rates supported on the scalable codec of for a data source of size 352 x 240 and 30fps frame rate. Note that all rates are related by a scale factor of 2.

excellent example of work in progress in this field. The performance of this codec in terms of compression ratio for a specified quality level was found to be equivalent to or in excess of MPEG-1.

The significant finding from this and other related works from our perspective is that it is possible to achieve compression and multiresolution access within a single image (video) format. We are therefore able to provide supports for the key requirements of multimedia within the framework of the data file format (or, more specifically the data representation).

Providing scalability within the MPEG framework has also been considered in the literature. While there are several approaches to this, the (representative) conclusion drawn by Anastassiou (1994) is that not more than three hierarchical multiresolution levels are practical for a single source and that the predominant application of MPEG be single resolution. In line with this, the current American HDTV standard does not provide for a hierarchical system.

4.2 Other Advantages of Wavelet based Scalable Codes

In addition to the scalability property, there is yet another reason why wavelet based (image and video) codes are well suited to use in multimedia applications. As alluded to in section 2.3, the provision of complex interaction often requires computer analysis which can be very expensive when performed on raw data. It turns out that a number of frequently used image operations have relatively simple implementations in the wavelet domain (Dorrell & Lowe 1995). This means that it is possible to perform image manipulations without having to fully decode the compressed data stream—thus providing significant computational saving. In addition, the nice mathematical properties of the wavelet transform mean that the encoded data may be directly useful in a range of image analysis tasks (Dorrell 1995).

4.3 Limitations

It is conceivable to suggest that the most efficient operation supported on a scalable image (or video) format is “resize”. As discussed in sections 2.2–2.4 this property has potential to considerably improve the efficiency of a broad class of interactions required for multimedia. So why are scalable image codes not in more widespread use?
There are several reasons why scalable image (and video) codes have not received the attention that single level codes (predominant in the JPEG and MPEG standards) have. One reason is their lesser maturity but possibly the most significant reason is that there is a price to pay for scalability in both coder complexity and compression. This has been influential as much of the recent development of (image and video) codes has been motivated by the implementation of High Definition Television (HDTV) services. For such services, compression and low decoder cost are the key requirements as display sizes are predetermined and fixed. For these services, multiresolution access is only needed in a very limited form—in order to provide backward compatibility with existing TV receivers. This is a very different environment to that of multimedia computing where, while complexity and compression are still important, efficient access across a finely graded and wide ranging set of resolutions is of utmost importance.

In addition to these issues, a second problem exists with the majority of scalable codecs which have been proposed in the literature. That is they support decoding only at a limited set of resolutions and specifically dyadic multiples of the "minimum" resolution (i.e. 1, 2, 4, 8...times minimum resolution). Providing a finer graduation remains an open area for research (Dorrell 1995).

5 Conclusions

For many years now, multimedia researchers have taken an open attitude towards formats for image and video information. In many ways this is appropriate as an application must be able to define operations on the information at the abstract level. But, because of the processing requirements involved in handling images and video data, its inclusion in multimedia applications is significantly influenced by the data format. With current formats being driven by the needs of compression they rarely support the type of abstractions, the level of interactivity or the varying presentation modes required for a complete integration into multimedia systems.

We argue that while current standards may have been appropriate for early multimedia systems, in which capture and fixed mode presentation predominated, they are inadequate for the current generation of interactive multimedia applications. The need for greater scalability in particular suggests that some rethinking of the current direction for image and video compression standards used in multimedia. For this reason, current efforts to develop a wavelet based compression standard are promising and should be embraced with enthusiasm be the multimedia community. But it must not stop there. The development of image and video formats for use in multimedia must take into account the scope of operations which are required to be performed on the data. The need to be able to perform analysis on the format, as well as have fine control over its presentation on a wide range of output devices must receive a similar level of consideration as compression. Perhaps only when this happens will there start to be truly integrated image and video information (until then we must continue to deliver the illusion and depend on tomorrow's faster architecture).

References


Taubman, D. S. (1995), Fully scalable, low-latency video codec…, public software release available by ftp from robotics.eecs.berkeley.edu in / pub / multimedia / scalable2.tar.Z.


9