International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075, Volume-8 Issue-4S2 March, 2019

A Novel Technique for Buffer Based Advanced Video Coding

Shafee Vunnisa Sayyad, V.S.K. Reddy

Abstract--- Day by Day demand for video coding is enormously increasing. One of the major areas of the video communication application is video streaming i.e video on demand. Streaming meant for simultaneous transmission as well as playing. The frames that are stored in the buffer are playing meanwhile the rest of the frames are filling the buffer. For the continuous playing of the video with out any interruption the rate of filling the buffer must be greater than the rate of playing the frames. If not it leads to buffering. How fast the buffer is filling depends upon the available bandwidth. In this paper a new algorithm introduced which will work on adaptive video coding depends on the percentage of the buffer capacity .

Index Terms—PSNR , SP ,Buffer, ARPS, ES, NTSS, DS

I. INTRODUCTION

There are majorly two areas where the video communication applications can broadly be categorized. They are:

1. Conversational video
2. Streaming video

Category of conversational video involves video telephony and video conferencing which could be either wired or wireless. Streaming video is another major category of video communication applications. This video can be watched through the internet as the one which is on demand and is capable to download in our terminal. The Video on demand is an important application of a streaming video category where a video that is already recorded before is made available on a digital medium and can be accessed by any user through the internet and played on their systems. Simply, a streaming video means that simultaneous transmission of video stream and watching it by the user is possible where the user has no purpose of waiting for the whole video to be downloaded.

Once the user is in a position to receive from the encoder and play some of the frames, it could be watched meanwhile the remaining part would be transmitted from the encoder to the receiver. So there is no scope for delay. Therefore, video on demand, HDTV applications and video or image database services is also another application set of a streaming video.

Buffering is nothing but the time taken to fill the buffer with the frames which are sufficient to play the video. Always we need to maintain a minimum number of frames in buffer to have a uninterrupted play of video. If the rate of filling is lesser are equal to the rate of playing the video then the buffer is out of stock for frames which leads to buffering.

This paper is an attempt to avoid buffering by varying the compression ratio of the frames. First we estimate the buffer capacity which in turn depends on availability of Bandwidth.

II. SOLUTION FOR BUFFERING

The goal here is to provide the best video streaming in the face of variable network bandwidth. Finally we need a best video resolution. But at the same time we like to avoid waiting for buffering. Buffering does help allot it can mask variations in the network performance . in fact even an interactive application like skype uses buffering in the order of 100ms or so. For streaming video applications several seconds of buffering seen as acceptable because it doesn't interactive. You can fill up the buffering in the start playing smoothly and then buffer covers the variations in the networks Bandwidth. So as the available bandwidth reduces the buffer drains little bit and as the bandwidth increases buffer fills up again But if the buffer does empty out, have a pause in the playback in that hampers the video experience. The common approach to solve this problem is

a. Encode video in multiple bit rates
b. Estimate connection available Bandwidth
c. Pick a video rate ≤ available Bandwidth

Video encoded is in multiple bit rates for example from low resolution standard definition for SD through various rates up to 1024 HD or higher. The video player at the client estimates network connection available bandwidth. It then takes the video rate slightly less than the available Bandwidth. So the rate of incoming data of the network exceeds the data rate playing out from the buffer on the client screen.

For time being the format of downloading video is fixed whether it is SD or HD or higher. In this paper we have gone for buffer based video coding. Depends on the available percentage of buffer capacity the coding is done. If the left out Buffer capacity is high then the coding must take lesser time and fill the buffer. According to the buffer capacity adaptive coding is needed. Variable coding or compression is implemented by using variable Size of the Block and Search Range.

Revised Manuscript Received on March 02, 2019.

SHAFEE VUNNISA SAYYAD, Research scholar, Rayalaseema University, kurnool, Associate Professor, SNIST, AP, INDIA. (E-mail: Shafiunnisa8@gmail.com)

DR. V.S.K. REDDY, Director, MRCET, Telangana, INDIA. (E-mail: Vskreddy2003@gmail.com)

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

III. MATCHING CRITERIA &INPUT VIDEO

Motion estimation s carried out two adjacent frames. The type of motion estimation we employ is a block-based one, in which the image is divided into several blocks that are non-overlapping and for each of this block, estimation of motion is done. The block that has to be searches is known as the candidate block and to search this, a search space is defined. After defining, the block is searched and is compared with the position in the reference block. To perform this comparison, there has to be a measure for matching. The matching criteria are mentioned bellow:

$$MSE = \frac{1}{n^2}\sum_{i=0}^{n-1}\sum_{j=0}^{n-1}\left(C_{ij} - R_{ij}\right)^2$$

$$MAD = \frac{1}{n^2}\sum_{i=0}^{n-1}\sum_{j=0}^{n-1}\left|C_{ij} - R_{ij}\right|$$

$$PSNR = 10Log_{10}\left[\frac{255^2}{MSE}\right]$$

Input Videos

	Duration (sec)	Frame rate	No. of frames	Width	Height
Suzie	9.933	15	149	176	144
Stefan	2.933	30	88	352	288

IV. RESULTS & CONCLUSION

Table 1 and 2 will give the summary of PSNR ratios and Search points for ES, DS, ARPS & NTSS Algorithms. For Suzie input with different block sizes and Range. Suzie input video is slow motion video.

