Overview of Screen Content Video Coding: Technologies, Standards, and Beyond

Peng, W.-H.; Walls, F.; Cohen, R.A.; Xu, J.; Ostermann, J.; MacInnis, A.; Lin, T.

TR2016-167 December 2016

Abstract

This paper presents recent advances in screen content video coding, with an emphasis on two state-of-the-art standards: HEVC/H.265 Screen Content Coding Extensions (HEVC-SCC) by ISO/IEC Moving Picture Experts Group and ITU-T Video Coding Experts Group, and Display Stream Compression (DSC) by Video Electronics Standards Association. The HEVC-SCC enhances the capabilities of HEVC in coding screen content, while DSC provides lightweight compression for display links. Although targeting different application domains, they share some design principles and are expected to become the leading formats in the marketplace in the coming years. This paper provides a brief account of their background, key elements, performance, and complexity characteristics, according to their final specifications. As we survey these standards, we also summarize prior arts in the last decade and explore future research opportunities and standards developments in order to give a comprehensive overview of this field.

IEEE Journal on Emerging and Selected Topics in Circuits and Systems

Overview of Screen Content Video Coding: Technologies, Standards, and Beyond

Wen-Hsiao Peng*, Frederick Walls, Robert A. Cohen, Jizheng Xu, Jörn Ostermann, Alexander MacInnis, and Tao Lin

***Abstract*—This paper presents recent advances in screen content video coding, with an emphasis on two state-of-the-art standards: HEVC/H.265 Screen Content Coding Extensions (HEVC-SCC) by ISO/IEC Moving Picture Experts Group and ITU-T Video Coding Experts Group, and Display Stream Compression (DSC) by Video Electronics Standards Association. The HEVC-SCC enhances the capabilities of HEVC in coding screen content, while DSC provides lightweight compression for display links. Although targeting different application domains, they share some design principles and are expected to become the leading formats in the marketplace in the coming years. This paper provides a brief account of their background, key elements, performance, and complexity characteristics, according to their final specifications. As we survey these standards, we also summarize prior arts in the last decade and explore future research opportunities and standards developments in order to give a comprehensive overview of this field.**

***Index Terms*—Display Stream Compression (DSC), High Efficiency Video Coding (HEVC/H.265), Joint Collaborative Team on Video Coding (JCT-VC), Screen Content Video Coding (SCC), Video Electronics Standards Association (VESA)**

I. Introduction

SCREEN content, as typified by the computer and mobile display content shown in Fig. 1, has recently emerged as a popular video type due to the fast rising demands for transporting or storing screen visuals in the form of video. This is driven partly by rapid advances in mobile, cloud and display technologies, which enable a bewildering variety of screen applications over various networks/links, such as wireless displays, second screen, screen sharing and collaboration, cloud computing and gaming, PC-over-IP, display stream compression, etc. In these inter-device-oriented applications, sending screen text and graphics as video data enables platform-independent rendering, making easy exchange of screen content between devices and across platforms possible. Transport of screen content also arises in in-device communications, for example, between the application processor and the display interface in a mobile device [1], [2]. In addition, it is common to store displayed screen images in the display driver for improving panel response time via overdrive techniques [3]. With the ceaseless quest for even higher video/images resolutions, efficient coding of screen content is the key to save bandwidth, power, and/or frame buffer storage.

Fig. 1. Sample pictures of screen content: (a) text and graphics with motion, (b) mixed content, (c) animation, and (d) mobile display image.

Screen content coding poses numerous challenges. It has certain peculiar signal characteristics that make it difficult to compress using the conventional methods designed to code camera-captured content. For example, screen content often features computer-generated objects, text, and line art, which are discrete-tone and full of sharp edges. The loss of few high-frequency components due to compression can make text illegible and thin lines smeared. Camera-captured video commonly constitutes a portion or portions of such content. Therefore, separate tools are needed for coding different parts of the video; moreover, the encoder has to decide wisely which tool to apply. The encoder may also need to consider the human visual system's sensitivity to distortion in different types of content. Generally, the human eye is more sensitive to artifacts occurring in synthetic areas, particularly when a familiar pattern is made unrecognizable due to compression. As such, *visually lossless quality*, showing non-detectable quality degradation to the eye after compression and decompression, or *mathematically lossless quality*, producing an exact replica of the input after compression and decompression, may be required for all or part of the video.

