Fast H.264 to HEVC Transcoder Based on Post-Order Traversal of Quadtree Structure

Jean-François Franche[1] Stéphane Coulombe[1]

[1] École de technologie supérieure, Université du Québec, Canada

International Conference on Image Processing (ICIP), 2015

Outline

What is H.264, HEVC and video transcoding?

H.264 (2003)

- A popular video coding format for various applications

HEVC (2013)

- Successor to H.264
- Halved the bit-rate for similar video quality compared to H.264

Video transcoding

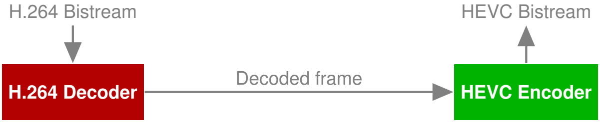
- Converts a video stream from one format to another

Why we need to transcode H.264 streams to HEVC ?

- Reduce the bit-rate
- Assure interoperability between systems

The Cascaded Pixel-Domain Transcoding (CPDT) Architecture

The full transcoding architecture decodes the H.264 input in pixel domain and re-encodes the data in the HEVC format.



Advantages

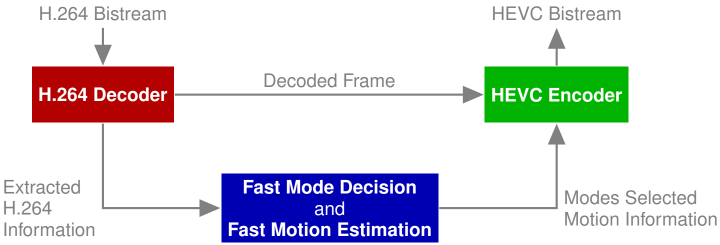
+ Easy implementation
+ Flexible
+ High coding efficiency

Drawback

– Very complex computationally

Transcoding Architecture Based on Reuse of Extracted Information from H.264

To reduces the transcoding complexity many works reuse extracted information from H.264 to simplify complex encoding tasks



Transcoding Architecture Based on Reuse of Extracted Information from H.264

Transcoder must adapt the extracted information from H.264 to HEVC coding tools

Table 1: Relevant differences between H.264 and HEVC coding tools

Tool	**H.264**	**HEVC**
Basic Unit Size	16×16 (MB)	64×64* (CTU)
Inter Partitioning	From 16×16 to 4×4	From 64×64 to 8×4 (4×8)
Motion Prediction	Median predictor	AMVP (2 candidates)
Motion Copy	Skip (1 candidate)	Skip/merge (5 candidates)
Intra Partitioning	16×16, 4×4	From 32×32 to 4×4
Intra Prediction	Up to 9 predictors	Up to 35 predictors

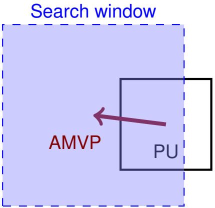
*CTU size is configurable from 8×8 to 64×64. In literature, 64×64 is a common size.

In pratice, the transcoder must achieves a trade-off between the computational complexity reduction and the coding efficiency

Motion Vector Refinement

Motion search in video encoding

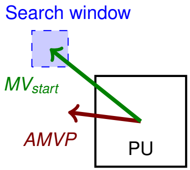
- Motion is unknown
- Requires a large search window



Motion search in video transcoding

- Refines motion from H.264
- Requires a smaller search window

+ Less integer positions evaluated
– Unchanged sub-pixel refinement
– Redundancy in prediction errors computation

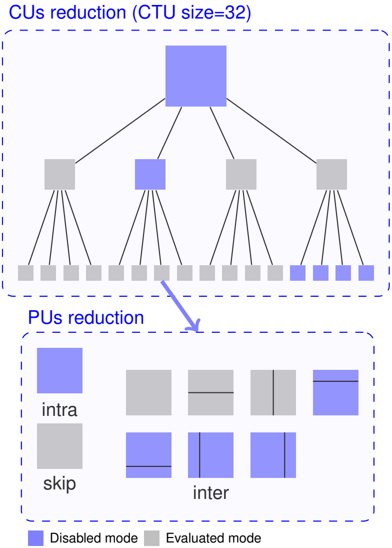


Mode Reduction Based on H.264 Information

- Several approaches create a mapping between H.264 information and HEVC modes [1–10].
  - Fang et al. use H.264 modes, residuals and variances of motion vectors to disable some prediction unit (PU) modes [1]

\+ Reduce a reasonable number of modes.