Table1. PSNR and SP (ES, DS) for Variable block size and Variable search Range - Suzie

Block Size	P	ES		DS	
		PSNR	SP	PSNR	SP
16	7	34.375	184	34.312	11.424
	9	34.384	295	34.317	11.424
	11	34.388	431	34.319	11.424
8	7	35.436	204	35.216	12.234
	9	35.464	323	35.226	12.234
	11	35.478	465	35.230	12.234
4	7	36.889	210	36.177	13.229
	9	36.943	332	36.191	13.229
	11	36.985	481	36.198	13.229

Table2. PSNR and SP (ARPS, NTSS) for Variable block size and Variable search Range - Suzie

Block Size	P	ARPS		NTSS	
		PSNR	SP	PSNR	SP
16	7	34.254	6.21	34.346	16.11
	9	34.257	6.22	34.346	16.11
	11	34.259	6.23	34.346	16.11
8	7	35.142	6.45	35.307	17.55
	9	35.142	6.46	35.307	17.55
	11	35.157	6.47	35.307	17.55
4	7	36.165.	7.03	36.496	19.05
	9	36.173	7.042	36.496	19.05
	11	36.176	7.047	36.496	19.05

The Table 3 and 4 will give the summary of PSNR ratios and Search points for ES, DS, ARPS & NTSS Algorithms. For Stefan input with different block sizes and Range. Stefan input video is fast motion video.

Table3. PSNR and SP (ES, DS) for Variable block size and Variable search Range - Stefan

Block Size	P	ES		DS	
		PSNR	SP	PSNR	SP
16	7	23.9885	204.28	23.006	19.71
	9	24.2	327.27	23.05	20.962
	11	24.24	479.1	23.064	21.11
8	7	25.649	214	23.577	19.798
	9	25.943	342	23.605	20.514
	11	26.018	496	23.611	20.654
4	7	27.409	217	24.553	19.56
	9	27.73	346	24.571	19.9
	11	27.85	504	24.576	19.98

Table4. PSNR and SP (ARPS, NTSS) for Variable block size and Variable search Range - Stefan

Block Size	P	ARPS		NTSS	
		PSNR	SP	PSNR	SP
16	7	23.6748	8.04	23.546	22.132
	9	23.8249	8.15	23.546	22.132
	11	23.8353	8.20	23.546	22.132
8	7	25.024	8.29	24.596	23.218
	9	25.203	8.37	24.596	23.218
	11	25.220	8.40	24.596	23.218
4	7	26.164	9.05	25.6624	24.193
	9	26.302	9.11	25.6624	24.193
	11	26.308	9.14	25.6624	24.193

Table 5 and 6 given the summary of PSNR and SP for DS, ES,NTSS and ARPS algorithms for suzie and Stefan videos.

At the next instant of time bandwidth available is unpredictable depends on the network traffic.

To calculate the PSNR and SP for buffer based coding, repeatedly executed the same algorithms and listed out. All average PSNR's and SP's are quiet different for the same algorithm and the same input due to the available bandwidth at that instant of time.

Table5. Buffer based. PSNR and SP -5 iterations of available Band width - Suzie

Iteration	Variable	ES	DS	NTSS	ARPS
1	PSNR	35.204	34.975	35.009	34.919
	SP	213.33	13.177	17.083	6.476
2	PSNR	35.19	34.96	35.04	34.91
	SP	226	13.17	17.06	6.47
3	PSNR	35.25	35.047	35.042	34.96
	SP	218	13.17	17.04	6.44
4	PSNR	35.05	34.85	34.86	34.82
	SP	223	13.07	16.94	6.41
5	PSNR	35.31	35.05	35.08	34.98
	SP	221	13.16	17.1	6.46

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Table6. Buffer based. PSNR and SP -5 iterations of available Band width - Stefan

Iteration	Variable	ES	DS	NTSS	ARPS
1	PSNR	25.038	23.410	23.28	24.44
	SP	244	16.63	21.67	8.27
2	PSNR	24.77	23.34	23.37	24.25
	SP	218	16.57	21.54	8.23
3	PSNR	25.43	23.59	23.65	24.74
	SP	242	16.76	21.84	8.40
4	PSNR	25.05	23.38	23.71	24.50
	SP	253	16.69	21.73	8.28
5	PSNR	25.1	23.49	23.29	24.51
	SP	219	16.67	21.59	8.3