Attempts to find a compact representation for mixture content date back to the late 90's. The ITU-T produced a Mixed Raster Content (MRC) standard [4] for coding compound document images, defining a layer-based imaging model in which an image is segmented into foreground, background and

* Corresponding author

mask layers. Existing codecs were used to encode individual layers rather than utilizing new coding techniques. In this model segmentation plays a critical role; however, it is a non-trivial and potentially time-consuming task. Thus, the layer-based approach is less favorable for real-time screen content applications.

Because of these aspects of layer-based methods, block-based approaches draw much attention for their lower complexity. Most of them base their designs on existing block-based codecs, such as JPEG [5], AVC/H.264 [6] or HEVC/H.265 [7], to leverage their capabilities of coding camera-captured content. Normally, these approaches begin with classification of coding blocks into pictorial and non-pictorial ones, followed by a block-adaptive coding. For the non-pictorial blocks, composed mainly of computer-generated objects, there are a number of coding techniques to exploit their content characteristics for better compression. For example, Nautsch *et al.* [8] extend the MRC representation to the block level along with a matching-pursuit-based block transform. Similarly, Hu *et al.* [9] develop a shape-adaptive transform for a new type of geometry partitioning introduced for intra-predicted residual blocks. By contrast, Lan *et al.* [10] show that certain screen content can benefit from skipping transforms, the effectiveness of which is confirmed experimentally on HEVC/H.265 in [11], [12]. This, however, relies on additional supporting tools such as residual scalar quantization [10], which introduces an adaptive quantization and a short-distance prediction for spatial-domain residual coding. In addition, the discrete-tone nature of screen content motivates a number of studies [10], [13]-[15] on palette coding, which converts pixel values in a block into indices associated with few representative colors[1]. Taking a different approach, Lin *et al.* [16] signal groups of contiguous pixels in single color as dots, lines, and rectangles. Another category of approaches [17]-[23] apply the notion of string matching as in Lempel-Ziv (LZ) coding [24] to remove the redundancy of recurrent patterns in screen content. Some of these approaches develop into methods of copying blocks [25]-[27], strings [28], [29] or lines [30], [31] within the same picture. There are also hybrid methods that combine palette coding with shape-based coding [16], inter-prediction [15], transform coding [32], residual scalar quantization [10], [33], string matching [34]-[36] and portable network graphics (PNG) coding [37], [38].

In addition to the aforementioned methods that aim to maximize compression efficiency, there are lightweight screen content coding techniques targeting low-delay and low-cost applications, e.g., frame buffer compression for display devices and display link compression [1]. In these applications, screen images are usually accessed line-by-line in raster order. Thus, line-based coding is a natural choice. The prior arts in [39]-[47] address the spatial correlation between horizontally neighboring pixels with dictionary-based coding. A dictionary is used to keep a few previously coded pixels in the same line as an input pixel in order to predict its value. Some methods additionally include candidates obtained by extrapolating from the coded pixels [43] and/or by refining them with small difference values [41], [44], [47]. For an input pixel not matching any dictionary pixel, its value may be signaled directly in the bitstream subject to adaptive significant bit truncation [41]. In addition, run length coding [43], [44], [46] is commonly used for representing consecutive identical pixels, which often appear in computer-generated areas. There are also attempts to perform 1-D transform coding using wavelet/Hadamard transform [39], [45], followed by the coding of transform coefficients with an adaptive Golomb-Rice code. Most of these approaches process lines independently to minimize the usage of line buffers. In [39], Dikbas *et al*. relax this constraint to incorporate one additional line buffer for inter-line prediction. At a greater cost, some [3], [48]-[51] turn to block-based approaches, such as block truncation coding [52], which is a moment-preserving, 1-bit quantization scheme, and adaptive block quantization [53], which adapts the step size of a uniform quantizer applied to every pixel in a block according to its dynamic range. In particular, some of the aforementioned techniques [39], [41], [44], [47] also find applications in embedded reference frame compression for video codecs, even though block-based methods [54]-[56] are generally more preferred in this application area. For easy reference and comparison, Table I summarizes the existing methods into few categories according to their processing granularity and applications.

