Hard-Decision Quantization Algorithm Based on Deep Learning in Intra Video Coding

Hongkui Wang, Shengju Yu, Ying Zhang, and Zhuo Kuang, Li Yu*
School of Electron. Inf. & Commun., Huazhong Univ. of Sci. & Tech.
Wuhan, 430074, P.R.China
{hkwang, shengju_yu, zhangying9512, kuangzhuo, hustlyu}@hust.edu.cn

***Abstract**:* In video encoder, hard-decision quantization (HDQ) is well-suited for parallel processing, but suffers from non-negligible coding performance degradation compared with soft-decision quantization (SDQ). In this paper, by fully simulating the behavior of SDQ, a coefficient-adaptive offset model constructed by the deep learning approach is proposed to adjust the output of HDQ. Experiment results show that the proposed algorithm achieves promising RD performance and well-suited for hardware encoder implementation design.

Traditional HDQ with fixed-offset is widely adopted in various video encoders because of its lower complexity. Relatively speaking, SDQ outperforms the HDQ in RD performance by employing the inter-coefficient correction fully. In SDQ, the rate distortion optimization occurs in different levels and the output of SDQ is the result of HDQ fine-tuning. Thus, the adaptive offset model is built as a classification model of the quantized remainder (QRC model) with a fully connected network. The factors come from frame-level, transform block (TB) level, coefficient group level and coefficient level are used to train the QRC model. For different size of TB (i.e., 4×4~32×32), four QRC models are trained and applied in DL-HDQ algorithm.

As shown in Figure 1, DL-HDQ can be divided into three stages. Coefficients in TB are quantized by HDQ (offset, f=0), and the integer part (Φ) and decimal part (Ψ) of the TB are obtained. At meanwhile, all levels factors (Ω) for each remainder are collected. Following, according to the TB size, the decimal part and factors are entered into the corresponding QRC model and get the results of remainder classification. At last, the sum of integer part and QRC results is deemed to the output of DL-HDQ. Experiment results show that in comparison with HDQ, DL-HDQ achieves 2.17% bitrate saving on average while remaining friendly to hardware implementation; in comparison with SDQ, DL-HDQ exhibits 1.11% bitrate increment with 42% quantization time decrease on average.

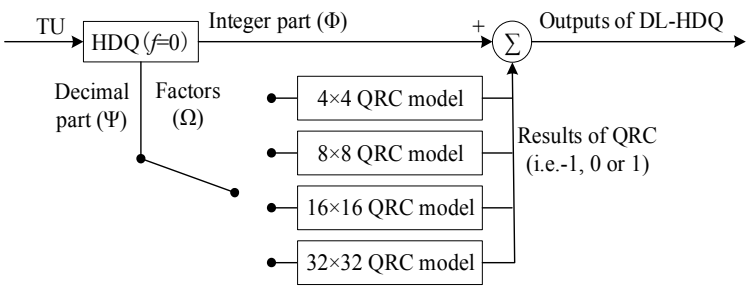

Figure 1: The proposed DL-HDQ algorithm.

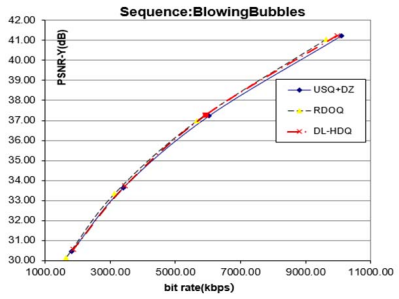

Figure 2: The RD performance comparisons

ACKNOWLEDGMENT

This work was supported by the NSFC (No. 61871437 and No. 61702205).

2375-0359/19/$31.00 ©2019 IEEE
DOI 10.1109/DCC.2019.00119