– Used alone, the speed-up is limited to 2-3x

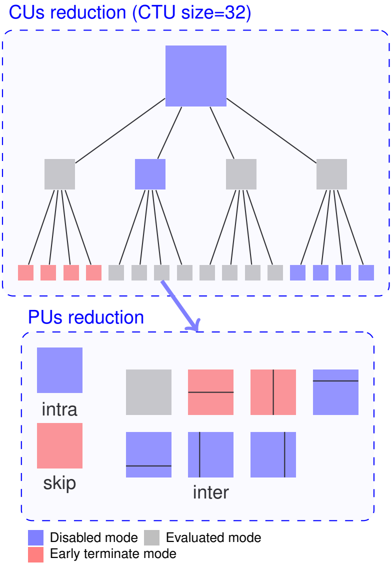


Mode Reduction Based on HEVC Information

- In addition to H.264 information, some works consider HEVC information to reduce more modes.
  - Peixoto et al. propose an early termination method based on rate-distortion (RD) cost [3]

+ Reduce more modes and achieves greater speed-ups

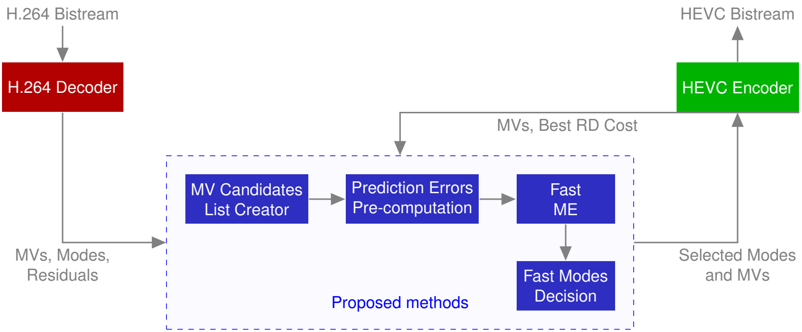
– The most promising modes are not firstly processed
  - The CTU quadtree is still processed with a pre-order traversal



Architecture of the proposed transcoder

Our approach is composed of two main contributions:

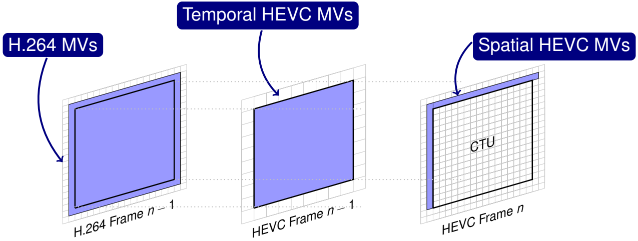
1. A two stages motion estimation based on motion propagation
2. A fast mode decision framework based on post-order traversal of the CTU quadtree structure



Two Stages Motion Estimation Algorithm

Stage 1 - Operate at the CTU level

- Create a motion vector (MV) candidates list
  - Composed of H.264 MVs, HEVC MVs and MV (0,0)
- Compute the prediction errors for each candidate
  - Interpolate the predicted CTU region
  - Divide the CTU region in 4×4 blocks
  - Compute and store the sum of absolute transformed differences (SATD) for each 4×4 block



Two Stages Motion Estimation Algorithm

Stage 2 - Operate at the PU level

- Compute the prediction errors for each candidate
  - Sum the SATD (computed in stage 1) of the 4×4 blocks covering the PU region
- Select the best combination of MV and AMVP predictors
  - Combination that minimize:

$$J_{\mathrm{PM}} = \mathrm{SATD}(R) + \lambda_{pred} \times \mathrm{bits}(M) \qquad (1)$$

R is the difference between the predicted and original blocks, λ_{pred} depends on QP and M is the motion information.

Advantages

+ Eliminates computational redundancy
  - Prediction errors are computed only once
+ Very low complexity at PU level
  - No SATD, no interpolation, no motion refinement

Early Termination of the RD Cost Computation

Problem

- Determining motion parameters for a PU is a low-complexity task
- However, determining the RD cost J_{RD} is still a very complex task:

$$J_{\mathrm{RD}} = (\mathrm{SSE}_{\mathrm{Y}}(\widetilde{R}) + 0.57\,\mathrm{SSE}_{\mathrm{CbCr}}(\widetilde{R})) + \lambda_{\mathrm{mode}} \times \mathrm{bits}\,(\mathrm{mode}), \quad (2)$$

$\widetilde{R}$ is the difference between the original and reconstructed blocks, $\lambda_{\mathrm{mode}} = \lambda_{pred}^2$ and bits (mode) is the number of bits to encode the mode.

