A NOVEL FRAMEWORK OF FRAME RATE UP CONVERSION INTEGRATED WITHIN HEVC CODING

Guo Lu, Xiaoyun Zhang, Zhiyong Gao*

Institute of Image Communication and Network Engineering
Shanghai Jiao Tong University, China

ABSTRACT

Frame rate up conversion (FRUC) is a key technology for providing high frame rate video with better visual quality, but at the cost of huge computational complexity and bandwidth consumption. In this paper, a standard compatible framework of FRUC integrated within high efficiency video coding (HEVC) is innovatively proposed. First, spatial and temporal motion candidates of skip mode are collected and employed for the motion estimation of FRUC, where FRUC selects the best motion vectors (MVs) according to the texture-aware criterion based on coding block size. Then, the interpolated frame is handed over to HEVC coding and mostly encoded as skip mode, which consumes very limited bits. Experimental results show that the proposed method has a great advantage both in computational complexity and bandwidth consumption compared with the traditional method of FRUC cascaded with coding.

***Index Terms*—** Frame rate up conversion, motion estimation, motion compensation, high efficiency video coding

1. INTRODUCTION

Frame rate up conversion (FRUC) is widely used to reduce the motion blur of hold-type displays such as liquid crystal display. Recent growth of multimedia devices have also led to the demand for FRUC, since it can enhance video temporal resolution which makes consumers enjoy better visual experience.

Most FRUC algorithms utilize motion information to perform interpolation along motion trajectory. Such FRUC method is also referred to as motion-compensation frame rate up conversion (MC-FRUC) [1, 2, 3, 4, 5]. A typical MC-FRUC scheme consists of two steps: motion estimation (ME) and motion compensation interpolation (MCI). The process of ME searches the motion vectors (MVs) to approximate the true motion between consecutive frames. Then MCI uses the estimated MVs to interpolate the intermediate frame. However, in order to obtain accurate MVs or faithful interpolated frame, these algorithms have to estimate multiple MV fields [4] or calculate a large number of weighting factors [1], which leads to high computational complexity.

Some researches [6, 7] utilize the existing MVs from decoder to alleviate the computation burden of FRUC. However, the reconstructed video quality depends on the decoded frames. If videos are transmitted in limited bandwidth and the quality of decoded frames has already been deteriorated, it is difficult to reconstruct good quality intermediate frames even a sophisticated FRUC method is adopted. In [8], the encoder calculates and sends side information to improve the quality at decoder side. It achieves better performance than the FRUC only using the decoder information. However, this method is not standard compatible and needs a non-standard decoder. In addition, for the terminal devices such as TVs, high quality FRUC has been traditionally designed and implemented using Application Specific Integrated Circuit (ASIC), while the video is decoded by specific hardware module integrated within a system on chip (SoC). Such hardware architecture makes it difficult to share the information such as MVs between the decoder and FRUC. Therefore this kind of FRUC at decoder side is not feasible in practical application.

Considering power and cost, many portable devices do not provide hardware implemented FRUC function. Hence it is essential and worthwhile to make an effort to conduct FRUC at the encoder side, and the converted high frame rate video is transmitted and displayed directly for the consumers with better visual quality.

In this paper, we propose a novel framework of FRUC integrated within HEVC coding. By investigating the intrinsic relationship between Bi-prediction in coding and frame interpolation in FRUC, we build a system where FRUC is embedded into the encoding loop. In the proposed framework, ME procedure of FRUC is based on the motion candidates of *skip mode*, and the motion compensation interpolation of FRUC is modeled as the Bi-prediction in HEVC coding. Since MVs are shared between FRUC and coding, a new texture-aware criterion based on coding block size is designed to select the best motion candidate. Then the interpolated frames with the corresponding best MVs are mostly encoded in *skip mode*, which leads to low bit rate consumption. More importantly, this realization is compatible with the coding standard and the encoded bit stream of the up-converted video sequence

