# COPY MODE FOR STATIC SCREEN CONTENT CODING WITH HEVC

*Thorsten Laude and Jörn Ostermann*

Institut für Informationsverarbeitung (TNT), Leibniz Universität Hannover, Germany

## ABSTRACT

Screen content largely consists of static parts, e.g. static background. However, none of the available screen content coding tools fully employs this characteristic. In this paper we present the copy mode, a new coding mode specifically aiming at increased coding efficiency for static screen content. The basic principle of the copy mode is the direct copy of the collocated block from the reference frame. Mean weighted BD-Rate gains of 2.4% are achieved for JCT-VC test sequences compared to SCM-2.0. For sequences containing lots of static background, coding gains as high as 7.6% are observed. The new coding mode is further enhanced by several encoder optimizations, among them an early skip mechanism. Thereby, the encoder runtime complexity is reduced by up to 39%.

***Index Terms***— video coding, HEVC, screen content coding

## 1. INTRODUCTION

The recent years have seen an enormous change in the use of computer technologies. With the progressive replacement of traditional (laptop or desktop) computers by mobile devices (tablets and smartphones) new application scenarios such as remote computing or wireless displays have appeared. The commonality of these scenarios is the separation of computer program execution and program output display to different devices. In the case of remote computing, the computer program may be executed in the cloud while end user devices only provide the program output display. For wireless displays, the screen of one device, e.g. of a tablet, is mirrored to another device, e.g. a television set. Apple AirPlay and Miracast are examples in this context. These scenarios are accompanied by the necessity of transmitting computer generated video signals between the involved devices. Furthermore, considering that the signals might be transmitted over channels with limited transmission capacity (e.g. Internet connections), efficient coding of the transmitted signals is required. The coding of computer generated video signals is often referred to as screen content coding (SCC). Since compatibility between different devices is imperative, it is advisable to use standardized methods for SCC. Video coding standards like MPEG-2 [1] or AVC [2] have been studied extensively during the last decades and are well known to provide the desired compatibility. In January 2013, the Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T VCEG and ISO/IEC MPEG has finished the technical work for the latest video coding standard, High Efficiency Video Coding (HEVC) [3]. It achieves the same visual quality with half the bit rate compared to the predecessor standard AVC [4].

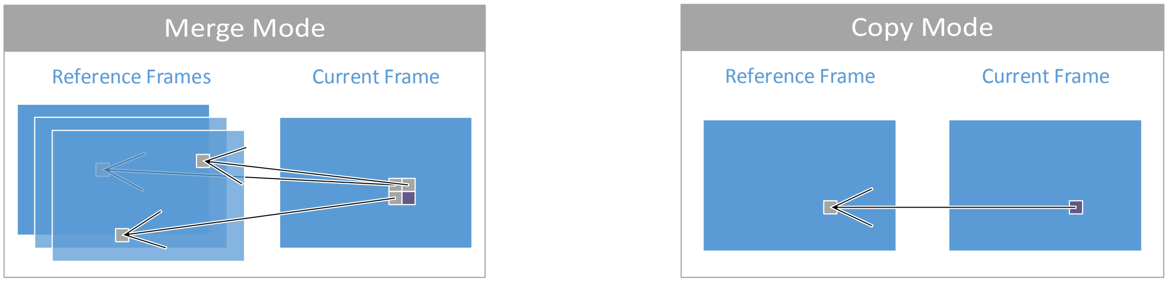
Even though this state-of-the-art video coding standard provides superior coding efficiency for camera captured videos, it is not the optimal coding method for screen content (SC). The reason therefor is that HEVC (as well as the predecessor standards) has not been developed with careful consideration of screen content signals. Thus, after finalizing HEVC, the JCT-VC started the development of a screen content coding extension to HEVC, often referred to as HEVC SCC, in 2014 [5]. This extension, which is planned to be finalized in 2015 or 2016, brings new coding tools addressing several key characteristics of screen content (small number of different colors, recurrent patterns, RGB source material, no noise, etc.).

However, none of these additional SCC tools aims at static screen content. Therefore, taking into account that static parts (e.g. background) are very common in screen content signals, a new coding tool for static screen content coding is proposed in this work. Our contributions are this new coding mode which we refer to as *copy mode* along with several encoder optimization mechanisms (which are crucial for real time screen content applications) assisting the copy mode.

