Battery Power Sensitive Video Processing in Wireless Networks

Prathima Agrawal[1] Jyh-Cheng Chen[2] Shalinee Kishore[3]
Parameswaran Ramanathan[4] Krishna Sivalingam[5]

[1]AT&T Labs, Whippany, NJ 07981
[2]Dept. of Electrical & Computer Engg., State University of New York, Buffalo, NY 14260
[3]WINLAB, Rutgers University, Piscataway, NJ 08855
[4]Dept. of Electrical and Computer Engineering, University of Wisconsin, Madison, WI 53706
[5]School of Electrical Engg. & Computer Science, Washington State University, Pullman, WA 99164

Abstract

Mobile computers typically have limited energy for computing and communications due to short battery lifetimes. Encoding, decoding, and transmission of video information require significant computing and communication resources. Low power encoding and decoding schemes have been researched extensively. In this paper, we focus on processing encoded video for transmission under low battery power conditions. Such processing, while conserving battery power, attempts to reduce deterioration of video quality.

Keywords— **Multimedia over wireless, video processing, video transmission, portable devices, low-power operation.**

I. Introduction

To meet the growing demand for anytime, anywhere, anymedia communications, next generation portable devices must have computing and communications support for multimedia information exchange. Video processing and transmission is a key component of multimedia information exchange. Hence, it is expected to be an integral part of next generation portable devices. To support mobility, portable devices communicate with the network over a wireless link. They also often operate on battery power. Since batteries have limited energy, conserving power is crucial in portable devices.

Most prior work on power conservation in video processing has focused on low-power circuit design. That is, the emphasis is on designing low power integrated circuits for encoding and decoding of video streams [1], [2]. Some of the suggested techniques include variable clock speed processors [3], reducing supply voltage [1], and use of flash memories [4]. Although these techniques prolong the lifetime of batteries, a portable device is still likely to encounter situations where its battery power levels are low. In this paper, we focus on techniques for reducing energy consumption in video processing and transmission under low battery power conditions. The techniques proposed in this paper can be incorporated along with low power circuit design techniques to further prolong the time between battery recharges and, thereby enhance mobility.

The basic idea of the proposed techniques is to decrease the number of bits transmitted over the wireless link to relay a given video stream. The challenge is to accomplish this goal while preserving or minimally degrading the video quality. Decreasing the number of transmitted bits reduces the energy consumption because considerable energy is expended in transmitting the bits over a wireless link. In fact, several studies have shown that transmission accounts for more than a third of the energy consumption in video processing and exchange in a portable device [5].

Prior work on decreasing the number of bits in video streams have mainly focused on different compression techniques (see Section II for a review of these techniques). Although techniques with higher compression ratio can be used as part of our approach, a choice of compression techniques is often not available to an application in a portable device. For instance, if the compression algorithm is implemented in hardware, then the flexibility available to an application is usually limited. Also, the portable may be transmitting a pre-recorded video stream under low battery conditions, in which case, the compression has already been performed. Therefore, the proposed approach also relies on discarding selected packets from compressed video streams in order to reduce the effective number of transmitted bits.

The rest of this paper is organized as follows. Section II review video processing techniques. Section III discusses the proposed approach. The paper concludes with Section IV.

II. Video Processing Schemes

A video stream is a sequence of image frames sent a specific rate. For example, the rate for a NTSC-compatible stream is 30 frames per second whereas that for video telephony is 10 frames per second [6]. Generally, there are two encoding types for compression of video information: *intraframe* and *interframe* coding [7], [8]. Intraframe coding exploits the spatial redundancy within a frame, while interframe coding exploits temporal redundancy between frames. Figure 1 depicts both intraframe and interframe encoding schemes in MPEG-1 and MPEG-2 standards. In the spatial domain, each video frame is divided into blocks with 8×8 pixels. Discrete cosine transform (DCT) is applied to these blocks, and the DCT coefficients are quantized to a user-defined level. Four blocks are grouped to create a **macroblock** (MB). A number of adjacent MBs are grouped to create a **slice**. A slice is the basic transmission and resynchronization unit for the video stream.

In the temporal domain, there are three types of frames in MPEG-1 and MPEG-2: intra (I), predictive (P), and bidirectional (B) frames. The encoding for I-frame is based on spatial redundancy and is described above. The purpose of P-frame is to reduce temporal redundancy by motion estimation. This is accomplished by searching for a best-match MB from the previous I-frame, for every MB in the current frame. The *motion compensated predictive error*, the difference between this MB and the best-match MB, is then stored. For a B frame, the best-match MB from both the previous and the next I or P frame is searched for. Both forward prediction and backward prediction can be used as the motion compensated prediction. Since a B frame needs the information from the previous and next the I or P frames, the processing order is different from the natural sequence of the video frame. Figure 1 shows the processing order for encoding and decoding of the frames, and the displaying order at the receiver. Video sequences are sent as a group of pictures (GOP) which contains several frames. The figure shows how the set of seven frames constitute a GOP.