*Corresponding author

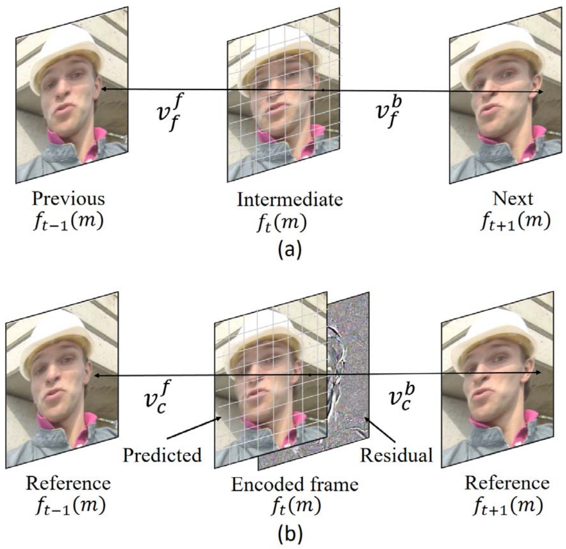
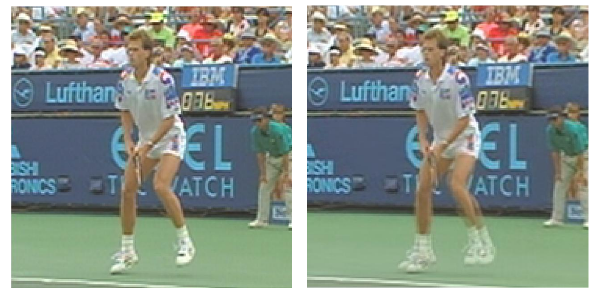


Fig. 1. (a)Interpolation in FRUC. (b)Bi-prediction in HEVC coding.



(a) (b)

Fig. 2. (a) Original frame. (b) Bi-predicted frame using skip mode.

can be decoded by a standard HEVC decoder. Experimental results show that the proposed method has a great advantage both in computational complexity and bandwidth consumption compared with the traditional method of FRUC cascaded with coding.

The rest of paper is organized as follows. Section 2 analyses the relationship between FRUC and Bi-prediction in coding. Then the proposed framework is introduced in Section 3. Section 4 presents the experimental results, and the conclusion is drawn in Section 5.

2. ANALYSIS OF BI-PREDICTION AND FRUC

Bi-prediction in video coding exploits the motion compensation prediction to eliminate the temporal dependence [9], while FRUC utilizes the temporal dependence to predict new frames. In this section, we discuss how to integrate frame interpolation of FRUC with Bi-prediction of the coding process in a tightly coupled way.

2.1. Unified model of FRUC and Bi-prediction in HEVC coding

In traditional FRUC methods, motion is assumed to be continuous and the intensity of image pixels remains approximately constant along motion trajectories. Therefore, intermediate frame $f_t(m)$ in Fig.1(a) can be estimated by adjacent $f_{t-1}(m)$ and $f_{t+1}(m)$. Namely,

$$\hat{f}_t(m) = \frac{1}{2} \cdot [f_{t-1}(m - v_f^f) + f_{t+1}(m + v_f^b)] \qquad (1)$$

where v_f^f and v_f^b represent the forward and backward MV used for interpolation in FRUC.

Fig.1(b) illustrates the Bi-prediction procedure in HEVC. $f_{t-1}(m)$ and $f_{t+1}(m)$ are the forward and backward reference frames of $f_t(m)$, respectively. For motion vector v_c^f and v_c^b obtained by ME algorithm in coding, the motion compensation prediction for predicted frame $\hat{f}_t(m)$ is formulated as follow:

$$\hat{f}_t(m) = \frac{1}{2} \cdot [f_{t-1}(m - v_c^f) + f_{t+1}(m + v_c^b)] \qquad (2)$$

Generally, residual frame is compensated to reduce prediction error. However, it may consume too many bits to improve the quality of the encoded frame. HEVC provides *skip mode* to achieve better rate-distortion performance. When *skip mode* is chosen for Bi-prediction, no residual is compensated. Therefore, the encoded frame is actually interpolated by adjacent frames based on the MV estimated during coding. If the (v_f^f, v_b^f) and (v_f^c, v_b^c) are the same or similar, then interpolation of FRUC in (1) and Bi-prediction of coding in (2) can be unified together as the same problem. In this situation, the process of encoding a frame is equivalent to the interpolation in FRUC. This provides us a novel perspective on integrating FRUC with HEVC coding at encoder side.

2.2. Shared motion information of coding and FRUC

In the above subsection, we have analyzed the feasibility of integrating FRUC with HEVC coding based on the unified model of (1) and (2). In fact, in order to ensure the quality of intermediate frame, (v_c^f, v_c^b) should reflect motion trajectory. One solution is to utilize ME algorithms in FRUC to replace the original ME algorithm in the coding process. However, this approach will yield high computational complexity and decrease the coding efficiency. We adopt an alternative solution to share MVs in coding with FRUC as much as possible.