TABLE I
CATEGORIES OF SCREEN CONTENT CODING TECHNIQUES

	Non-lightweight	Lightweight
Applications	Compound documents/images compression, wireless displays, screen sharing, desktop collaboration, cloud gaming, virtual desktop infrastructure	Frame buffer compression for display devices, display stream compression
Methods	**Block-based** – Block-level MRC, Shape-adaptive trans., Matching-pursuit trans., Transform skipping, Pixel-based intra pred., Residual scalar quant., Palette coding, Shape-based coding, Block/line/string copy, Hybrid techniques **Layer-based** – (mainly for documents/images) MRC representation	**Block-based** – Block truncation coding, Adaptive quant. coding **Line-based** – Dictionary coding, Significant bit truncation, Run-length coding, 1-D Hadamard/wavelet

With technologies in this field maturing after a decade of research, and in response to an increasing expectation for industry-wide interoperable solutions, the standards communities have recently developed standards for screen content coding in different application domains. In February

[1]A *pixel* refers collectively to its three color components and a *sample* to one of the components. The value of a pixel is a collection of its samples' values.

TABLE II
SCREEN/MIXED CONTENT CODING STANDARDS

	HEVC/H.265 Screen Content Coding Extensions (HEVC-SCC)	**Display Stream Compression (DSC)**	**Mixed Raster Content (MRC)**
Standards organization	ISO/IEC and ITU-T	VESA	ITU-T
Year	HEVC/H.265 version 1 – 2013 HEVC Range Extension (RExt) – 2014 HEVC-SCC – 2016	DSC version 1.1 – 2014 DSC version 1.2 – 2016	ITU-T T.44 – 1999, 2005
Application	Compression of screen content for applications such as wireless displays, screen/desktop sharing and collaboration, virtual desktop infrastructure, cloud gaming	Compression of raster-oriented screen content for display links/interfaces inside devices (e.g. MIPI DSI) or between devices (e.g. Display Port)	Compression of raster-oriented mixed content with multi-level and bi-level images (e.g. scanned documents)
Input	Images, Video	Images, Video	Images
Color format[a]	Y'CbCr/RGB in 4:4:4/4:2:0	Y'CbCr/RGB in 4:4:4/4:2:2/4:2:0	CIE-LAB/ITU-YCC in 4:4:4/4:2:2/4:2:0
Bit bepth[b]	8 or 10	8, 10, 12, 14, or 16	Any depth
Quality	Up to mathematically lossless	Visually lossless	Mathematically lossless for layers using JBIG or run-length encoding; up to visually lossless for layers using JPEG
Coding framework	Block-based	Line-based	Layer-based
Implementation complexity	High (comparable to HEVC version 1)	Low	High (due to the need for layer segmentation)
Coding tool	**Inter-picture prediction** – Adaptive motion vector resolution **Intra-picture prediction** – Intra block copy Residual DPCM (HEVC-RExt), Cross component prediction (HEVC-RExt) **Transform** – Transform skip (HEVC/H.265 version 1) Adaptive color transform **Dictionary-based coding** – Palette mode	**Intra-picture prediction** – Modified median-adaptive prediction Midpoint prediction Block prediction **Dictionary-based coding** – Indexed color history coding	**Multi-layer coding** – Multi-level coding standard (e.g. JPEG, JBIG, or run-length encoding) for layers containing mainly continuous-tone images or multi-level data; Bi-level coding standard (e.g. JBIG and MMR) for bi-level mask layers.

[a]The color spaces and subsampling formats that can be processed natively by the codec.
[b]The number of bits per color component that can be supported natively by the codec.

