1. INTRODUCTION

Fast search motion estimation (ME) algorithms are extremely important to reduce the complexity of high efficiency video coding (HEVC) encoding.

Massively parallel architectures, such as GPUs, provide a promising computing platform to calculate the best motion vector (MV) of several blocks in parallel in order to achieve fast encoding.

2. MULTIPLE TEMPORAL PREDICTORS (MTP)

- Perform RCME in two steps:
  1. Use a list of MVP candidates, RCME data is calculated in parallel in the GPU.
  2. Best MV and MVP is selected when actual MV is available in RDO thread of GPU.

- RCME for all possible PUs of a CTU (425 possible PUs) are scheduled at the same time into the execution queue.

- Threads of a warp are being executed efficiently since there is no diverged execution path and all of 64 threads of each PU follow the same execution path.

- For a CTU of size 64x64, there are 16 temporal MVs already stored in the encoder’s picture buffer and there is no significant overhead in terms of memory.

- GPU load is reduced from 86% for full search to 52% for our proposed NDS method.

3. NESTED DIAMOND SEARCH (NDS)

- Smallest execution unit in GPU is a wavefront/warp that contains 64 parallel thread executing the same instruction.

- Designed for GPU architecture, we define a nested fixed diamond pattern consisting of 64 positions.

- RCME is performed in several iteration by finding best MVMs of our fixed pattern.

4. EXPERIMENTAL RESULTS

- Methodology
  1. Software: Implementation in the HEVC test model HM15.0
  2. Hardware: Intel(R) Xeon(R) CPU E5-2670 @ 2.60GHz, equipped with an AMD Radeon R9-270 GPU.

- Encoder is set to “Low-delay P” configuration and quantization parameters (QPs) of 22, 27, 32, and 37.

5. CONCLUSIONS

- Proposed method achieves 41% time saving with 0.9% BD-rate increase compared to fast search method (T2S).

- GPU load is reduced from 86% for full search to 52% for our proposed NDS method.