In HEVC, MVs in Bi-prediction can be obtained through two approaches: ME in *non-skip mode* and ME in *skip mode*. Instead of searching new MVs, ME in *skip mode* chooses the

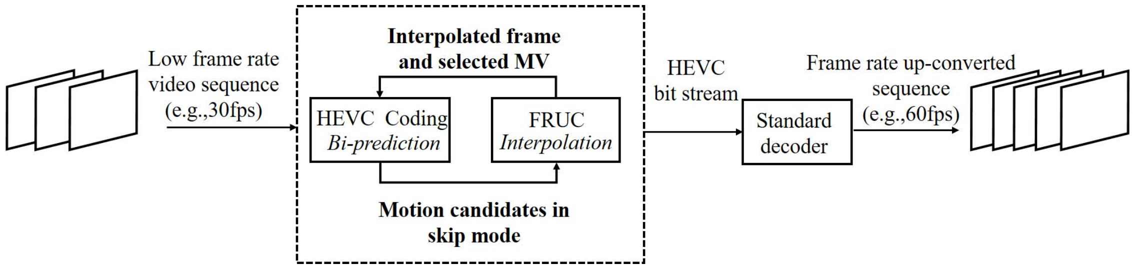


Fig. 3. Framework of the proposed FRUC integrated within HEVC coding

optimal MV in a set of motion candidates which are derived from spatial or temporal neighbours. This can reduce the bitrate overhead of motion vectors since the bitstream transmits an index only in *skip mode* [9]. Also, it imposes smoothness constrain implicitly to the motion vector field which can improve the quality of intermediate frame and decrease the irregular MVs caused by ME in *non-skip mode*. Therefore, it is expected that such spatial and temporal motion candidates in *skip mode* is also very useful for FRUC.

An experiment is conducted to evaluate the effectiveness of MVs in *skip mode* for FRUC. We encode an intermediate frame by setting the only use of *skip mode* for all blocks in coding process. In Fig.2, we find the frame encoded by *skip mode* is close to original frame in most regions even no residuals is compensated, while the mismatch only occurs in motion boundary. It verifies that MVs collected during the coding process are valuable for the interpolation of FRUC.

3. THE PROPOSED FRAMEWORK OF FRUC INTEGRATED WITHIN HEVC CODING

3.1. Framework of the proposed method

The proposed novel framework of FRUC integrated within HEVC coding is shown in Fig.3. A video sequence at low frame rate, e.g., 30fps, is input to the system. It is then up-converted and coded in a unified FRUC and HEVC coding process. A high frame rate video sequence, e.g., 60fps, is finally output as a compressed bitstream which can be decoded by a standard HEVC decoder.

The framework can be generalized into two procedures. Firstly, spatial and temporal motion candidates of *skip mode* during HEVC coding is collected and employed for the motion estimation of FRUC, where FRUC selects the best MV according to the proposed texture-aware criterion and interpolate the intermediate frame. This procedure actually interpolates a new frame in the encoding loop, and the computation complexity of motion estimation is significantly reduced because of the adoption of effective MVs from coding.

Secondly, the interpolated frame and the corresponding best MVs of FRUC are then handed over to HEVC Bi-prediction coding. Such interpolated frame is mostly encoded as *skip mode* with no residual, and the bit cost is very limited.

3.2. Texture-aware criterion based on coding block size

In order to select the optimal motion candidate, we proposed a texture-aware criterion. Here we assume the motion is consistent in $f_{t-1}(m)$, $f_t(m)$ and $f_{t+1}(m)$. We use the sum of bilateral absolute differences (SBAD) to measure the reliability of motion candidates v^f and v^b in *skip mode*.

$$SBAD(v^f, v^b) = \sum_{m \in S} |f_{t-1}(m - v^f) - f_{t+1}(m + v^b)| \tag{3}$$

S indicates search range.

However, SBAD may fail to choose the reliable MV when an object with complex texture and a background with homogeneous texture are simultaneously present. In this situation, matching blocks in homogeneous texture region has a greater chance to be selected. Therefore it is essential to impose a texture complexity constrain to improve the robustness of interpolation in FRUC.