The remainder of the paper is organized as follows: In Section 2 we analyze the related work, introduce the new coding mode and describe the encoder optimizations. The experimental results are evaluated in Section 3. Section 4 gives a conclusion of the paper.

## 2. STATIC SCREEN CONTENT CODING

Typical screen content signals consist in large parts of static areas, i.e. areas without changes between consecutive frames. That is why it is desirable to use a specific coding mode for static screen content in order to employ this characteristic. In the following parts of this section, already known screen content coding methods and their lack of coding efficiency for static screen content are analyzed. After that the copy mode for static screen content and various encoder optimizations are introduced.




(a) (b)

**Fig. 1**. Comparison of the merge mode (a) with the copy mode (b): While the merge mode selects the motion information from several candidates (with the necessity of signaling the selection) the copy mode has only one candidate (the optimal candidate for static screen content, i.e. the zero motion vector), and thus avoids this signaling.

### 2.1. Related work

Numerous methods for screen content coding are known in the literature [6], [7], [8]. Among them there are the palette mode, intra block copy and adaptive color space transformations. These three methods are the most notable innovations of the current HEVC SCC draft [6] and address specific screen content properties. While the palette mode is suitable to code signal parts with a small number of potentially very different colors and sharp edges [7], [8], the intra block copy mode is capable of describing recurrent patterns within one frame (e.g. text characters in longer texts). Assuming that typical screen content signals are captured in the RGB color space, the adaptive color space transformation can be used to transform the signal to a color space where the signal components are less correlated, e.g. to the YCbCr or to the YCgCo color space. However, none of these tools specifically addresses static screen content. The merge mode [9], which is part of the regular HEVC standard, is beneficial for the coding of areas with homogeneous motion. Furthermore, static areas, i.e. areas with no motion, can be considered as a special case of homogeneous motion. For this reason, the merge mode (or more specifically the skip mode, i.e. the merge mode without residual) is most likely to be chosen by encoders for static screen content. As illustrated in Fig. 1(a), the basic principle of the merge mode is the motion information copy for the current block (purple) from one of several merge candidates (gray). These merge candidates are spatial or temporal adjacent blocks. This principle leads to several drawbacks with respect to static screen content coding: Firstly, the optimal motion information for static screen content, i.e. the zero motion vector, is not necessarily among the merge candidates. Secondly, taking into account the low bit rates for screen content, the signaling of the selected merge candidate is costly. Additionally, in case the motion vector of the selected merge candidate has sub-pel precision, interpolation for the reconstruction process might introduce unnecessary complexity.

The fundamental idea of the introduced copy mode is similar in spirit to the well-known conditional replenishment technique which has been introduced by Mounts in 1969 [10] (who signals the replenishment for each pel) and further enhanced by Jones and Hein (who use a separate change indicator map for each $8 \times 8$ block) in subsequent decades [11], [12]. However, while conditional replenishment is based on the omission of static frame parts during encoding, we incorporate a distinct coding mode into the coder to fully enable rate-distortion optimization.

### 2.2. Copy mode

The new copy mode exploits that the uncompressed sample values of corresponding blocks, i.e. of blocks located at the same spatial position in consecutive frames, are identical for static screen content. Therefore, the copy mode prediction of the block (which in case of HEVC is referred to as coding unit or CU) in the current frame is formulated as sample value copy from the block in a reference frame. Fig. 1(b) illustrates this prediction process.

Commonly, several reference frames are available for prediction. To avoid signaling overhead, the reference frame for the copy mode prediction is selected implicitly at the encoder and at the decoder without any signaled information. In general, it is desirable that the selected reference frame has both a high fidelity and a small temporal distance to the current frame. As part of our research two selection methods have been investigated: The selection of the reference frame with the smallest picture order count (POC) difference (i.e. with the smallest temporal distance) and the selection of the reference frame with the smallest quantization parameter (QP) value, i.e. with the highest fidelity. However, if the selection is QP driven, it is hardly probable (without major decoder modifications) that the content is static due to the increased temporal distance. Thus, the first option is chosen for this paper.

