Improving HEVC Coding Efficiency Using Virtual Long-Term Reference Pictures

Buddhiprabha Erabadda, Thanuja Mallikarachchi, Gosala Kulupana and Anil Fernando
Centre for Vision Speech and Signal Processing, University of Surrey, United Kingdom
Email: {e.buddhiprabha, g.kulupana, w.fernando}@surrey.ac.uk, tmallikarachchi@cardiffmet.ac.uk

***Abstract*—Inter-frame prediction in HEVC uses two types of reference pictures: short-term and long-term. Out of these, long-term reference (LTR) pictures enable exploiting correlation among frames with extended temporal distances. In addition, LTR pictures improve the inter-frame prediction where video scenes are repeated such as in TV-series episodes, news broadcasts and movies. In this context, this paper proposes an algorithm to calculate LTR pictures using artificially generated virtual reference frames for static-camera scenes. The experimental results demonstrate an average coding improvement of 2.34% in terms of Bjøntegaard Delta Bit Rate(BDBR), when compared with the HEVC reference encoder HM16.8.**

***Index Terms*—HEVC, inter-prediction, long-term reference, virtual reference frames**

I. Introduction

Ever increasing advancements in consumer electronics, multimedia technologies, novel video formats, and high-resolution video contents demand continuous improvements in video compression techniques. In this regard, High Efficiency Video Coding (HEVC) is the latest stable video coding standard, and it reports $\approx 50\%$ more compression gain compared to its predecessor H.264 [1]. Yet, the standardisation bodies continue to work on next-generation video encoding algorithms to match the ever increasing demand in video consumption.

In this context, this paper proposes a novel technique to achieve higher coding gains for video sequences with repeated static camera scenes using virtual long-term reference pictures.

II. Background and Related Work

HEVC employs intra- and inter-prediction to remove the redundancies in the spatial and temporal domains of video frames, respectively. Inter-prediction can be further subdivided depending on the type of the reference pictures used for prediction. Two types of reference pictures are used, namely short-term and Long-Term Reference (LTR) pictures. The difference between the two types are attributed to the temporal distance from the current picture, where the former has a shorter span compared to the latter [2]. The work proposed in this paper is based on LTR pictures, hence the following discussion focuses on LTR usage and related work.

HEVC supports up to a total of 32 LTR pictures. The standard, however, does not specify how these reference pictures should be selected, giving the encoder the flexibility to do so.

This work was supported by the CONTENT4ALL project, which is funded under European Commission's H2020 Framework Program (Grant number: 762021).

Improving coding efficiency employing LTR pictures has been considered on numerous occasions in the recent literature. For example, Zhang *et al.* [3] propose a method for surveillance cameras to use the background as a LTR picture. Paul *et al.* [4] propose a method where the most common frame in a given scene is used as the LTR picture. Zuo and Yu [5] use *k-means* clustering for scene change frames to select LTR pictures in hierarchical B-picture-based encoding.

While these existing methods achieve increased compression, there exist avenues for further coding gains using virtual reference frames (VRFs), where frames are created and designated as LTR frames for the sequence. This enables aggregating the common components across multiple frames, which is not possible if an existing frame is selected. The focus of the work is on static-camera scenarios, where the video sequence returns to shots from the same camera (e.g., the same scene is revisited throughout the sequence). To this end, this paper proposes a method to exploit redundancy in repeated scenes captured through static cameras in TV-series episodes to achieve higher compression efficiencies.

III. Proposed Method

The objective is to use VRFs as LTR pictures to achieve an increased coding efficiency for sequences with repeated scenes. Here, a virtual frame is a picture that is synthetically created from information accumulated from multiple frames. This work extends from our previous work in [6] and the remainder of this section explains the key steps involved.

The frames in the sequence are clustered using k-means clustering [7] in order to group all frames corresponding to a particular scene, into a single cluster. In this case, the Euclidean Distance(ED) is used to assign the frames to clusters and it is calculated as,

$$ED = \sqrt{\sum_{i=1}^{n}(p_i - \mu_i)^2}, \tag{1}$$

where p_i and μ_i represent cluster members and the cluster centre, respectively, in the n-dimensional Euclidean space,where $i \in n$. Here, elbow method [8] is used to automatically identify the number of distinct scenes in a given video sequence. The number of clusters are selected to minimise the cost parameter defined as the Within Cluster Sum of Squared Errors (ζ) [8] for K clusters, where $K \in \{1, 2, 3, .., 10\}$. In this case, ζ for a particular cluster in set of K clusters is calculated as,

TABLE I
DETAILS OF THE SEQUENCES

Sequence Name	Resolution	No. of frames	No. of VRFs	No. of repeated scenes
Koombiyo-1	1920x1080	644	1	3
Koombiyo-2	1920x1080	980	2	6
DeweniInima-1	1920x1080	700	3	11
DeweniInima-2	1920x1080	1500	2	18
Sidu-1	1280x720	700	1	14
Sidu-2	1280x720	3300	1	5
Johnny/ KristenAndSara	1280x720	1200	2	24

$$\zeta = \sum_{k=0}^{K} \sum_{x_i \in k} (x_i - \mu_k)^2 \tag{2}$$

where μ_k represents the cluster centre of the k^{th} cluster.