2016, the ISO/IEC Moving Picture Experts Group (MPEG) and the ITU-T Video Coding Experts Group (VCEG) concluded their joint standardization work of HEVC/H.265 screen content coding (SCC) extensions (referred hereafter to as HEVC-SCC). This new standard builds on HEVC/H.265 to add support for efficiently coding screen content. In parallel, the Video Electronics Standards Association (VESA®) produced in July 2014 a Display Stream Compression (DSC) standard for lightweight, visually-lossless compression on display links. A new version was released in early 2016 for additional support of higher bit depths (i.e. more bits per color component sample) and extended color formats. Both the HEVC-SCC and DSC represent the latest state-of-the-art in their respective areas. Highlighted in Table II are their design specifics as well as those of the ITU-T MRC.

The remainder of this paper provides a brief account of the HEVC-SCC and DSC from perspectives of their background, coding pipelines, tool features, compression performance and complexity characteristics. We start with the HEVC-SCC in Section II and continue with the DSC in Section III, with a compression performance comparison between these two standards given in Section IV. In Section V, we point out some ongoing research and standardization activities, followed by a summary in Section VI. Part of this paper supplements three earlier overview papers on HEVC-SCC [57]-[59], which were published when the development of HEVC-SCC was not yet fully completed, by presenting the most up-to-date information that reflects the final standard specification.

II. HEVC/H.265 SCREEN CONTENT CODING EXTENSIONS

The HEVC-SCC is a new standardized extension to HEVC/H.265 developed jointly by the ISO/IEC MPEG and ITU-T VCEG in a Joint Collaborative Team on Video Coding (JCT-VC). This extension substantially enhances HEVC/H.265's capabilities for coding screen content. It is expected to be deployed in applications demanding visually or mathematically lossless quality at moderate-to-high compression ratios, e.g. wireless displays, video conferencing with screen sharing, cloud gaming, and PC-over-IP.

A. From HEVC/H.265 to HEVC-SCC

The idea of supporting screen content video in HEVC/H.265 was brought up during its infancy. A preliminary study in [60] pointed out the deficiency of the then first test model (under consideration) of HEVC/H.265 when coding screen content. This study was followed up in JCT-VC with a series of investigations into related issues, including the identification of test materials, quality assessment, and potential coding technologies. At the time, many tools showing promising results were proposed. However, it was decided that tools for SCC-only purposes with significant complexity impact should not be targeted for the base specification of HEVC/H.265. As a result, the HEVC/H.265 first edition culminated with only one SCC tool, known as Transform Skip [11], [12], which is also a tool for achieving lossless compression for generic video content.

Following the issuing of HEVC/H.265 version 1 in April 2013, the study of SCC was continued in the context of its format range extensions (referred hereafter to as HEVC-RExt). The goal of HEVC-RExt was to expand HEVC/H.265 version 1 for a wider range of applications with support for extended color formats (i.e. 4:2:2, 4:4:4, 4:0:0, RGB inputs, etc.), higher bit depths, improved lossless or near-lossless coding, and screen content coding. A large number of SCC tools were investigated. But only few of those that can also benefit the coding of camera-captured and/or mixed content in the aforementioned applications were adopted into the final specification of the HEVC-RExt. This was motivated in part by the results from the joint Call-for-Proposals (CfP) on SCC [61], [62], which led to creating separate, additional extensions for SCC. For that reason, the intra block copy (IBC) tool [25], [63], which had been a major source of coding gain for screen content sequences, was moved from the HEVC-RExt to be included in HEVC-SCC. Thus, the HEVC-RExt ended up having less emphasis on SCC.

Recognizing that the HEVC-RExt had to be wrapped up quickly to address the market needs while there were promising SCC tools not yet fully developed, the ISO/IEC MPEG and ITU-T VCEG issued a joint CfP on SCC in January, 2014, with the aim of developing specific extensions for SCC. A total of seven responses were received from both industry and academia, with the proposed technologies covering the key areas of intra block/line copy, string matching, color palette, color transform, inter color prediction, inter picture coding, in-loop filtering, etc. The results clearly demonstrated that when coding screen content, substantial compression benefit over the HEVC-RExt can be achieved at a reasonable complexity cost. In several SCC test cases, there were proposals found to attain mathematically lossless quality at rate points where the picture quality produced by the HEVC-RExt was not lossless. The standardization process for the HEVC-SCC was then kicked off. After two years of active work, it was officially published as an International Standard in 2016.