In HEVC, the coding block size varies from 64x64 to 8x8 and the corresponding depth level increases from 0 to 3 [9]. It is observed that the coding block size has a strong relationship with the texture complexity, i.e., when the region has complex texture, it tends to be encoded with a small block size; otherwise, a large block size is preferred. Therefore, we utilize the depth level of the search blocks in reference frames to determine the degree of texture complexity. Namely,

$$\rho(v^f, v^b) = \sum_{m \in S} (D_{t-1}(m - v^f) + D_{t+1}(m + v^b)) \tag{4}$$

where ρ means the texture complexity. $D_{t-1}(m)$ and $D_{t+1}(m)$ represent the depth level in the reference frames. The proposed texture-aware criterion is defined as follows:

$$Dis(v^f, v^b) = SBAD(v^f, v^b) - \lambda \cdot \rho(v^f, v^b) \tag{5}$$

λ is a weight factor. The best motion vector is determined by minimizing Dis. This method can improve the performance of ME for FRUC and does not introduce additional computational complexity owing to the effective utilization of the coding information.

Table 1. Objective evaluation of the proposed method compared with three traditional FRUC algorithms plus coding

	AOBMC[1]		DSME[2]		NLM[10]	
Sequence	BDBR	BD-PSNR	BDBR	BD-PSNR	BDBR	BD-PSNR
BD	-18.3	0.32	-12.2	0.21	-22.5	0.40
BQ	-29.6	0.55	-22.2	0.42	-25.4	0.43
CT	-13.9	0.25	-9.8	0.16	-14.7	0.25
KI	-17.4	0.42	-12.8	0.31	-13.3	0.33
PS	-14.5	0.40	-11.5	0.30	-10.7	0.27
Average	-18.7	0.38	-13.7	0.28	-17.3	0.34

3.3. Adaptive motion re-estimation

As we mentioned before, motion candidates in *skip mode* are reliable enough in most regions except for motion boundaries, such as the feet in Fig.2(b). To address this issue, an adaptive motion re-estimation method is proposed. We measure the smoothness of motion vector field by ζ

$$\zeta = \frac{Dis_{cur}}{\sum_{i \in N} Dis_i / N} \tag{6}$$

Dis_{cur} is the optimal Dis in current location. N represents the locations where optimal MVs are same with the current location in neighbour blocks.

In background region or interior of moving objects, Dis tends to a stable value. Therefore, if $Th_l < \zeta < Th_h$, we classify this block into homogeneous region where MV in *skip mode* is reliable enough; otherwise we should implement motion estimation to track more accurate motion trajectory. A lot of ME algorithms in FRUC [1, 11, 12] can be adopted. In order to reduce computational complexity, we adopt 3DRS [11] in our method.

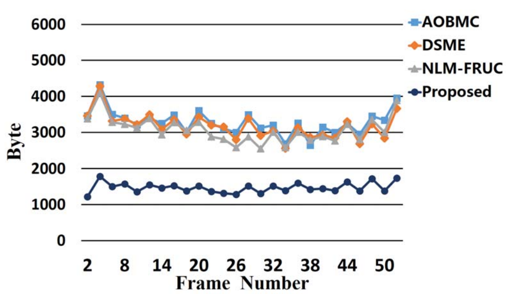


Fig. 4. Bitrate cost (Byte) comparison for interpolated frames in Kimono when QP is set as 32.

4. EXPERIMENTS

The proposed framework is implemented in the open source HEVC encoder x265 [13] with the GOP structure of IBBBP, where the first and the third B are the to be interpolated frames. Quantization parameter (QP) of 22,27,32,37 are used, and the interpolation of FRUC uses block size of 16x16. Experiments are conducted on five image sequences with full HD (1920x1080) resolutions and different motion characteristics: BasketballDrive (BD), BQTerrace (BQ), Cactus (CT), Kimono1 (KI), ParkScene (PS). Th_l and Th_h are set as 0.5 and 2, respectively. Even frames in the original test sequences are skipped and saved as the ground truth for the intermediate frames of FRUC.

Three FRUC algorithms including AOBMC [1], DSME [2] and NLM [10] are tested, and the corresponding up converted sequences are then encoded using x265 with the same setting. They are used as benchmarks. The results are shown in Table.1, where BDBR and BD-PSNR [14] are used to assess the overall performance. We can see that the proposed method can achieve 13%-18% bitrates reduction or 0.28-0.38 dB increase in PSNR on average for the up-converted coded bitstream. Fig.4 shows the bits used for the intermediate frames in different methods. As we can see, the proposed method reduces bit costs of interpolated frames significantly in comparison with other methods.

