|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**  9th Meeting: Geneva, CH, 27 April – 7 May 2012 | Document: JCTVC-I0293 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | Merge candidate refinement for uni-predictive block | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Tomoyuki Yamamoto, Tomohiro Ikai  1-9-2 Nakase, Mihama-ku, Chiba-shi, Chiba 261-8520 JAPAN | Tel: Email: | +81-43-299-8526 [yamamoto.tomoyuki@sharp.co.jp](mailto:yamamoto.tomoyuki@sharp.co.jp) |
| *Source:* | SHARP Corporation | | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

This contribution proposes alternative merge candidates for uni-predictive block which replace combined merge candidate in bi-predictive block. The proposed method aims to improve coding efficiency of P Slice without introducing additional worst case computation increase in B Slice. The experimental results show that the proposed method introduces coding efficiency improvement of 0.3 % and 0.3 % for LP Main and LP HE10 cases respectively.

# Introduction

In the development of HEVC standard, more efforts have been put to improve coding efficiency of B Slice compared to P Slice. Those results in more sophisticated coding tools for B Slice (bi-predictive block) compared to P Slice (uni-predictive block). This contribution presents merge mode of uni-predictive block with better coding efficiency.

The proposed method applies additional merge candidates which are derived by refining the first merge candidate in merge candidate list when bi-prediction cannot be applied in a block. It is worth mentioning that similar approach (but for both P and B Slice) has proposed in the past meeting [1].

The additional candidates are used instead of combined merge candidate. That means that the worst case decoding scenario for merge mode is still B Slice, and the proposed method does not worsen the case.

# Proposed method

**Refined merge candidate**

In a block where bi-prediction cannot be applied, refined merge candidates are added to the merge candidate list instead of combined merge candidates.

The derivation process of refined merge candidates is as follows:

1) Select the first candidate in the merge candidate list. Let’s call it reference candidate.

2) Derive refined merge candidate by adding an offset to mvL0 of the reference candidate.

- The offset is either (+4, 0) or (-4, 0).

Up to two refined merge candidates could be derived and added to the candidate list. Note that no refined merge candidate is derived when there is no candidate in the list or the candidate list is already full.

No syntax change is needed.

# Experimental results

The proposed method is implemented on HM-6.0 and evaluation is done based on common test condition. Since the proposed method only affects P Slices, LP-MAIN and LP-HE10 conditions were tested. The results are shown in Table 1.

It is observed that the proposed method improves coding efficiency by 0.3 % for both LP-MAIN and LP-HE10 cases. It is especially good for Kimono where 0.9 % and 1.1 % gain were achieved for LP-MAIN and LP-HE10 respectively.

Table 1: Result of the proposed method (compared to HM-6.0)



# Additional experimental results

The proposed method can also be applied in B Slice when bi-prediction is restricted in smaller blocks. The contribution JCTVC- I0107 [2] proposes such restriction. The proposed method is implemented on the software used in JCTVC- I0107 and impact on coding efficiency is evaluated. The results of 8x4/4x8 bi-pred restriction with or without proposed method are shown in Table 2 and 3.

By comparing those results, it can be said that the proposed method improves coding efficiency by 0.1% for RA-MAIN, RA-HE10, LB-MAIN, and LB-HE10 cases when 8x4/4x8 bi-pred restriction based on JCTVC-I0107 is applied.

Table 2: Result of 8x4/4x8 bi-pred restriction with proposed method (compared to HM-6.0)



Table 2: Result of 8x4/4x8 bi-pred restriction in JCTVC-I0107 (compared to HM-6.0)



# Conclusion

This contribution proposes alternative merge candidates for uni-predictive block which replace combined merge candidate in bi-predictive block. The proposed method aims to improve coding efficiency of P Slice without introducing additional worst case computation increase in B Slice. The experimental results shows that the proposed method introduces coding efficiency change of 0.3 % and 0.3 % for LP MAIN and LP HE10 cases respectively. In addition, the proposed method improves coding efficiency by 0.1% when 8x4 / 4x8 bi-pred restriction is used. It is recommended to adopt the proposed method to P Slice. And if bi-pred restriction would be adopted in this meeting, it is also recommended to apply the proposed method when bi-pred restriction is used in B Slice.

# Reference

[1] Y. Zheng, X. Wang, W.-J. Chien, V. Seregin, and M. Karczewicz, “JCTVC-G683: Non-CE13: Simplification and improvement of additional merge candidate,” JCT-VC 7th Meeting, Geneva, 2011

[2] K. Kondo, T. Suzuki, T. Yamamoto, and T. Ikai, “JCTVC-I0107: AHG7: Modification of merge candidate derivation to reduce MC memory bandwidth,” JCT-VC 9th Meeting, Geneva, 2012

# Patent rights declaration(s)

**SHARP Corporation may have current or pending patent rights relating to the technology described in this contribution and, conditioned on reciprocity, is prepared to grant licenses under reasonable and non-discriminatory terms as necessary for implementation of the resulting ITU-T Recommendation | ISO/IEC International Standard (per box 2 of the ITU-T/ITU-R/ISO/IEC patent statement and licensing declaration form).**