A number of other algorithms for video encoding have also been developed. The JPEG standard [9] defined for encoding still images can be used for video by treating each frame as a still image. The coding scheme for each frame in JPEG is similar to intraframe coding of MPEG. The resulting video coding is often called Motion JPEG (MJPEG). The H.261 [10] standard is developed for videophone and video conferencing. It is similar to MPEG-1 but only has I and P frames. Both MPEG-1 and MPEG-2 define I, P, and B frames, and the coding algorithms are similar. MPEG-2, however, has some enhanced features, such as interlaced video manipulation, scalability, compatibility, error resilience, etc. Like MPEG-2, the H.263 standard is a hybrid of interframe prediction to reduce the temporal redundancy and intraframe coding. However, additional frame modes are defined for coding. The frame types are: intra (I) mode, predictive with one motion vector (P) mode, predictive with four motion vectors (P4) mode, predictive-bidirectional (PB) mode, and uncoded (U) mode. The I- and P-modes are similar to those in H.261 and MPEG-2.

Video encoding can done in either software or hardware. Some freely available software for video encoding can be found in [11], [12]. Since the coding of video generally requires intensive computation, custom hardware is also available. VLSI chips have been designed specially for DCT, motion estimation, or entropy coding which can be used as the auxiliary for video codecs. An overview of the hardware implementation and commercially available products can be found in [13].

III. Proposed Approach

The key components in a portable unit responsible for video processing and transmission are the *Video encoder* (VE), the *Video decoder* (VD), and the *Wireless Network Interface Card* (WNIC). The VE converts uncompressed video bit-stream into a compressed video stream and sends it to the WNIC for transmission. The VD receives compressed video stream from WNIC and converts it into an uncompressed video stream for display. As stated earlier, in this paper, we focus on reducing the energy consumed when a video stream is sent by the portable device under low battery power conditions. Therefore, the components of interest are the VE and the WNIC.

The proposed approach works as follows. The portable device runs a daemon in the background which periodically monitors the battery power level. If the power level is judged to be *almost full*, then the daemon does not impose any energy conservation mechanism on the video streaming applications. When the battery power level drops, the daemon requests a reduction in the number of bits transmitted over the wireless link to reduce energy consumption. The amount of reduction requested depends on the battery power level. For example, the battery power level can be considered to be in one of the following four distinct levels: *almost full*, *half-full*, *low-power*, and *nearly empty*. For each one of these levels, the daemon can associate a reduction factor r by which the application must reduce its transmitted bit rate. More specifically, if the application transmits at A bits/second when the battery level is almost full, then the application must transmit at a rate $(1 - r)A$ bits/second when the requested reduction factor is r.

The value of r corresponding to a battery power level can be chosen by the daemon based on its knowledge of the energy consumption profile and the traffic characteristics of the video streaming application. For example, suppose the WNIC in the portable device is a 915 MHz WaveLAN card. Then, studies have shown that energy consumption for transmitting a packet is directly proportional to its length and approximately 0.367 μwatt-second energy is required to transmit a bit in this WNIC card [14]. This profile information along with current battery power level, estimated battery power prolong time and the desired average bit rate of the application can be used to determine the reduction factor. Specifically, due to the linear relation between the energy consumption and the transmitted bit rate, a 20% savings in energy consumption can be achieved by using a reduction factor of 0.2. In this paper, we refer to the reduced bit rate when the battery is at a given power level as the *target bit rate* of the application.

Note that, a reduction in the number of transmitted bits may result in some loss of video quality at the receiver. The objective is to keep the deterioration in the video quality as small as possible. The reduction in the number of bits can be achieved in one of two ways: (i) reduce the number of bits in the compressed video stream generated by the VE, and (ii) discard selected packets at the WNIC

