K.Rajalakshmi, K.Mahesh / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 3, Issue 3, May-Jun 2013, pp.1112-1115

Lossless Compression in MPEG4 Videos

K.Rajalakshmi, K.Mahesh
Research scholar(M.Phil),Dept. of CSE, Alagappa University, Karaikudi, Tamilnadu,India.

Abstract

This work considers object repetition, which means the occurrence of same object with similar motion is repeated in two or more frames. A single video is divided into several frames. All the objects are extracted from the stationary background and their features like motion and shape are preserved. After extraction, a two step compression is employed in which the algorithm checks for the continuous motion of all objects and then the video sequence are segmented. Now, the segmented video is compressed. This methodology maintains the video quality of the compressed video to be the same as source video.

Index Terms- Video, Compression.

I. INTRODUCTION

Nowadays, usage of video file increases due to the advent of youtube, facebook etc., This paper considers MPEG video as input. Video compression without quality loss is a challenge and this paper works on it.

In this work, an MPEG video is broken up into a hierarchy of layers to help with error handling, random search and editing, and synchronization, for example with an audio bitstream. From the top level, the first layer is known as the video sequence layer, and is any self-contained bitstream, for example a coded movie or advertisement.

The second layer down is the group of pictures, which is composed of 1 or more groups of intra (I) frames and/or non-intra (P and/or B) pictures that will be defined later. Of course the third layer down is the picture layer itself, and the next layer beneath it is called the slice layer.

Each slice is a contiguous sequence of raster ordered macroblocks, most often on a row basis in typical video applications, but not limited to this by the specification. Each slice consists of macroblocks, which are 16x16 arrays of luminance pixels, or picture data elements, with 2 8x8 arrays of associated chrominance pixels.

The macroblocks can be further divided into distinct 8x8 blocks, for further processing such as transform coding. Each of these layers has its own unique 32 bit start code defined in the syntax to consist of 23 zero bits followed by a one, then followed by 8 bits for the actual start code.

These start codes may have as many zero bits as desired preceding them. The MPEG encoder also has the option of using forward/backward interpolated prediction. These frames are commonly referred to as bi-directional interpolated prediction frames, or B frames for short. As an example of the usage of I, P, and B frames, consider a group of pictures that lasts for 6 frames, and is given as I,B,P,B,P,B,I,B,P,B,P,B**.** As in the previous I and P only example, I frames are coded spatially only and the P frames are forward predicted based on previous I and P frames. The B frames however, are coded based on a forward prediction from a previous I or P frame, as well as a backward prediction from a succeeding I or P frame. As such, the example sequence is processed by the encoder such that the first B frame is predicted from the first I frame and first P frame, the second B frame is predicted from the second and third P frames, and the third B frame is predicted from the third P frame and the first I frame of the next group of pictures. From this example, it can be seen that backward prediction requires that the future frames that are to be used for backward prediction be encoded and transmitted first, out of order.

This process is summarized in Fig 1.There is no defined limit to the number of consecutive B frames that may be used in a group of pictures, and of course the optimal number is application dependent. Most broadcast quality applications however, have tended to use 2 consecutive B frames (I,B,B,P,B,B,P) as the ideal trade-off between compression efficiency and video quality.

B-Frame Encoding

The main advantage of the usage of B frames is coding efficiency. In most cases, B frames will result in less bits being coded overall. Quality can also be improved in the case of moving objects that reveal hidden areas within a video sequence. Backward prediction in this case allows the encoder to make more intelligent decisions on how to encode the video within these areas.

Also, since B frames are not used to predict future frames, errors generated will not be propagated further within the sequence. One disadvantage is that the frame reconstruction memory buffers within the encoder and decoder must be doubled in size to accommodate the 2 anchor frames.

This is almost never an issue for the relatively expensive encoder, and in these days of inexpensive DRAM it has become much less of an issue for the decoder as well. Another disadvantage is that there will necessarily be a delay throughout the system as the frames are delivered out of order. Most one-way systems can tolerate these delays, as they are more objectionable in applications such as video conferencing systems.