A binary flag at the very beginning of the coding unit syntax (before the skip flag) is used to signal the copy mode usage. If this flag is equal to 1, all of the remaining syntax is skipped. Otherwise, the syntax for the other coding modes like skip, merge, inter, intra, etc. is signaled. Considering the low overall bit rates achieved for screen content signals, it is indispensable to avoid any unnecessary signaling overhead. Based on empirical results it has been observed that static

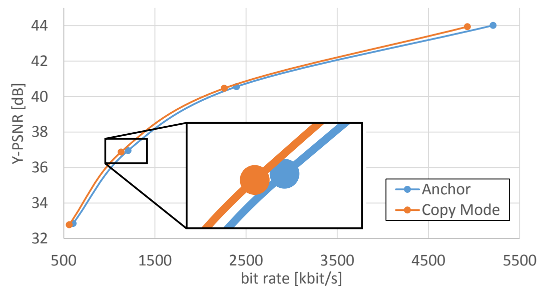



**Fig. 2.** RD curve for the sequence *Basketball Screen RGB*. The new copy mode coding tools achieve lower bit rates for all Y-PSNR values.

parts of screen content signals are typically partitioned into large coding units while the remaining parts which change over time are encoded using smaller coding units. Therefore, in order to achieve the goal of small signaling overhead, the binary copy mode flag is only signaled for the largest coding units (size $64 \times 64$), which are called coding tree units (CTU) in the context of HEVC.

In comparison to the merge mode, following Fig. 1, the copy mode has several advantages: The optimal motion vector candidate for static screen content, i.e. the zero motion vector, is always used for the prediction. Since only one candidate is available, less syntax elements are required to signal the prediction (only one flag for the mode usage itself, no additional syntax to signal the selected candidate). Finally, no interpolation is required to reconstruct the block since the zero motion vector has full-pel precision.

### 2.3. Encoder optimization

The presented copy mode is assisted by several encoder optimization mechanisms. Taking into account that the copy mode is only beneficial if static content is coded, it is desirable to enable and disable the copy mode based on the presence of static content. For this purpose, the copy mode is enabled on frame level. By this mean it is possible to enable the copy mode if the current frame mostly consists of static content. Furthermore, if no or only little static content exists in the current frame, the copy mode is disabled to avoid unnecessary signaling overhead for a coding mode which is not beneficial for the content. This is necessary, not least because the copy mode flag is signaled at the very start of the CU syntax. Hence, the signaling is inevitable, independent of the used coding mode. Since the copy mode is applied on CTU level, a preprocessing step is executed to determine the ratio of static CTUs within the current frame. If this ratio exceeds a threshold $\alpha$, as formulated in Equation 1, the copy mode is enabled for the frame:

$$n_{\text{static}}/n_{\text{total}} \geq \alpha. \quad (1)$$

$n_{\text{static}}$ and $n_{\text{total}}$ denote the number of static CTUs and the total number of CTUs in the current frame, respectively. Exhaustive experiments suggest that a value of $0.6$ for $\alpha$ results in the highest coding efficiency. Thus, this value is adopted for our implementation. A binary flag is signaled in the slice header to indicate the copy mode activation.

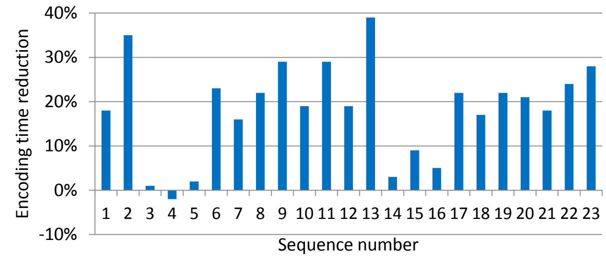



**Fig. 3.** Encoder runtime reductions: By means of the early skip mechanism the complexity is drastically reduced. For the sake of easy readability, the sequences are numbered according to their appearance in Table 1.