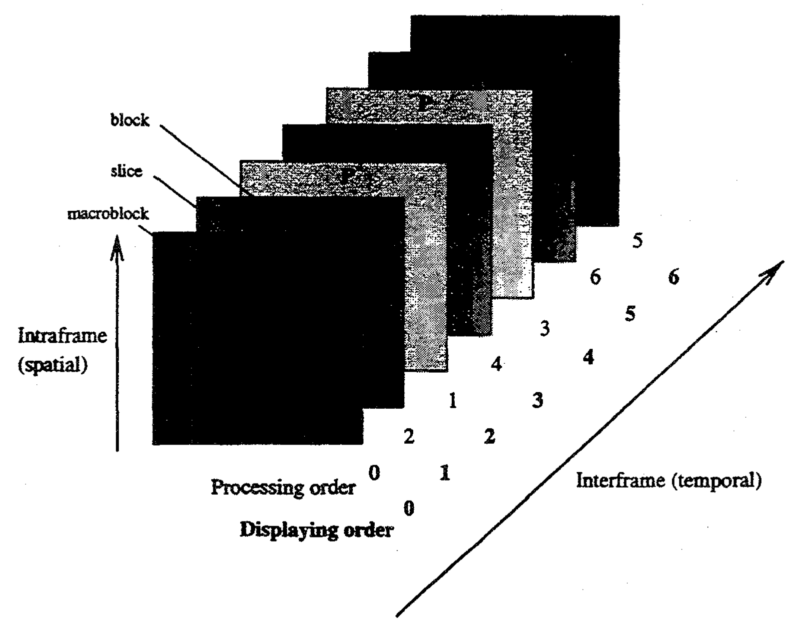

Fig. 1. Video coding in intraframe and interframe.

and reduce the number of bits transmitted over the wireless link. The first reduction approach is possible only if the following two conditions are satisfied: (i) the portable device must be encoding the video stream as opposed to transmitting a stored video, and (ii) the encoding parameters in the VE can be modified by the application. The second reduction approach is possible if WNIC is flexible and sensitive to battery power conditions. In the rest of this section, we assume that both reduction approaches are possible in the portable unit. If one of these approaches is not possible, then the techniques discussed in the relevant paragraphs are not applicable in that portable.

Reducing number of bits in the compressed video stream: The techniques for reducing the average bit rate of the compressed video stream depends on the compression scheme. For example, if only intraframe encoding is being used (e.g., MJPEG), then the average bit rate can be reduced either by reducing the frame rate or by increasing the step sizes in the corresponding quantization matrix. On the other hand, if the VE uses an interframe encoding technique such as in MPEG-1 and MPEG-2, then the average bit rate can also be reduced by changing the encoding pattern. For instance, in MPEG-2, VE can change the encoding pattern from IBBPBBPBBI to IBBPBBPBBPBBI and decrease the average number of bits required to encode the stream. However, this may involve more computation at the VE, which in turn will increase its energy consumption. This increase in energy compensation will most likely be compensated by the reduced power consumption in the WNIC due to the fewer number of bits that have to be transmitted over the wireless link.

Discarding packets at the WNIC: Selection of the packets to discard in the encoded frame sequence has significant impact on the quality of the video stream. The type of the packets discarded and their relative location affects the quality. For example, video encoding standards like MPEG-2 allow for layered encoding where packets are prioritized for discards. Specifically, discarding packets from higher numbered layered has less impact on the quality than those containing lower numbered layers. For such algorithms, WNIC can maintain statistics on the average bit rate of each layer in the stream and use that statistics to discard minimum number of layers to approximately meet the target bit rate. Of course, WNIC will discard all packets from a higher numbered layer before discarding packets from a lower numbered layer.

If layered encoding is not being used in the video stream but interframe encoding is being used, then the proposed strategy for discarding packets is shown in Figure 2. For ease of presentation, Figure 2 assumes that the video stream contains I, P, and B frames. The strategy can be easily modified to deal with streams which contain other types of frames (e.g., PB, P4, U). The basic idea of this strategy is to first discard as many B frames as necessary to achieve the target bit rate. If discarding B frames is not sufficient to meet the target bit rate, the P frames are discarded. If discarding all B and P frames does not achieve the target bit rate, then I frames are discarded. In each case, the discarded frames are equally spaced over a specified N frame interval.

The rationale for discarding frames in this order is as follows. If a B frame is discarded, then only the information contained in that frame is lost. The decoder at the receiver can use interpolation techniques to estimate

```
DISCARD_FRAME (N, r)   /* Reduce average bit rate by factor r over N frames */
    /*
    f_B:      Fraction of B frames in the video stream;
    f_P:      Fraction of P frames in the video stream;
    f_I:      Fraction of I frames in the video stream;
    B_ave:    Average number of bits in a B frame;
    P_ave:    Average number of bits in a P frame;
    I_ave:    Average number of bits in a I frame;
    A:        Average number of bits in the video frame;
    */

    A = f_B · B_ave + f_P · P_ave + f_I · I_ave;
    k_B = r · A · N/B_ave;      /* Number of B frames to be discarded */
    If k_B ≤ f_B · N then
        Discard one out of every f_B · N/k_B B frames;
    Else {
        Discard all B frames;
        k_P = (r · A · N − f_B · N · B_ave) / P_ave ;     /* Number of P frames to be discarded */
        If k_P ≤ f_P · N then
            Discard one out of every f_P · N/k_P P frames;
        Else {
            Discard all P frames;
            k_I = (r · A · N − f_B · N · B_ave − f_P · N · P_ave) / I_ave;     /* Number of I frames to be discarded */
            Discard one out of every f_I · N/k_I I frames;
        }
    }
    End.
```