II. Motion Estimation

The temporal prediction technique used in MPEG video is based on motion estimation. The basic premise of motion estimation is that in most cases, consecutive video frames will be similar except for changes induced by objects moving within the frames.

In the trivial case of zero motion between frames (and no other differences caused by noise, etc.), it is easy for the encoder to efficiently predict the current frame as a duplicate of the prediction frame. When this is done, the only information necessary to transmit to the decoder becomes the syntactic overhead necessary to reconstruct the picture from the original reference frame.

When there is motion in the images, the situation is not as simple. Fig 1 shows an example of a frame with 2 stick figures and a tree. The second half of this figure is an example of a possible next frame, where panning has resulted in the tree moving down and to the right, and the figures have moved farther to the right because of their own movement outside of the panning.

The problem for motion estimation to solve is how to adequately represent the changes, or differences, between these two video frames.

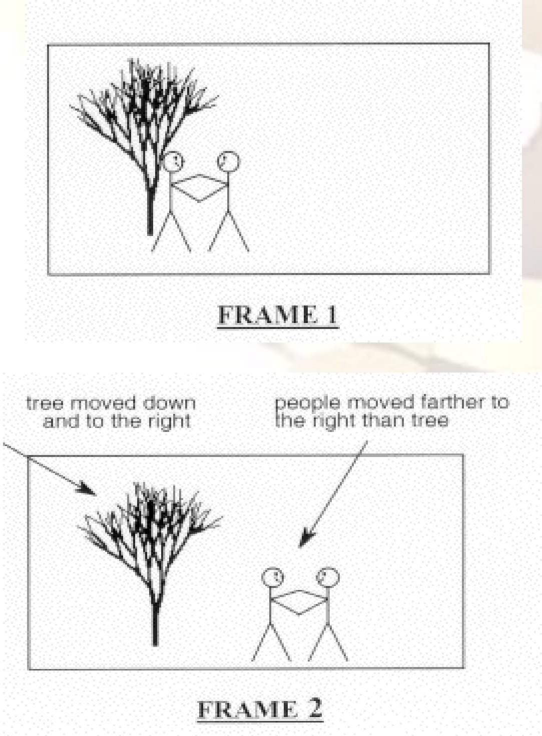

Fig 1: Video Sequence

The way that motion estimation goes about solving this problem is that a comprehensive 2-dimensional spatial search is performed for each luminance macroblock. Motion estimation is not applied directly to chrominance in MPEG video, as it is assumed that the color motion can be adequately represented with the same motion information as the luminance.

It should be noted at this point that MPEG does not define how this search should be performed. This is a detail that the system designer can choose to implement in one of many possible ways.

It is well known that a full, exhaustive search over a wide 2-dimensional area yields the best matching results in most cases, but this performance comes at an extreme computational cost to the encoder. As motion estimation usually is the most computationally expensive portion of the video encoder, some lower cost encoders might choose to limit the pixel search range, or use other techniques such as telescopic searches, usually at some cost to the video quality.

After a predicted frame is subtracted from its reference and the residual error frame is generated, this information is spatially coded as in I frames, by coding 8x8 blocks with the DCT, DCT coefficient quantization, run-length/amplitude coding, and bitstream buffering with rate control feedback.

This process is basically the same with some minor differences, the main ones being in the DCT coefficient quantization. The default quantization matrix for non-intra frames is a flat matrix with a constant value of 16 for each of the 64 locations. This is very different from that of the default intra quantization matrix which is tailored for more quantization in direct proportion to higher spatial frequency content.

As in the intra case, the encoder may choose to override this default, and utilize another matrix of choice during the encoding process, and download it via the encoded bitstream to the decoder on a picture basis. Also, the non-intra quantization step function contains a dead-zone around zero that is not present in the intra version.

This helps eliminate any lone DCT coefficient quantization values that might reduce the run-length amplitude efficiency. Finally, the motion vectors for the residual block information are calculated as differential values and are coded with a variable length code according to their statistical likelihood of occurrence.