Table III summarizes the HEVC/H.265 version 1, HEVC-RExt and HEVC-SCC in terms of their target input formats and SCC tool features. In regard to the SCC tools, these standards form a nested relationship; from left to right; the standard that comes later forms a superset of the previous ones. A plus sign "+" by each tool indicates the incremental adoption of tools from one standard to another.

TABLE III
HEVC/H.265 VERSION 1, HEVC-REXT AND HEVC-SCC

	HEVC/H.265 v1	**HEVC-RExt**	**HEVC-SCC**
Target input	Camera-captured content	Camera-captured content	Screen and mixed content
Color space	Y'CbCr	Y'CbCr, RGB	Y'CbCr, RGB
Color sampling	4:2:0	4:0:0 (monochrome), 4:2:0, 4:2:2, 4:4:4	4:2:0, 4:4:4
Bit depth	8 – 10	>10 (Up to 16)	8 – 10
SCC tool	* Transform skip	* Transform skip + Residual rotation + Residual DPCM + Cross-component prediction	* Transform skip * Residual rotation * Residual DPCM * Cross-component prediction + Intra block copy + Palette mode + Adaptive color transform + Adaptive motion vector resolution

B. Algorithm Overview

Fig. 2 depicts an encoder block diagram of HEVC-SCC. As an extension to HEVC/H.265, it shares the same coding architecture as HEVC/H.265. But, several new elements, some inherited from HEVC/H.265 version 1 and HEVC-RExt, are introduced specifically for SCC, including adaptive color transform (ACT), adaptive motion vector resolution (AMVR), cross-component prediction (CCP), IBC, palette mode, residual rotation (RR), residual differential pulse code modulation (RDPCM), and transform skip (TS).

As with HEVC/H.265, the encoding of an input image begins with dividing it into fixed-size coding tree units (CTU), where the image can be in either the RGB or Y'CbCr color space. A CTU may then be further split using a quadtree partitioning into smaller coding units (CU), and each CU can specify the sizes and partitioning of prediction units (PU) and transform units (TU), which are the basic processing units for inter/intra-picture prediction and spatial transform, respectively.

As before, a CU can be predictively coded using an intra-picture or inter-picture prediction. In particular, a new prediction mode called IBC is devised to address the unique phenomenon of recurrent patterns in screen content. For the ordinary inter-picture prediction, AMVR allows fractional-pixel motion compensation to be disabled adaptively to save signaling overhead for motion vectors, given that much computer-generated screen content moves in whole-pixel increments.

After prediction, the residual block may go through a series of processes before it is entropy coded. First, two