Once the frames are clustered, next step involves creating masks for the objects that exists in the scene. To this end, an initial masks for the objects are created manually using GIMP [9] followed by a Recurrent Video Object Segmentation Algorithm (RVOS) [10] to propagate masks in the video sequence. This is made possible due to the continuous nature of frames within a particular scene. Providing the initial mask to the RVOS algorithm increases the accuracy of the video object segmentation algorithm.

Next, the background of a given scene is calculated using an accumulated weighted running average algorithm [11]. Once the objects and the background are extracted, the final step is to create the VRF by combining them. To select the instance of a given object, all instances from that object are clustered using k-means clustering and the cluster centre is selected as the object instance for the VRF. Finally, the VRFs are appended to the beginning of the video sequence before sending them to the decoder. In addition, these frames are signalled as LTR frames within HEVC bit streams using syntax elements available for reference picture management [2].

It is worth to note that the clustering step could be processed without masks in case where there is only one object, saving a significant amount of time and computations.

IV. RESULTS AND DISCUSSION

The proposed algorithm is implemented in HEVC reference software HM16.8 [12]. Test sequences are episodes from TV-series *Koombiyo*, *Deweni Inima*, and *Sidu* and two standard test sequences in the common test configurations [13]. Table I provides the details of the test sequences utilised in the experiments.

The results in the Table II compare the proposed work with HM16.8 in the *LowDelay_P_Main* configuration. All tests were completed in Intel(R) Core i9 CPU @ 3.6 GHz system with 32 GB RAM and Windows10 64-bit operating system.

The results demonstrate that the proposed method achieves considerable coding gains for the sequences, measured in terms of BDBR, with an average gain of 2.34%. Although it is reasonable to assume that the coding gains should increase with the number of scene changes, it is observed that this increase is not linear. For example, *Koombiyo-2* and *Sidu-2* sequences have 6 and 5 scene changes respectively, however, the *Koombiyo-2* sequence reports a significantly higher coding gain. It was also observed that the *Sidu-2* sequence has much common background across different scenes, making the predictions possible from the previous frame, even though it is from a different scene.

TABLE II
PERFORMANCE OF THE PROPOSED METHOD(LOW DELAY P)

Sequence	Proposed vs HM16.8	
	BDBR (%)	**BD-PSNR (dB)**
Koombiyo-1	-0.057	0.002
Koombiyo-2	-1.000	0.030
DeweniInima-1	-3.065	0.105
DeweniInima-2	-2.888	0.101
Sidu-1	-2.427	0.093
Sidu-2	-0.231	0.009
Johnny/ KristenAndSara	-6.739	0.106
Average	**-2.344**	**0.064**

Finally, it should be noted that the additional bits from VRFs have been incorporated in the calculations. Therefore, it is evident that the additional bits from the VRFs are trivial when compared with the gains obtained from the VRFs.

V. FUTURE WORK

At present, VRFs are transmitted from the encoder, resulting in an addition to the bits being transmitted. Future work will focus on creating the VRF at the decoder side to enable further coding gains. Furthermore, extending the proposed approach to moving-camera scenarios will also be considered.

REFERENCES

[1] G. J. Sullivan, J. Ohm, W-J Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[2] Rickard Sjöberg and Jill Boyce, "Hevc high-level syntax," in *High Efficiency Video Coding (HEVC)*, pp. 13–48. Springer, 2014.

[3] Xianguo Zhang, Luhong Liang, Qian Huang, Yazhou Liu, Tiejun Huang, and Wen Gao, "An efficient coding scheme for surveillance videos captured by stationary cameras," in *Visual Communications and Image Processing 2010*. International Society for Optics and Photonics, 2010, vol. 7744, p. 77442A.

[4] Manoranjan Paul, Weisi Lin, Chiew-Tong Lau, and Bu Sung Lee, "A long-term reference frame for hierarchical B-picture-based video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 10, pp. 1729–1742, 2014.

[5] Xu-guang Zuo and Lu Yu, "Long-term prediction for hierarchical-B-picture-based coding of video with repeated shots," *Frontiers of Information Technology & Electronic Engineering*, vol. 19, no. 3, pp. 459–470, 2018.

[6] Buddhiprabha Erabadda, Thanuja Mallikarachchi, Gosala Kulupana, and Anil Fernando, "Virtual Frames as Long-Term Reference Frames for HEVC Inter-Prediction," in *2020 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, 2020, pp. 1–2.

[7] John A Hartigan and Manchek A Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[8] Trupti M Kodinariya and Prashant R Makwana, "Review on determining number of Cluster in K-Means Clustering," *International Journal*, vol. 1, no. 6, pp. 90–95, 2013.

[9] GIMP Team et al., *GIMP: GNU Image Manipulation Program*, GIMP Team., 2019.

[10] Carles Ventura, Miriam Bellver, Andreu Girbau, Amaia Salvador, Ferran Marques, and Xavier Giro-i Nieto, "RVOS: End-to-end recurrent network for video object segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5277–5286.

[11] Gary Bradski and Adrian Kaehler, "OpenCV," *Dr. Dobb's journal of software tools*, vol. 3, 2000.

[12] "HM 16.8," https://hevc.hhi.fraunhofer.de/trac/hevc/browser/tags/HM-16.8.

[13] Frank Bossen et al., "Common test conditions and software reference configurations," *JCTVC-L1100*, vol. 12, 2013.