III. RELATED WORK

In this section, we are presenting the research work of some prominent authors in the same field and explaining a short description of various techniques used for video compression.

G.Suresh, P.Epsiba, Dr.M.Rajaram, Dr.S.N.Sivanandam "A LOW COMPLEX SCALABLE SPATIAL ADJACENCY ACC-DCT BASED VIDEO COMPRESSION METHOD",2010 proposed a video compression approach which tends to hard exploit the temporal redundancy in the video frames to improve compression efficiency with less processing complexity.

It produces a high video compression ratio. Many experimental tests had been conducted to prove the method efficiency especially in high bit rate and with slow motion video. The proposed method seems to be well suitable for video surveillance applications and for embedded video compression systems. Tzong-Jer Chen, Keh-Shih Chuang "A Pseudo Lossless Image Compression Method",2010 present a lossless compression which modifies the noise component of the bit data to enhance the compression without affecting image quality. Data compression techniques substantially reduce the volume of the image data generated and thus increase the efficiency of the information flow. Method is information lossless and as a result, the compression ratio is smaller.

Qiang L iu, Robert J . Sclabassi , Mark L . Scheuer, and Mingui Sun "A Two-step Method For Compression of Medical Monitoring Video"2010 present a two-step method to compress medical monitoring video more efficiently. In the first step, a novel algorithm is utilized to detect the motion activities of the input video sequence.

Then, the video sequence is segmented into several rectangle image regions (video object planes), which contain motion activities restricted within these windows. In the second step, the generated video object planes are compressed. Our experimental results show that the two-step method improves the compression ratio significantly when compared with the existing algorithms while still retaining the essential video quality.

Raj Talluri, , Karen Oehler, Thomas Bannon, Jonathan D. Courtney, Arnab Das, and Judy Liao "A Robust, Scalable, Object-BasedVideo Compression Technique for Very Low Bit-Rate Coding" 1997 describes an object-based video coding scheme (OBVC) this technique achieves efficient compression by separating coherently moving objects from stationary background and compactly representing their shape, motion, and the content. In addition to providing improved coding efficiency at very low bit rates, the technique provides the ability to selectively encode, decode, and manipulate individual objects in a video stream. Applications of this object-based video coding technology include video conferencing, video telephony, desktop multimedia, and surveillance video.

Ian Gilmour, R. Justin Dávila "Lossless Video Compression for Archives: Motion JPEG2k and Other Options"2011 algorithm is clearly for end-user distribution through narrow bandwidths, and where no subsequent re-coding or re-puposing is required. The optimisation of image quality within individual frames allows true lossless data-reduction for applications such as archiving, where no loss of image quality is acceptable.

Yucel Altunbasak, A. Murat Tekalp, and Gozde Bozdagi "TWO-DIMENSIONAL OBJECT-BASED CODING USING A CONTENT-BASED MESH AND AFFINE MOTION PARAMETERIZATION"1995 present a complete system for 2-D object-based video compression with a method for 2-D content-based triangular mesh design, two connectivity preserving affine motion parameterization schemes, two methods for temporal mesh propagation, a polygon-based adaptive model failure detection/coding scheme, and bitrate control strategies.

Raj Talluri "A HYBRID OBJECT-BASED VIDEO COMPRESSION TECHNIQUE"1996 describes a hybrid object-based video coding scheme that achieves efficient compression by separating coherently moving objects from stationary background and compactly representing their shape, motion and the content.

In addition to providing improved coding efficiency at very low bit rates the technique provides the ability to selectively encode, decode and manipulate individual objects in a video stream Applications of this object- based video coding technology include video conferencing, video telephony, desktop multimedia, and surveillance video. Video conferencing, video telephony, desktop multimedia and surveillance video.