In contrast to many traditional video coding scenarios (e.g. movie distribution) where encoding time does not pose severe constraints to encoders, fast and low complexity encoders are crucial for typical real time screen content coding scenarios. For this reason, an early-skip procedure is applied if the copy mode usage is enabled for a frame. The premise of the early-skip procedure is that the copy mode provides a good prediction for static CTUs. Thus, if a CTU is static, the copy mode is the only mode which is checked in the rate distortion (RD) loop. All remaining coding modes and the splitting of the CTU into smaller blocks are omitted. Thereby, the encoding process can be drastically accelerated. It should be noted that this is not the best rate distortion optimization, especially considering that the reference frame might have a slightly higher quantization parameter (lower fidelity) than the current frame. However, since this error has a temporal noise free character and is thereby scarcely perceptible, this effect is neglected for the sake of fast encoding decisions.

## 3. EXPERIMENTAL RESULTS

In this section, the proposed coding tools are evaluated. For this purpose, the copy mode and the encoder optimizations were implemented into the HEVC SCC reference software SCM-2.0 [6].

An extensive set of 23 JCT-VC test sequences as listed in Table 1 is used for the evaluation. Both the RGB and the YCbCr versions of the sequences are utilized. Except for one 4:2:0 color sampling sequence (*Slide Editing*), all sequences have 4:4:4 color sampling. Since typical screen content scenarios are real time oriented, the low delay configuration as defined by the HEVC SCC common test conditions (CTC) [13] is applied. This is the only CTC encoder configuration which does not imply any structural temporal delay and is therefore suitable for real time scenarios. A set of four different QPs (27, 32, 37 and 42) is used for quantization, covering a wide range of different bit rates.

The Bjøntegaard-Delta (BD)-Rate as defined in [14] is

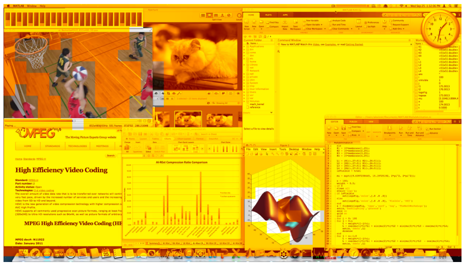


**Fig. 4**. Copy mode usage for the second frame of the sequence *Basketball Screen YCbCr*. Parts of the frame which are coded using the copy mode are colored in yellow. It can be observed that the copy mode is the predominant coding mode.

calculated to evaluate the coding efficiency of the proposed coding tools. Additionally, as suggested in [15], weighted average BD-Rates $\mathrm{BD}_{\mathrm{avg}}$ are calculated in accordance with Equation 2:

$$\mathrm{BD}_{\mathrm{avg}} = \begin{cases} \dfrac{4 \times \mathrm{BD}_{Y/G} + \mathrm{BD}_{Cb/B} + \mathrm{BD}_{Cr/G}}{6} & \text{for 4:4:4} \\ \dfrac{6 \times \mathrm{BD}_{Y/G} + \mathrm{BD}_{Cb/B} + \mathrm{BD}_{Cr/G}}{8} & \text{for 4:2:0.} \end{cases} \quad (2)$$

$\mathrm{BD}_c$ denotes the BD-Rate for the color component $c$ with different weighting factors depending on the color sampling format (4:4:4 or 4:2:0). Table 1 presents the resulting BD-Rates. The weighted average gains are in the range of $-0.5\%$ to $7.6\%$ with a mean value of $2.4\%$, indicating that the proposed coding tools considerably improve the coding efficiency for static screen content. For individual color components, gains as high as $12.2\%$ are observed. It is worth noting that minor losses are observed for the sequences *Mission Control 3 RGB* and *ppt doc xls YCbCr* due to the in some cases sub-optimal early skip mechanism. Furthermore, nine sequences (Robot, EBU Rainfruits, Kimono and Twist Tunnel, each sequence in RGB and in YCbCr as well as Basketball Drill Text in YCbCr 4:2:0) for which the copy mode is conceptually not beneficial (camera captured or no noteworthy static areas) were evaluated. The BD-Rates for all of these sequences are 0.0%, demonstrating that the frame level on/off switch avoids unnecessary signaling overhead. Figure 2 depicts the RD curve (bit rate on the horizontal axis, luminance peak-signal-to-noise-ratio (Y-PSNR) on the vertical axis) for the sequence *Basketball Screen RGB*. The coding efficiency is improved considerably across all bit rates.