Fig. 1 & 2 are the comparison of buffer based PSNR and SP's of various algorithms with respect to each frame.(Stefan)

Fig. 3 & 4 compares the buffer based PSNR's for ES and DS algorithms respectively with the PSNR's of Block Size 16, 8, & 4, Search Range 7. (Stefan)

Fig.5 compares the search points for ES algorithm with the SPs of Block Size 16, 8, & 4, Search Range 7. (Stefan)

Fig.6 compares the search points for ARPS algorithm with the SPs of Block Size 16, 8, & 4, Search Range 7. (Suzie)

Fig. 7 & compares the buffer based PSNR's for ARPS and ES algorithms respectively with the PSNR's of Block Size 16, 8, & 4, Search Range 7. (Suzie)

By comparing all the results we can conclude that frame to frame the adaptive coding is done. Buffering is avoided by maintaining the sufficient number of frames in buffer with different PSNR levels.

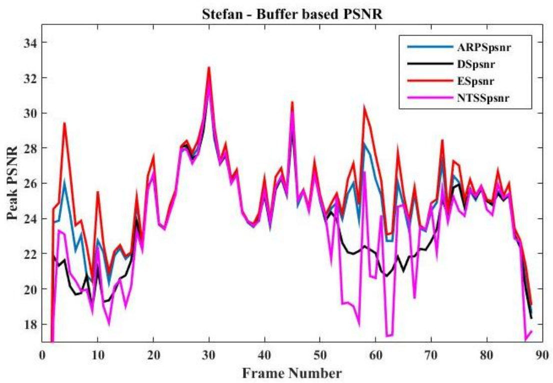

Fig1. Buffer based PSNR for various algorithms – Stefan

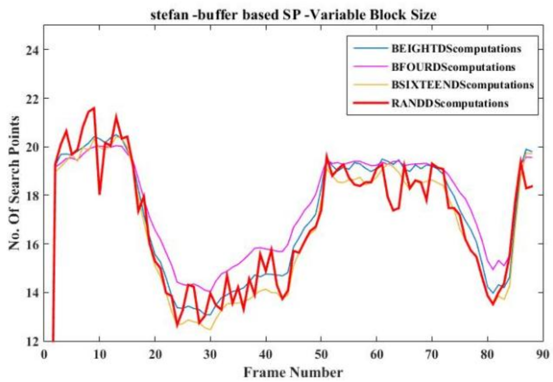

Fig2. Buffer based Search Points for various algorithms – Stefan

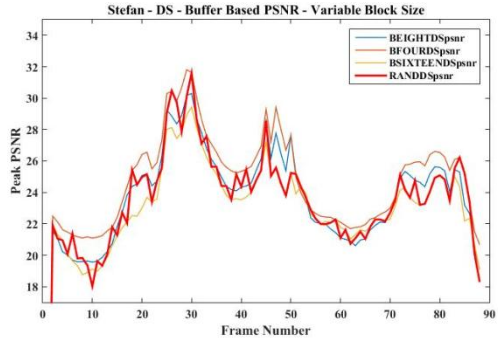

Fig3. Buffer based PSNR for DS algorithm, variable block Size - Stefan

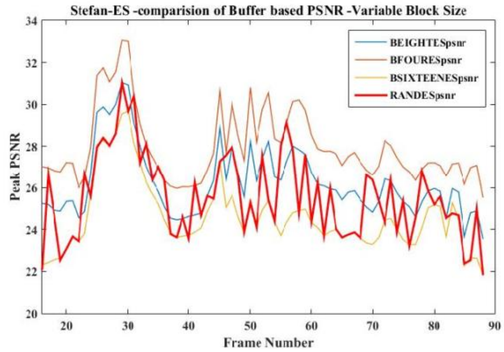

Fig4. Buffer based PSNR for ES algorithm, variable block Size – Stefan

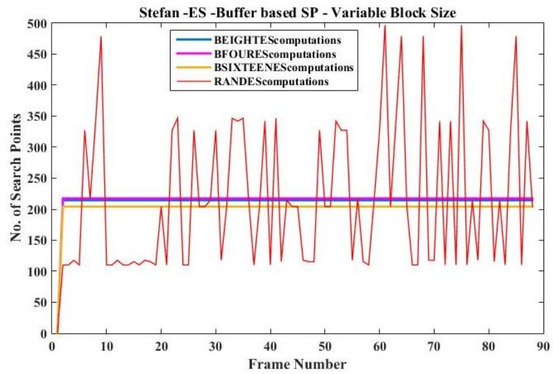

fig5. buffer based search points for es algorithm, variable block size - stefan

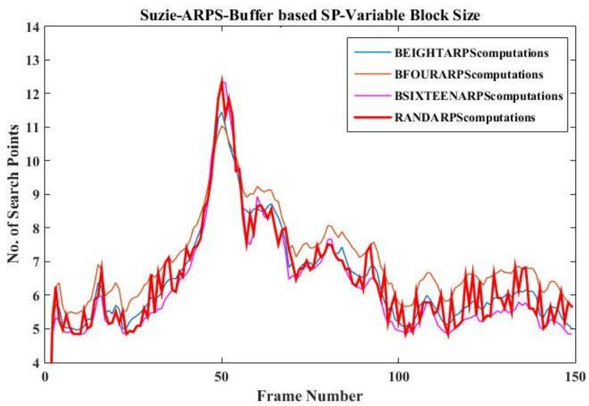

Fig6. Buffer based Search points ARPS algorithm, variable block Size - Suzie

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

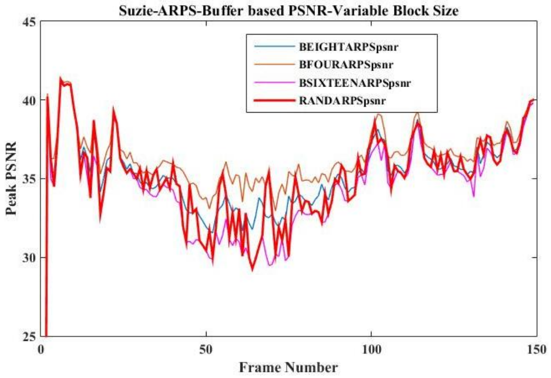

Fig7. Buffer based PSNR for ARPS algorithm, variable block Size – Suzie

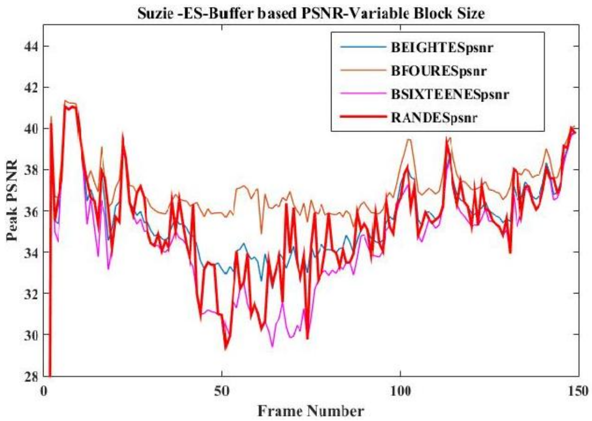

Fig.8.Buffer based PSNR for ES algorithm, variable block Size - Suzie