Zixiang Xiong "Video Compression Based on Distributed Source Coding Principles" 2009 This paper applies distributed source coding principles to three areas of video compression, leading to Witsenhausen-Wyner video coding, layered Wyner- Ziv video coding, and multiterminal video coding. Compared to H.264/AVC and H.26L-FGS video coding, the advantage of Witsenhausen-Wyner video coding and layered Wyner-Ziv video coding is error robustness when the compressed video has to be delivered over a noisy channel. The price for this added error robustness is a small loss of compression performance.

IV. PROPOSED WORK

Our work follows the below given algorithm, which successfully segments the video and compresses it without any quality loss.

```
Break video into layers;
Compute B frames;
Estimate motion;
if(motion=0)
 duplicate frame;
else
 subtract the predicted frame from previous frame;
end if;
obtain motion vectors;
end;
```

Fig:2 Algorithm

In this work, an MPEG video is broken up into a hierarchy of layers to help with error handling, random search and editing, and synchronization, for example with an audio bitstream. From the top level, the first layer is known as the video sequence layer, and is any self-contained bitstream, for example a coded movie or advertisement.

The second layer down is the group of pictures, which is composed of 1 or more groups of intra (I) frames and/or non-intra (P and/or B) pictures that will be defined later. Of course the third layer down is the picture layer itself, and the next layer beneath it is called the slice layer.

Each slice is a contiguous sequence of raster ordered macroblocks, most often on a row basis in typical video applications, but not limited to this by the specification. Each slice consists of macroblocks, which are 16x16 arrays of luminance pixels, or picture data elements, with 2 8x8 arrays of associated chrominance pixels.

The macroblocks can be further divided into distinct 8x8 blocks, for further processing such as transform coding. Each of these layers has its own unique 32 bit start code defined in the syntax to consist of 23 zero bits followed by a one, then followed by 8 bits for the actual start code.

These start codes may have as many zero bits as desired preceding them. The MPEG encoder also has the option of using forward/backward interpolated prediction. These frames are commonly referred to as bi-directional interpolated prediction frames, or B frames for short.

The main advantage of the usage of B frames is coding efficiency. In most cases, B frames will result in less bits being coded overall. Quality can also be improved in the case of moving objects that reveal hidden areas within a video sequence. Backward prediction in this case allows the encoder to make more intelligent decisions on how to encode the video within these areas. After all the processes mentioned in Introduction part, the motion vectors for the residual block information are calculated as differential values and are coded with a variable length code according to their statistical likelihood of occurrence.

V. CONCLUSION

Thus, this work segments the input video and then estimates the motion. After extraction, a two step compression is employed in which the algorithm checks for the continuous motion of all objects and then the video sequence are segmented. Now, the segmented video is compressed. This methodology maintains the video quality of the compressed video to be the same as source video.

REFERENCES

[1] Raj Talluri, Member, IEEE, Karen Oehler, Member, IEEE, Thomas Bannon, Jonathan D. Courtney, Member, IEEE, Arnab Das, and Judy Liao "A Robust, Scalable, Object-Based Video Compression Technique for Very Low Bit-Rate Coding", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL. 7, NO. 1, FEBRUARY 1997.

[2] G.Suresh, P.Epsiba, Dr.M.Rajaram, Dr.S.N.Sivanandam "A LOW COMPLEX SCALABLE SPATIAL ADJACENCY ACC-DCT BASED VIDEO COMPRESSION METHOD", 2010 Second International conference on Computing, Communication and Networking Technologies.

[3] Tarek Ouni, Walid Ayedi, Mohamed Abid "New low complexity DCT based video compression Method", 2009 IEEE.

[4] K.Uma, P.Geetha palanisamy, P.Geetha poornachandran "Comparison of Image Compression using GA, ACO and PSO techniques" , IEEE-International Conference on Recent Trends in Information Technology, ICRTIT 2011.

[5] Tzong-Jer Chen, Keh-Shih Chuang "A Pseudo Lossless Image Compression Method", 2010 3rd International Congress on Image and Signal Processing.

[6] Nasir D. Memon, Khalid Sayood "Asymmetric Lossless Image Compression", 2011IEEE