Solution

- The RD cost J_{RD} is highly correlated with the SATD cost J_{PM}
- We developed an early termination criterion that ends the RD cost computation when:

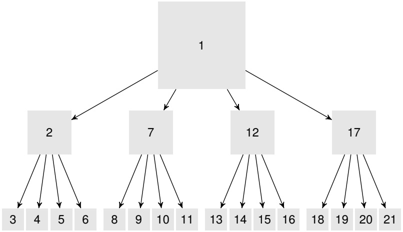
$$J_{PM}^{current} > (J_{PM}^{best} + T), \text{where } T \geq 0 \quad (3)$$

* In our simulations, the threshold T was set to : $3\lambda_{\mathrm{pred}}$. This value offers a excellent trade-off between speed-ups and coding efficiency

Post-Order Traversal of the CTU quadtree structure

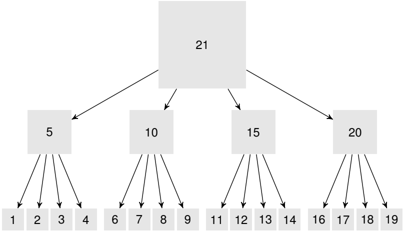
Problem with the pre-order traversal

- The early termination criterion can only be applied when the compared modes have the same size.
- In the pre-order traversal, it impossible to apply this criterion on a CU and a sub-CU



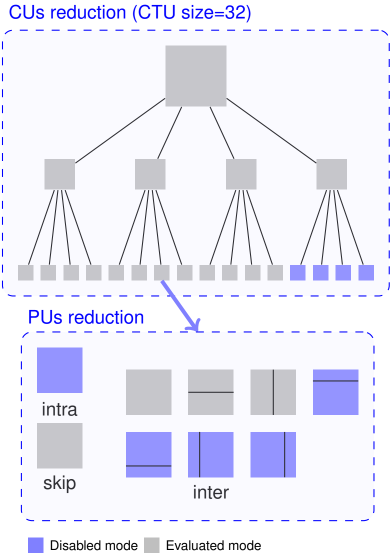
Solution: post-order traversal

- The post-order traversal allows to apply the early termination criterion on the combination of 4 sub-CUs and the CU parent



Mode Reduction based on H.264 information

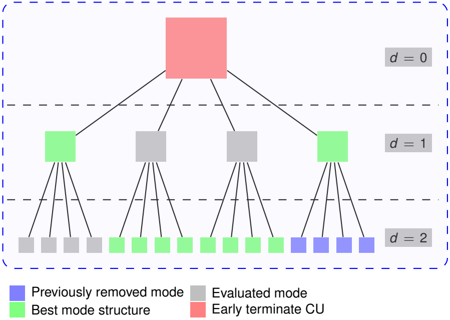
- HEVC partitions finer than H.264 partitions are disabled
- Intra modes are disabled when the H.264 region contains no intra.
- Inter modes (excepts split/merge modes) are disabled when H.264 region contains no inter.
- AMP modes are disabled since they are time-consuming and have low impact on coding efficiency



Mode Reduction based on HEVC information

- The idea
  - If the current best HEVC mode have a complex partitioning structure, a really more simple partitioning structure is unpromising
- The heuristic
  - When all descendants of the current CU of depth d are processed, the CU is early-terminated if the best mode contains a CU deeper than $d + 1$

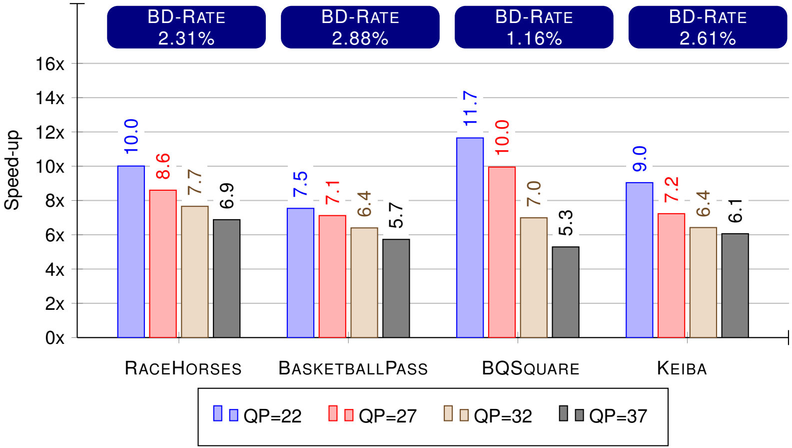
CUs reduction (CTU size=32)