REFERENCES

http://en.wikipedia.org/wiki/H.264
http://en.wikipedia.org/wiki/Video_compression
http://en.www.wikipedia.com
http://www.chiariglione.org
http://www.vodex.com

[1] sherin Abdelaal, yuhanis yusof. “Aaptation of motion Estimation Algorithmsfor Rea time Vdeo Sequesnces”, 2018 Intrnation conference on Computer and Aplications (ICCA),

[2] Muhanned Al Anwer, M.A.Alia, Kh.Razaz “Detecting and tracking of ASD using new block flow in ultrasound (a novel approach)”, 2013 International conference on computing, electrical and Electronic Engineering.

[3] I.E.G. Richardson, (2003). H.264 and MPEG-4 Video Compression: Video Coding for Next generation Multimedia, John Wiley & Sons, Ltd, 0-470-84837-5, UK

[4] shafee vunnisa sayyad(2016), “Flow Management for Video applications in Wireless Local Area Network”ccsn 2016 IEEE conference, west Bengal, December 2016

[5] Iain E.Richardson, (2010) .*The H.264 Advanced Video Compression Standard* John Wiley & Sons ltd, 978-0-470-51692-8, UK.

[6] Lai Mingche, Dai Kui, Lu Hong-yi, Wang Zhi-ying, 2006, A Novel Data-Parallel Coprocessor for Multimedia Signal Processing”,IEEE Computer Society.

[7] Kue-Hwan Sihn, Hyunki Baik, Jong-Tae Kim, Sehyun Bae, Hyo Jung Song, (2009), Novel Approaches to Parallel H.264 Decoder on Symmetric Multicore Systems, *Proceedings of IEEE.*

[8] Kun Ouyang Qing Ouyang Zhengda Zhou, (2009), Optimization and Implementation of H.264 Encoder on Symmetric Multi- Processor Platform, *Proceedings of IEEE Computer Society, World Congress on Computer Science and Information Engineering.*

[9] Nuno Roma and Leonel Sousa, 2002, Efficient and Configurable Full-Search Block-Matching Processors, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 12, No. 12, December 2002.