MPEGTool: an X window-based MPEG encoder and statistics tool*

Toshiyuki Urabe, Hassan Afzal, Grace Ho, Pramod Pancha, and Magda El Zarki

Department of Electrical Engineering, University of Pennsylvania, Philadelphia, PA 19104, USA

Abstract. In this paper, we describe MPEGTool, an X window-based tool that can be used to generate a motion picture expert group (MPEG) encoded bit stream for video sequences and to study the statistical properties of the encoded data. The tool was designed to study the characteristics of variable bit rate video sources for transmission over asynchronous transfer mode (ATM) based broadband integrated services digital network (B-ISDN). The tool, which has a window-based graphic user interface, allows a user to specify several of the MPEG parameters such as the intraframe-to-interframe ratio and the quantizer scale. The tool also includes a statistical package that allows the user to plot graphs of various statistics including bit distributions, ATM cell distributions, time series, autocorrelation functions and cell interarrival times.

Key words: MPEG – Video encoding – Compression – ATM

1 Introduction

The demand for increased bandwidth to support new services that integrate diverse media such as video, voice, graphics and text has led to the introduction of broadband integrated services digital network (B-ISDN). Asynchronous transfer mode (ATM), a packet-switched approach based on fixed cell sizes, has been proposed as the switching and multiplexing scheme that can provide a unified transport structure for services in B-ISDN. Among the variety of services that will be supported by B-ISDN, video will be an important component because of its large bandwidth utilization and its stringent quality of service requirements.

ATM-based networks are especially well suited for variable bit rate (VBR) video transport because of their ability to allocate bandwidth on demand to these services. VBR video is potentially superior to constant bit rate (CBR) video since it can provide constant image quality for all scenes, as well as efficient bandwidth utilization. However, many aspects of the nature of the relation between VBR video services and network performance are still open issues [5].

One potential problem in ATM networks, caused by the "bursty" nature of traffic and statistical multiplexing, is cell loss. When several sources transmit at their peak rates simultaneously, the buffers available at some switches may be inadequate, causing overflow. The congestion at these switch buffers and the subsequent dropping of cells due to overflow will most likely be the major component of cell loss in BISDN. For VBR video sources this can lead to severe degradation in service quality, especially if cells are discarded indiscriminately at the switch buffers. Priority mechanisms are important in these situations as they can be used to flag cells that are essential for some minimum quality of service to be maintained.

The traffic characteristics of a video source are determined both by the properties of the video sequence, i.e., dimensions, motion content, etc., and the source coding technique. With the widespread acceptance of the MPEG coding standard, it is important that we understand the basic characteristics of MPEG-encoded video data and its behavior in ATM-based BISDN. Since hardware-based MPEG coders are not available right now (only a few vendors have started to ship sample versions of the MPEG chip set), the only viable method for performing MPEG coding is in software. The versatility and user friendly nature of the MPEGTool can aid the task of studying issues in video transmission over BISDN.

In this paper we describe the design of the MPEGTool that we are currently using for our studies of MPEG video. It consists of an MPEG encoder that encodes digital video data and a statistics package to study the characteristics of the encoded bit stream. Our MPEGTool is based on the current MPEG standard. This will be upgraded to MPEG II once it has been released. The tool consists of the following:

1. An X window-based graphic user interface
2. The MPEG encoder includes:
   a. Intraframe (I), Predictive (P) and Interpolative (B) frame coding
   b. Layering scheme that separates the encoded bit stream into a high-priority (HP) and a low-priority (LP) stream
3. A GNUPLLOT-based graphical statistics package

The outline of this paper is as follows: in Sect. 2 we present a brief overview of the MPEG coding algorithm and in Sect. 3, we describe the implementation of the MPEGTool. Section 4

* This work was supported in part by grant NCR 90-16165 from the National Science Foundation

Correspondence to: M. El. Zarki
presents some performance results for the coder and in Sect. 5, we display some of the statistical results generated by the tool.

2 MPEG coding

The MPEG coding algorithm was developed primarily for storage of compressed video on digital storage media [1]. Provisions were therefore made in the algorithm to enable random access, fast forward/reverse searches and other features when decoding from any digital storage media. However, the coding standard is suitable for a much wider range of video applications. Recent applications of MPEG-like coding algorithms have appeared for a variety of video services from multimedia workstations to high-definition television.

The MPEG coding scheme utilizes one of three coding modes for a frame in a video sequence; intraframe (I), predictive (P) or interpolative (B). Within a frame it is also possible to encode macroblocks (16 x 16 pixel blocks made up of 4 8 x 8 luminance pixel blocks and 2 8 x 8 chrominance pixel blocks) in one of several modes. A horizontal strip of macroblocks that makes up a row in the frame is called a slice. In the MPEG coding algorithm the slice is an important entity since it is the smallest unit that can be reconstructed independently at the receiver.

The macroblock coding modes that can be used are dependent on the type of coding mode (I, B, or P) used for the current frame. In I frames, all macroblocks are coded in intraframe mode using 8 x 8 two-dimensional discrete cosine transform (DCT) and are then quantized and variable length coded. In P frames, macroblocks can be coded with or without motion compensation. If motion-compensated mode is utilized, a motion vector is obtained by minimizing the absolute block difference between the current macroblock and a macroblock within the search window in the previous frame. The prediction error (the difference between the original macroblock and the motion compensated macroblock) is then coded using the DCT and quantized. Both the motion vector and the quantized DCT coefficients are then variable length coded. If the prediction error is larger than a given threshold, the macroblock is coded in the intraframe mode instead. For B frames, the coding procedure is similar to that for P frames. The major difference for B frames is that both forward and backward motion vectors are allowed, unlike P frames where only forward-motion vectors are utilized. The use of backward motion vectors allows for greater compression ratios, but has some disadvantages associated with it: it requires larger buffers at the source and receiver and involves longer processing times for each frame. It may therefore not be suitable for applications, such as interactive real-time video transmission.

We have enhanced the operation of the coder by adding a layering scheme, which separates the bits generated by the encoder into HP or LP bit streams. A parameter β specifies the number of AC coefficients (frequency components) in the HP stream (β = 64 puts all the coefficients into HP, i.e., the resultant bit stream is the same as the standard MPEG bit stream). More details on the layered coder can be obtained in [2]. In Fig. 1, we show a block diagram of the MPEG coder, which includes the layering mechanism.

Further details of the MPEG coding algorithm can be obtained in [1] and details on the application of the MPEG algorithm to variable bit rate video can be found in [3].

3 MPEGTool

Our tool consists of two components, an MPEG encoder and a statistics package. Since this tool has an X window-based graphical user interface, it is user friendly and easy to operate.

Figure 2 shows the data flow diagram of our tool. The encoder reads raw digital video data from a tape device, performs the encoding and generates an encoded bit stream. Since the encoding process is performed in software and not hardware, it cannot be done in real time. For this reason, all statistics are processed and displayed once encoding is complete.

The tool is versatile since it allows the user to specify several parameters for the encoding process. The first set of parameters deals with the intraframe-to-interframe coding ratio. A sequence of I, P and B frames is defined by the two parameters, M and N. N specifies the I frame interval whereas M specifies the I or P frame interval. N must be an integer multiple of M and when N is 1, the sequence contains only I