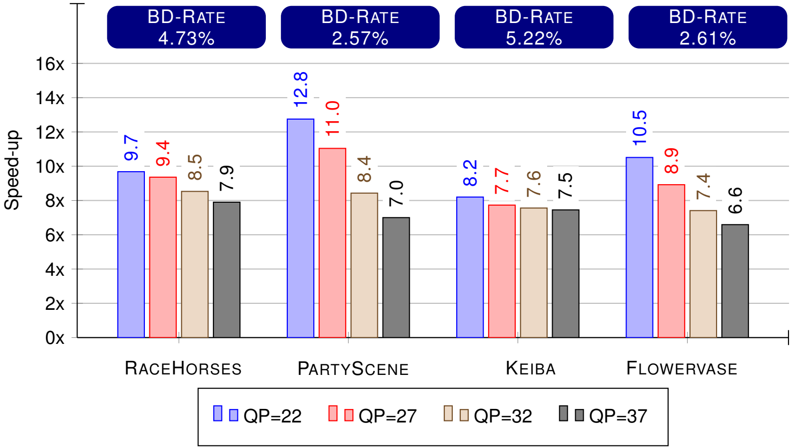
Validation Methodology

H.264 Encoder	JM 18.2, baseline, fast full search
HEVC Encoder	HM 12.1, low-delay, fast search
Configuration	H.264/HEVC common configuration ▸ Coding structure : IPPP ▸ QPs : 22, 27, 32, 37 ▸ Reference frames : 1 ▸ Search range : [-64,64]
Anchor	Full transcoding (CPDT)
Measures	Transcoding performance measures ▸ Bjontegaard Delta-Rate (BD-Rate) ▸ Speed-up based on HEVC encoding time

Performances for 416×240 sequences

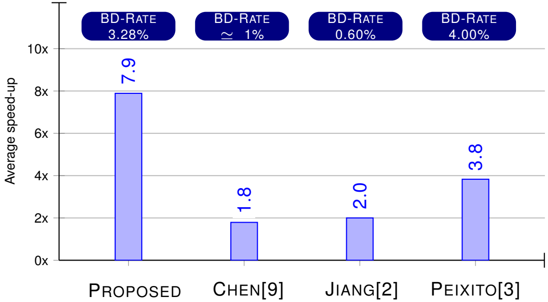


Performances for 832×480 sequences



Comparison with related works

- Compared to the state of the art approaches:
  - Greater speed-up
  - Better trade-off between speed-up and BD-Rate
- Experimental set-ups are different
  - Different configurations, parameters, sequences, HM version, etc.



Conclusion

Contributions

- Two stages motion estimation approach
  - No motion refinement, low complexity at PU level
- Fast mode decision framework based on post-order traversal of the CTU
  - Criterion to early terminate the RD cost computation
  - Mode reduction techniques
  - Exploits information from H.264 and HEVC

Future works

- Improve mode reduction techniques

Thanks for your attention! Any questions?

This research was funded by:

- The National Science And Engineering Research Council
- Vantrix Corporation

Reference

[1] J.T. Fang, Z.Y. Chen, T.L. Liao, and P.C. Chang, "A fast PU mode decision algorithm for H.264/AVC to HEVC transcoding," *Computer Science*, 2014.

[2] W. Jiang, Y. Chen, and X. Tian, "Fast transcoding from H.264 to HEVC based on region feature analysis," *Multimedia Tools and Applications*, pp. 1–22, 2013.

[3] E. Peixoto, B. Macchiavello, E.M. Hung, and R.L. de Queiroz, "A fast HEVC transcoder based on content modeling and early termination," *21st IEEE International Conference on Image Processing (ICIP 2014)*, pp. 1–5, 2014.

[4] D. Zhang, B. Li, J. Xu, and H. Li, "Fast transcoding from H.264/AVC to high efficiency video coding," in *Multimedia and Expo (ICME), 2012 IEEE International Conference on*. IEEE, 2012, pp. 651–656.

[5] W. Jiang and Y.W. Chen, "Low-complexity transcoding from H.264 to HEVC based on motion vector clustering," *Electronics Letters*, vol. 49, no. 19, pp. 1224–1226, 2013.

[6] C. Zong-Yi, Tseng C.T., and C. Pao-Chi, "Fast inter prediction for H.264 to HEVC transcoding," in *3rd International Conference on Multimedia Technology (ICMT-13)*. Atlantis Press, 2013.

[7] R. Luo, R. Xie, and L. Zhang, "Fast AVS to HEVC transcoding based on ROI detection using visual characteristics," in *Broadband Multimedia Systems and Broadcasting (BMSB), 2014 IEEE International Symposium on*. IEEE, 2014, pp. 1–6.

[8] T. Shen, Y. Lu, Z. Wen, L. Zou, Y. Chen, and J. Wen, "Ultra fast H.264/AVC to HEVC transcoder," in *Data Compression Conference (DCC), 2013*. IEEE, 2013, pp. 241–250.

[9] Z.-Y. Chen, J.-T. Fang, T.-L. Liao, and P.-C. Chang, "Efficient PU mode decision and motion estimation for H.264/AVC to HEVC transcoder," *Signal*, 2014.

[10] P. Xing, Y. Tian, X. Zhang, Y. Wang, and T. Huang, "A coding unit classification based AVC-to-HEVC transcoding with background modeling for surveillance videos," in *Visual Communications and Image Processing (VCIP), 2013*. IEEE, 2013, pp. 1–6.