A major aspect of our work is the introduction of the early skip mechanism. Since it aims at reduced encoding complexity, the encoding runtime reduction (ETR) as presented in Table 1 and in Figure 3 is analyzed to evaluate its performance. In the interest of measuring reliable runtimes, a homogeneous set of compute servers without the usage of hyperthreading capabilities was deployed for the evaluation. As it can be observed, an average encoder runtime reduction of 18% is achieved. For some sequences, e.g. *CAD Waveform RGB* and *CAD Waveform YCbCr*, the encoding time is drastically reduced with values rising up to 39%. Taking into account that

**Table 1**. BD-Rates: Negative numbers indicate increased coding efficiency. Weighting factors of 4/1/1 (4:4:4) and 6/1/1 (4:2:0) for the three color components are used to calculate the average results. The coding efficiency is noticeably improved by the new coding tools. Furthermore, high encoder runtime reductions (ETR) are observed.

| | | **RGB** | | | |
|---|---|---|---|---|---|
| | G | B | R | **Average** | ETR |
| 1. Basketball Screen | -4.6% | -4.6% | -4.8% | **-4.6%** | 18% |
| 2. CAD Waveform | -2.6% | -2.5% | -2.2% | **-2.5%** | 35% |
| 3. Console | -1.2% | -3.6% | -2.0% | **-1.7%** | 1% |
| 4. Desktop | -1.7% | -2.4% | -2.3% | **-1.9%** | -2% |
| 5. Flying Graphics | -0.5% | -0.5% | -0.6% | **-0.5%** | 2% |
| 6. Map | -1.1% | -3.4% | -3.7% | **-1.9%** | 23% |
| 7. Missioncontrol 3 | 1.1% | -2.1% | 0.4% | **0.5%** | 16% |
| 8. PCB Layout | -0.9% | -0.6% | -0.6% | **-0.8%** | 22% |
| 9. ppt doc xls | -0.2% | -0.1% | -0.1% | **-0.1%** | 29% |
| 10. Videoconferencing | -5.9% | -6.8% | -6.9% | **-6.2%** | 19% |
| 11. Web Browsing | -6.7% | -8.1% | -8.7% | **-7.2%** | 29% |
| Mean (RGB) | -2.2% | -3.2% | -2.9% | **-2.5%** | 17% |
| | | **YCbCr** | | | |
| | Y | Cb | Cr | **Average** | ETR |
| 12. Basketball Screen | -4.6% | -4.7% | -4.9% | **-4.7%** | 19% |
| 13. CAD Waveform | -3.2% | -3.2% | -2.9% | **-3.2%** | 39% |
| 14. Console | -1.9% | -2.9% | -0.9% | **-1.9%** | 3% |
| 15. Desktop | -1.9% | -2.3% | -2.3% | **-2.0%** | 9% |
| 16. Flying Graphics | -0.4% | -0.6% | -0.5% | **-0.4%** | 5% |
| 17. Map | 0.7% | -5.9% | -12.2% | **-2.5%** | 22% |
| 18. Missioncontrol 3 | -1.2% | -2.8% | -4.7% | **-2.1%** | 17% |
| 19. PCB Layout | -1.1% | -0.9% | -0.9% | **-1.0%** | 22% |
| 20. ppt doc xls | 0.4% | -0.5% | 0.1% | **0.2%** | 21% |
| 21. Slide Editing (4:2:0) | -0.7% | -0.4% | 0.1% | **-0.5%** | 18% |
| 22. Videoconferencing | -3.0% | -3.8% | -3.7% | **-3.2%** | 24% |
| 23. Web Browsing | -6.8% | -9.7% | -8.9% | **-7.6%** | 28% |
| Mean (YCbCr) | -2.0% | -3.1% | -3.5% | **-2.4%** | 19% |
| **Mean (overall)** | **-2.1%** | **-3.2%** | **-3.2%** | **-2.4%** | **18%** |

the early skip mode enforces the usage of the copy mode, it is expected that the copy mode is the predominant coding mode for sequences with lots of static content. This expectation is confirmed by Figure 4, which illustrates the copy mode usage for the second frame of the *Basketball Screen YCbCr* sequence. In this representative figure, it is demonstrated that the major part of the frame is coded with the copy mode.