Fig. 2. Proposed strategy for discarding video frames.

the contents of that frame. For example, the decoder can once again display the previous frame in place of the discarded frame. On the other hand, if a P frame is discarded, then information contained in the associated B frames are also not recoverable along with the contents in P frame. Therefore, the decoder in the receiver will have to interpolate information in several successive frames. This usually causes more severe deterioration in video quality than the loss of an isolated B frame. Likewise, if an I frame is discarded, then the information in the associated P and B frames are not recoverable. In other words, the decoder must interpolate more frames in this case than when a P frame is discarded. Consequently, it is better to discard B frames when possible, followed by P frames, and then I frames. The rationale for equally distributing the discarded frames over an interval is as follows. It is usually much easier for the decoder to interpolate the contents of an isolated discarded frame than successive discarded frames.

Based on this rationale, the discarding strategy in Figure 2 can be explained as follows. Let f_B, f_P, and f_I respectively denote the fraction of B, P, and I frames in the video stream. Also let B_{ave}, P_{ave}, and I_{ave} denote the average number of bits in a B, P, and I frame in the encoded video stream. Initial values for f_B, f_P, f_I, B_{ave}, P_{ave}, and I_{ave} are provided to WNIC by the application. WNIC can periodically update these values based on the statistics it maintains for the video stream. If frames need to be discarded, the application specifies a parameter N representing the number the frames over which the reduction in average bit rate is to be performed. WNIC then computes, k_B, the number of average size B frames which must be discarded to achieve the target bit rate. If k_B is larger than the expected number of B frames in a window of N frames, then WNIC computes k_P, the additional number of P frames which must be discarded to achieve the target bit rate. Once again, if k_P is greater than the expected number of P frames in a window of N frames, then WNIC computes k_I, the number of I frames which also need to be discarded in additional to all the P and the B frames.

IV. Summary

Energy conservation is critical in portable devices. Video processing in portable devices is gaining considerable importance because of the growing demand for mobility and multimedia communication. In this paper, we discussed energy conservation techniques for video processing under low battery power conditions. These techniques conserve energy by reducing the number of transmitted bits in a video stream with minimal degradation in quality. Work on experimental evaluation of the techniques discussed in this paper is in progress.

REFERENCES

[1] A. Chandrakasan and R. W. Brodersen, *Low Power Digital CMOS Design*, Kluwer Academic Publishers, Norwell, MA, 1995.

[2] B. M. Gordon, E. Tsern, and T. H. Meng, "Design of a low power video decompression chip set for portable applications," *Journal of VLSI Signal Processing Systems*, vol. 13, pp. 125–142, 1996.

[3] K. Govil, E. Chan, and H. Wasserman, "Comparing algorithms for dynamic speed-setting of a low-power CPU," in *Proc. ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Berkeley, CA, Oct. 1995, pp. 13–25.

[4] B. Marsh, F. Douglis, and P. Krishnan, "Flash memory file caching for mobile computers," Tech. Rep. MITL-TR-59-93, Matsushita Info Tech Lab, Princeton, NJ, June 1993.

[5] Jyh-Cheng Chen, Krishna M. Sivalingam, Prathima Agrawal, and Shalinee Kishore, "A comparison of MAC protocols for wireless local networks based on battery power consumption," in *Proc. IEEE INFOCOM*, San Francisco, CA, Apr. 1998, To appear.

[6] D. W. Lin, C.-T Chen, and T. R. Hsing, "Video on phone lines: Technology and applications," *Proceedings of the IEEE*, vol. 83, no. 2, pp. 175–193, Feb. 1995.

[7] ISO/IEC JTC1 CD 11172, "Coding of moving pictures and associated audio for digital storage media up to 1.5 Mbits/s," 1992.

[8] ISO/IEC JTC1 CD 13818, "Generic coding of moving pictures and associated audio," 1994.

[9] ISO/IEC JTC1 CD 10918, "Digital compression and coding of continuous-tone still image - part I: Requirements and guidelines," 1993.

[10] ITU-T Rec. H.261, "Video codec for audiovisual services at p x 64 kbit/s," Mar. 1993.

[11] "Digital video coding at Telenor R&D," http://www.fou.telenor.no/brukere/DVC/.

[12] "MPEG pointers and resources," http://www.mpeg.org/.

[13] Vascudev Bhaskaran and Konstantinos Konstantinides, *Image and video compression standards: algorithms and architectures. Second edition*, Kluwer Academic Publishers, Norwell, MA, 1997.

[14] Mark Stemm and Randy H. Katz, "Measuring and reducing energy consumption of network interfaces in hand-held devices," *IEICE Transactions on Fundamentals of Electronics, Communications, and Computer Science*, Aug. 1997.