Table 2. Computational complexity in comparison with traditional FRUC algorithms

Method	Number of AD
AOBMC	$> 32^2 \cdot 16^2 = 262144$
DSME	$2(32^2 \cdot 16^2 + 4^2 \cdot 16^2) = 532480$
NLM	$32^2 \cdot 16^2 = 262144$
Proposed	$\eta \cdot 15 \cdot 16^2 + (1-\eta) \cdot 5 \cdot 16^2 = 1280 + 2560 \cdot \eta$

Computational complexity of existing works and the proposed framework are also evaluated. Since ME is the most computation-intensive part in FRUC, we calculate the number of absolute difference (AD) per block. For a fair comparison, we assume all the selected methods [1, 2, 10] adopt 16x16 block size and the search range is 32. In our method, the number of motion candidates in *skip mode* is set as 5 and 3DRS [11] utilizes 10 candidates to obtain more accurate trajectory. η represents the possibility that current block may need motion re-estimation. In most cases, $\eta < 0.5$. Table.2 indicates that our method has a great advantage over traditional methods.

5. CONCLUSION

In this paper, a novel framework of FRUC integrated within HEVC coding is proposed. By exploiting motion information in *skip mode*, the intermediate frame is interpolated at low computational complexity. More importantly, bitrate cost for the intermediate frame is reduced significantly. Therefore it can be used in bandwidth limited communication channels to improve the visual quality by FRUC. Experiments show the overall quality is superior to the methods of FRUC plus coding in BDBR and BDPSNR.

6. REFERENCES

[1] B.-D Choi, J.-W. Han, C.-S. Kim, and S.-J. Ko, "Motion-compensated frame interpolation using bilateral motion estimation and adaptive overlapped block motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 4, pp. 407–416, 2007.

[2] D.-G. Yoo, S.-J. Kang, and Y. H. Kim, "Direction-select motion estimation for motion-compensated frame rate up-conversion," *J. Display Technol.*, vol. 9, no. 10, pp. 840–850, 2013.

[3] S.-J. Kang, S. Yoo, and Y. H. Kim, "Dual motion estimation for frame rate up-conversion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1909–1914, 2010.

[4] H. Liu, R. Xiong, D. Zhao, S. Ma, and W. Gao, "Multiple hypotheses bayesian frame rate up-conversion by adaptive fusion of motion-compensated interpolations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 8, pp. 1188–1198, 2012.

[5] H. Lim and H. W. Park, "A region-based motion-compensated frame interpolation method using a variance-distortion curve," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 518–524, 2015.

[6] A.-M. Huang and T. Nguyen, "A multistage motion vector processing method for motion-compensated frame interpolation," *IEEE Trans. Image Process.*, vol. 17, no. 5, pp. 694–708, 2008.

[7] A.-M. Huang and T. Nguyen, "Correlation-based motion vector processing with adaptive interpolation scheme for motion-compensated frame interpolation," *IEEE Trans. Image Process.*, vol. 18, no. 4, pp. 740–752, 2009.

[8] G. Dane, K. El-Maleh, and Y.-C. Lee, "Encoder-assisted adaptive video frame interpolation," in *Proc. ICASSP*, 2005, pp. 349–352.

[9] G. J Sullivan, J.-R. 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, 2012.

[10] R. Dehghannasiri, S. Reza Soroushmehr, and S. Shirani, "Frame rate up-conversion using nonparametric estimator," in *Proc. ICIP*, 2014, pp. 3872–3876.

[11] G. De Haan, P. W. Biezen, H. Huijgen, O. Ojo, et al., "True-motion estimation with 3-d recursive search block matching," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, no. 5, pp. 368–379, 1993.

[12] S. Dikbas and Y. Altunbasak, "Novel true-motion estimation algorithm and its application to motion-compensated temporal frame interpolation," *IEEE Trans. Image Process.*, vol. 22, no. 8, pp. 2931–2945, 2013.

[13] x265 HEVC Encoder. Available: `https://bitbucket.org/multicoreware/x265/wiki/Home/`.

[14] G. Bjontegaard, "Calcuation of average psnr differences between rd-curves," *Doc. VCEG-M33 ITU-T Q6/16, Austin, TX, USA, 2-4 April 2001*, 2001.