INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad-500 043

**Project Based Learning**

(Prototype / Design Building) External Evaluation Report

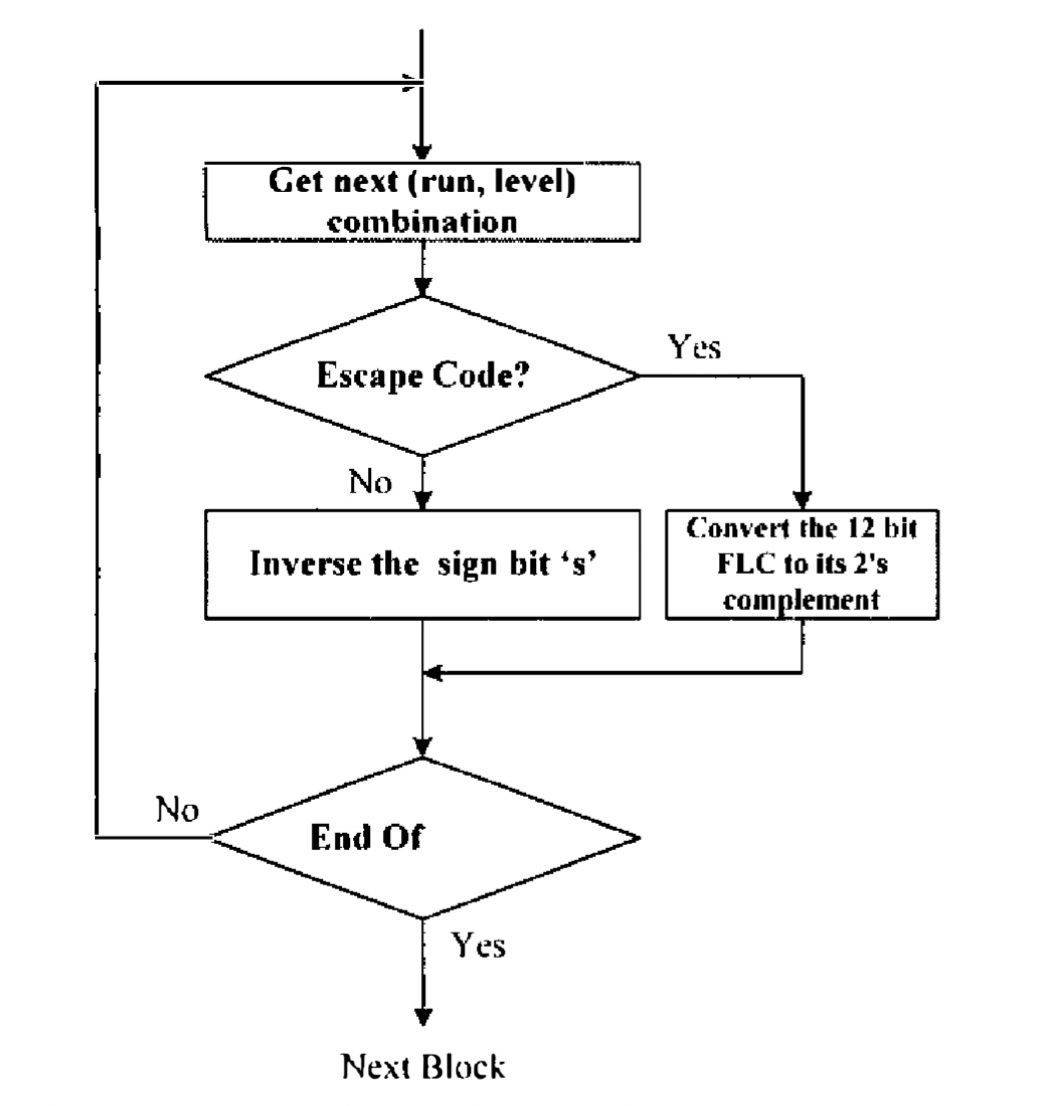
Title of your Idea : EFFICIENT ALGORITHM FOR REVERSE PLAYBACK IN MPEG VIDEO STREAMING

Thrust Area / Sector : Antenna designing

Branch : ECE-C

Year / Semester : 3rd Year / 5th semester

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S. No** | **Name of the Student** | **Roll Number** | **Mobile Number** | **Signature** |
| **1** | Chintah sudheer | 19951A04JO | +917569979663 |  |

