Fast H.264/AVC to HEVC transcoding using Mode Conditional Probability Models
Akshay Nagaraghatta, Sampath Kannangara, Yafan Zhao, and Grant Maxwell
School of Engineering, Robert Gordon University, Aberdeen, UK

Abstract—A fast H.264/AVC to HEVC transcoding algorithm is proposed. The inter prediction mode of HEVC Prediction Unit (PU) is modeled as a mode conditional probability function of the original H.264 macroblock mode, H.264 quantization parameter and the HEVC quantization parameter. Results show up to 50 percent complexity reduction with negligible rate distortion loss.

Keywords—HEVC, H.264/AVC, Video Transcoding

I. INTRODUCTION

High Efficiency Video Coding (HEVC) is the latest video coding standard [1][2], targeted for all existing video applications and the future beyond-HD high resolution formats and applications. HEVC provides better Rate-Distortion (RD) performance compared to previous H.264/AVC standard [3], however, at 2 to 10 times higher computational complexity [4][5]. With adaptation of HEVC standard, there is a growing need to transcode existing H.264 coded video into HEVC format in order to: (i) support legacy video content and (ii) reduce the storage requirements of the encoded content.

A number of algorithms have been proposed for H.264/AVC to HEVC transcoding. In [6], intra coded H.264/AVC blocks are merged to form larger HEVC Coding Units (CUs) if the direction of intra prediction is the same. In [7], dynamic thresholding is used to decide the HEVC prediction modes for 64x64 and 32x32 CUs, whereas, H.264/AVC modes are directly used for 16x16 and 8x8 CUs. In [8], an algorithm is proposed for MPEG-2 to HEVC transcoding where the MPEG2 mode information is mapped to the splitting depth of the HEVC CUs.

The existing algorithms reuse the original mode decisions for predicting the HEVC prediction modes. However, experiments show that final HEVC mode probabilities depend on quantization parameters as well as the original coding mode. Therefore, a new algorithm is proposed to derive the HEVC inter prediction modes using mode conditional probability functions based on H.264 coding mode, H.264 QP (Quantization Parameter) and the HEVC QP.

II. PROPOSED ALGORITHM

A. Mode Conditional Probabilities

An investigation was carried out to evaluate the effect of H.264/AVC and HEVC QPs in mapping H.264/AVC modes to HEVC coding modes during transcoding. Raw videos were encoded with H.264 and HEVC QPs of [20,26,32,38] resulting in 16 transcoded versions of each original video sequence.

Fig. 1. (i) the probability of a 8x8 block in H.264/AVC being coded as a 8x8 block in HEVC (ii) The number of 8x8 blocks in H.264/AVC for different QPs of {20,26,32,38} for 50 frames of ParkScene1920x1080 sequence

Using the probability data, a mathematical model is derived as a set of two one-dimensional third order polynomial equations to model the probability estimations. The model can be represented as:

\[ P_{\text{ed}} = \sum_{i=0}^{3} a_i x^i \]  

where \( a_1 = \sum_{j} k_j y_j, a_2 = \sum_{j} l_j y_j, a_3 = \sum_{j} m_j y_j, a_4 = \sum_{j} n_j y_j, x = \text{normalized value of QP}_{\text{HEVC}} \)
y = normalized value of QP_{AVC}, c = 1 to 5 and d = 1 to 5 (from Table 1).

The probability of a particular output mode (P_{\text{out}}) is modeled as a function of HEVC QP. Then the four constants, \(a_1, a_2, a_3\) and \(a_4\) are modeled as a function of H.264/AVC QP to obtain values of \(k_i, l_i, m_i\) and \(n_i\) for \(i = 0\) to 3.

C. Algorithm

In the HEVC encoder, the successive coding depths are traversed starting from 64 x 64 CU size. The AMP (Asymmetric Motion Partitioning) modes are not used as the computational complexity is very high for the limited RD gain obtained [8].

1. For every 16x16 prediction unit, the most probable HEVC mode is obtained from the mode conditional probability model corresponding to the AVC macroblock mode and the QP values.
2. Motion estimation is carried out for the most probable mode and the RD cost is compared with higher modes (such as 32x32 and 64x64). The mode with lowest RD cost and the corresponding CTU structure is selected.

III. EXPERIMENTAL RESULTS

The proposed algorithm was developed using the H.264/AVC reference codec, JM18.5 [9] and the HEVC reference codec HM12.0 [10]. The H.264/AVC and HEVC QPs of \{20,26,32,38\} were used. An IPPP GOP structure with one reference frame is used for both H.264 and HEVC encodings. The HEVC encoder is used with fast motion estimation mode and fast mode decision enabled. Four standard 1080p video sequences with varying motion and detail (Parkscene, Bluesky, Ducks and OldtownCross) are used for training.

The following terminologies are used: 
- **Prop:** Proposed mode conditional probability transcoder
- **BL:** Baseline HEVC encoder
- **DM:** Direct Mode Mapping transcoder

The RD performance is measured in terms of BD PSNR [11]. The direct mode mapping gives around 1% additional complexity savings.

### Table II. RD Performance and Complexity Savings of the Proposed Algorithm (on Intel Xeon, 3.5GHz, 32GB RAM, H.264 QP=20)

<table>
<thead>
<tr>
<th>Training</th>
<th>BL vs Prop (dB)</th>
<th>BL vs DM (dB)</th>
<th>Avg Saving (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parkscene 1920x1080</td>
<td>-0.1211</td>
<td>-0.1651</td>
<td>43</td>
</tr>
<tr>
<td>Bluesky 1920x1080</td>
<td>-0.1348</td>
<td>-0.1555</td>
<td>45</td>
</tr>
<tr>
<td>Ducks 1920x1080</td>
<td>-0.0239</td>
<td>-0.0634</td>
<td>41</td>
</tr>
<tr>
<td>Oldtown 1920x1080</td>
<td>-0.0476</td>
<td>-0.0992</td>
<td>44</td>
</tr>
</tbody>
</table>

### IV. Conclusion

In this paper, a fast H.264 to HEVC transcoding algorithm is proposed. The algorithm achieved around 35-50% performance gain in complexity with a very negligible loss in rate-distortion performance. The main contribution of this work is the conditional probability model that can be used to predict the most probable HEVC inter prediction coding mode for a given H.264 coding mode, H.264 QP and HEVC QP. Future work will involve extending the scope of the algorithm to cover smaller block sizes and to merge H.264 modes to form bigger block sizes in HEVC.

### References