## 4. CONCLUSIONS

In this paper, we present a new coding mode (the *copy mode* which is based upon the direct copy of the collocated block from the reference frame) for the static screen content coding with HEVC as well as several encoder optimizations specifically addressing this kind of content. These new coding tools are implemented into the HEVC SCC reference software SCM-2.0. Weighted average BD-Rate coding gains up to 7.6% for JCT-VC test sequences are demonstrated. These gains could be further increased by advanced reference selection algorithms with increased decoder complexity. Encoding time reductions up to 39% are measured.

## 5. REFERENCES


[1] ISO/IEC 138182, "Generic coding of moving pictures and associated audio informationPart 2: Video/ITU-T Recommendation H.262," 1994.

[2] ISO/IEC 1449610, "Coding of Audiovisual Objects-Part 10: Advanced Video Coding/ITU-T Recommendation H.264 Advanced video coding for generic audiovisual services," 2003.

[3] "ITU-T Recommendation H.265/ ISO/IEC 23008-2:2013 MPEG-H Part 2: High Efficiency Video Coding (HEVC)," 2013.

[4] Philippe Hanhart, Martin Rerabek, Francesca De Simone, and Touradj Ebrahimi, "Subjective quality evaluation of the upcoming HEVC video compression standard," in *SPIE Optical Engineering + Applications*, Andrew G. Tescher, Ed. Oct. 2012, p. 84990V, International Society for Optics and Photonics.

[5] ITU-T SG16/Q6 and ISO/IEC JTC 1/SC 29/WG 11, "N14175: Joint Call for Proposals for Coding of Screen Content," 2014.

[6] R. Joshi and J. Xu, "JCT-VC S1005: High Efficiency Video Coding (HEVC) Screen Content Coding: Draft 2. 19th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Strasbourgh, FR," 2014.

[7] Zhaotai Pan, Huifeng Shen, Yan Lu, Shipeng Li, and Nenghai Yu, "A Low-Complexity Screen Compression Scheme for Interactive Screen Sharing," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 6, pp. 949–960, June 2013.

[8] Liwei Guo, Wei Pu, Feng Zou, Joel Sole, Marta Karczewicz, and Rajan Joshi, "Color Palette for Screen Content Coding," in *IEEE International Conference on Image Processing (ICIP)*, 2014, pp. 5556–5560.

[9] Philipp Helle, Simon Oudin, Benjamin Bross, Detlev Marpe, M. Oguz Bici, Kemal Ugur, Joel Jung, Gordon Clare, and Thomas Wiegand, "Block Merging for Quadtree-Based Partitioning in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1720–1731, Dec. 2012.

[10] F. W. Mounts, "A Video Encoding System With Conditional Picture-Element Replenishment," *Bell System Technical Journal*, vol. 48, pp. 2545–2554, 1969.

[11] Harry W. Jones, "A Conditional Replenishment Hadamard Video Compressor," in *Proc. SPIE 0119, Applications of Digital Image Processing*, Andrew G. Tescher, Ed. Dec. 1977, pp. 91–98, International Society for Optics and Photonics.

[12] David Hein and Nasir Ahmed, "Video Compression Using Conditional Replenishment and Motion Prediction," *IEEE Transactions on Electromagnetic Compatibility*, vol. EMC-26, no. 3, pp. 134–142, Aug. 1984.

[13] H. Yu, R. Cohen, K. Rapaka, and J. Xu, "JCT-VC R1015: Common Test Conditions for Screen Content Coding. 18th Meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Sapporo, JP," 2014.

[14] Gisle Bjontegaard, "VCEG-AI11: Improvements of the BD-PSNR model. ITU-T Study Group 16 Question 6. 35th Meeting, Berlin, Deutschland," 2008.

[15] Gary J. Sullivan and Jens-Rainer Ohm, "Meeting report of the fourth meeting of the Joint Collaborative Team on Video Coding," *ITU-T/ISO/IEC Joint Collaborative Team on Video Coding (JCT-VC) document JCTVC-D500*, 2011.