* **Background of the Idea :**
* Reverse playback is the most common video cassette recording (VCR) function in many digital video players. However, the predictive processing techniques employed in MPEG severely complicate the reverse-play operation. We have proposed a compressed-domain algorithm for the MPEG video streaming system to provide efficient reverse playback . In the proposed video streaming server , it classifies macroblocks in the requested frame into two categories – a backward macroblock (BMB) and a forward macroblock (FMB). For BMB , a sign inversion technique has been proposed to reduce the number of macroblocks that need to be decoded by the decoder. However the required number of macroblocks for reconstructing FMB is still very high.
* The server identifies the related previous macroblocks of FMB and reduces the redundancie s of those related macroblocks to further reduce the computational complexity of the client decoder. Experimental results show that, as compared to the conventional and the previously proposed algorithms, the new streaming algorithm reduces the required network bandwidth and the decoding complexity significantly
* **Problem Statement (Min 100 words):**
* Our previous algorithm has already provided the way to obtain BMlh with much less decoding complexity and network bandwidth than the conventional system. In this a technique that reduces the redundancies of those related maroblocks for the reconstruction of FMBs has been proposed to further alleviate the computational burden of the client decoder in reverse-play operation. Experimental results show that, as compared to the conventional system, the new streaming system reduces the required network bandwidth and the decoding complexity significantly
* **Proposed Solution (Min 100 words):**
* We have improved our previously proposed reverse-play algorithm for an MPEG video streaming system. With the motion information, the video streaming server organizes the macrohlocks in the requested frame into two categories: backward macrohlocks (BMBs) and forward macroblocks (FMBs).
* The architecture of the proposed system is shown in Figure 3. In the forward-play operation, the switches SW,, SW,, SW, and SW, are connected to A,, A,, A, and AI respectively. On the other band, in contrast to the frame based scheme used in the conventional architecture, a macroblock-based scheme is proposed to use in the reverse play operation
* At the server motion vectors are extracted from the video stream and these motion vectors are used by a macroblock selector to identify the types of macroblocks
* .Two types of macroblocks are now defined. For illustration, assuming that a user requests a reverse-play command at frame n, the next frame to be display is frame n-I.
* We assume that ME represents the macroblock at the Ph row and rh column of frame n-I (the next frame to be displayed)
* **TECHNOLOGY CONCEPT FORMULATION:**
* We use the python language code for the reverse video play
* And we also use opencv for the above topic to run the program
* Opencv program
* **import cv2**
* **capture = cv2.VideoCapture("C:\\Users\\user\\Videos\\Captures\\1rr.mp4")**
* **if capture.isOpened() is False:**
* **print("Error opening video")**
* **frame\_idx = capture.get(cv2.CAP\_PROP\_FRAME\_COUNT) - 1**
* **print("Starting Frame: '{}'".format(frame\_idx))**
* **while capture.isOpened() and frame\_idx >= 0:**
* **capture.set(cv2.CAP\_PROP\_POS\_FRAMES, frame\_idx)**
* **ret, frame = capture.read()**
* **if ret is True:**
* **cv2.imshow('Frame in Reverse', frame)**
* **frame\_idx = frame\_idx - 1**
* **print("Next index: '{}'".format(frame\_idx))**
* **if cv2.waitKey(30) & 0xFF == ord('q'):**
* **break**
* **else:**
* **break**
* **capture.release()**
* **cv2.destroyAllWindows()**
* **Prototype of proposed system (UI screens / block diagrams / circuits / designs):**
* 
* **Detailed description of prototype / product / project (Min 1000 words):**
* The sign inversion 01' DCT cocrticients for BMB requires additional variable length decoding and re-encoding. To reduce the computational load of the server, the sign inversed OCT coefllcints can be computed in VLC·domain (Variahle Length Code). [n MI'EG video encoding, each nonzero DCT coeflicient is represented by the RUN-LEVEL symbol structure which has its corresponding variable length code. RUN reters 10 the number of zero coefficients before the next nonzero coefticiellt and LEVEL refers to thc amplitude or the nonzero coeftlcient. The trailing bit of each VLC is the 's' bit that indicates the sign of the nonzero coetricient.
* The coefficienl is positive; otherwise it is negative. for most comhinations, to perform sign inversion, the server just parses the MPEG video bitstream and inverts all's' bits of VLCs in BMB. However, some RUN-LEVEL combinations that are not frequently used are cod,d using a 6-bit "Escape" code followed by a 6-bit fixed kngth code (fLC) for RUN and a 12·bit FLC for LEVEL. In this case, the 12-bit fLC for LEVEL is converted into its 2's complement. The bit manipulation of VLCs in BMB is summarized in Figure 2. Since it is not necessary to perform VLC encoding, motion compensation, OCT, quantization, inverse OCT, inverse quantization and VLC decoding in the server, the loading of the server is reduced significantly.
* **CONCLUSION**
* We have improved our previously proposed reverse-play algorithm for an MPEG video streaming system. With the motion information, the video streaming server organizes the macrohlocks in the requested frame into two categories: backward macrohlocks (BMBs) and forward macroblocks (FMBs)
* Our previous algorithm has already provided the way to obtain BMlh with much less decoding complexity and network bandwidth than the conventional system. In this paper, a technique that reduces the redundancies of those related maroblocks for the reconstruction of FMBs has been proposed to further alleviate the computational burden of the client decoder in reverse-play operation. Experimental results show that, as compared to the conventional system, the new streaming system reduces the required network bandwidth and the decoding complexity significantly.

**REFERENCES**

* 1 L. Chiuriglione, "The development of an integrated audiovisual coding standard: MI'EG," Proceedings of [EEE. vol. 83 , pp. 151-157, Feb. 1995
* ISOIIEC 1 1172-2, "Information technology -- coding of moving pictures and associated audio for digital storage media
* ISO/IEC 13818-2, "Information technology -- generic coding of moving pictures and associated audio information: video," 1996.
* Y.L Chan and w.e. Siu, "An ellicient search strategy for block motion estimation using image features", IEEE Trans. on Image Processing, Vol. 10, No.8, pp. 1223-1238, Aug. 2001.
* M. S. Chen and D. D. Kandlur, "Downloading and stream conversion: supporting interactive playout of videos in a client station," in Proc. 2nd Inl. IEEE Conf. Multimedia Computing and Systems, pp. 73-80, 1995.
* S. 1. Wee, "Reversing motion vector fields," in Proc. IEEE International on Conference Image Processing 1998 (ICIP98), pp. 209-212, Oct. 1998
* C.l1. Fo, V.L. Chan and W.e. Siu, '"Macroblock-bascd reverse play algorithm for MPEG Video Streaming" in Proc. IEEE International Symposium on Circuits and Systems (ISeAS 2004), vol. 111, pp. 753-756, May 2)-26, 2004.

Dean-CLET Dean -TIIC