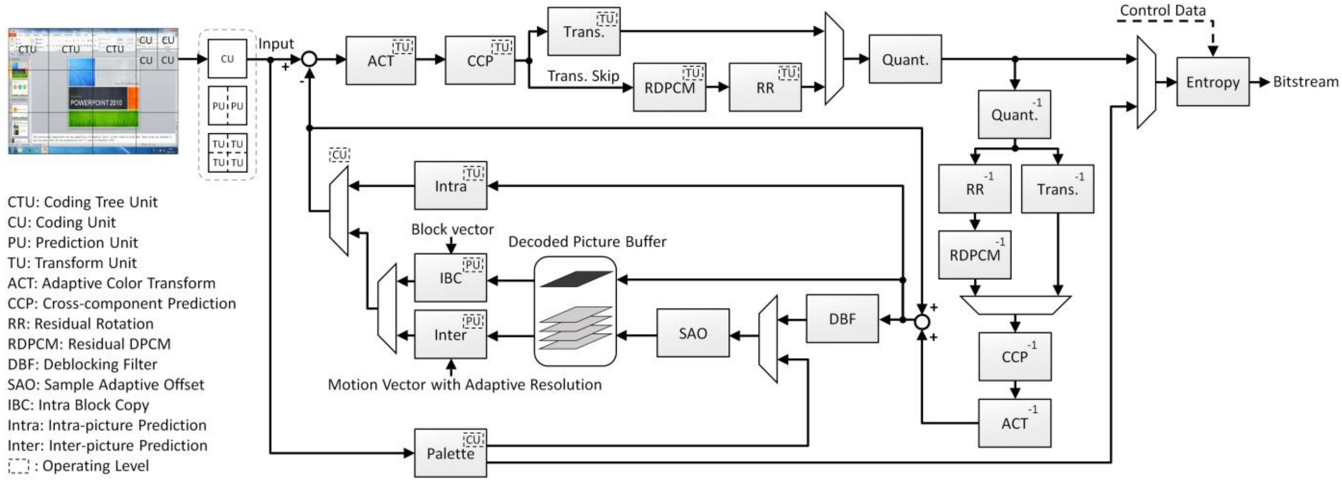

Fig. 2. HEVC-SCC encoder block diagram.

cross-component operations ACT and CCP can be enabled sequentially and independently at the TU level to de-correlate color components. Then, a TS mode can be chosen to skip the spatial transform. When TS is enabled, there is an option of applying RDPCM, which performs pixel-wise spatial DPCM in the residual domain. In the current design, TS blocks adopt the same entropy coding scheme as for transform blocks. Thus, for better entropy coding, a residual rotation by 180 degrees is applied to those with 4x4 intra-picture prediction so that the rotated TS blocks exhibit similar energy distributions to transform blocks.

In addition to these predictive coding modes, there is also a palette mode for coding CUs having few color values. As shown in Fig. 2, each of these modes has a corresponding data path in the encoder for reconstructing signals for future reference.

C. *Tool Features*

1) *Intra Block Copy (IBC)*

IBC [25], [27], [63] is an intra-picture prediction technique for addressing recurrent patterns in screen content. It creates a prediction of the current PU by finding a similar reconstructed block within the same picture, as shown in Fig. 3 (a). Because its operation is similar to motion-compensated prediction, IBC is currently implemented using the syntax for inter-picture prediction by referring to a particular long-term reference picture that indicates prediction from the reconstructed current picture. In particular, no in-loop filtering, i.e. deblocking and sample adaptive offset filtering, is applied to this picture; hence, extra space in the decoded picture buffer may be needed for its storage. In addition, the search area for IBC is limited to 1) the current slice/tile to support independent decoding of slices/tiles and 2) part of the decoded region outside of the current CU in order to support Wavefront Parallel Processing [64], [65] of CTUs and allow parallel processing of PUs within the same CU. An example of this search area is surrounded by the dashed line in Fig. 3 (a). Another distinction from inter-picture prediction is that the block displacement vector is always in integer-pixel

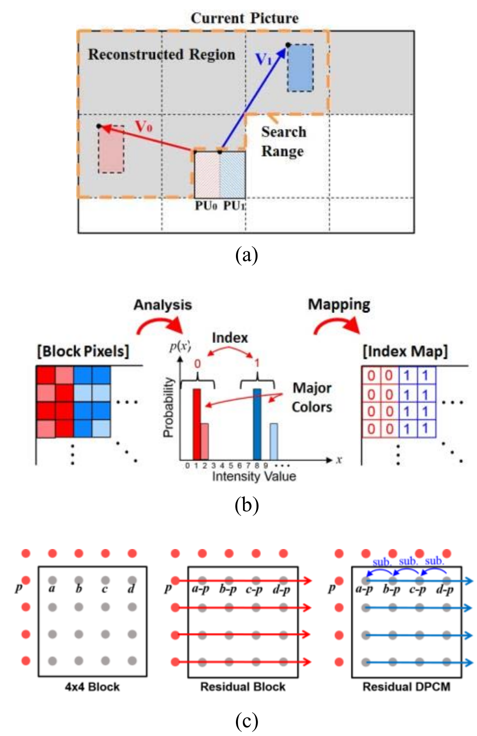

Fig. 3. Illustrations of (a) IBC, (b) palette mode, and (c) RDPCM for a horizontally intra-predicted 4x4 TU.

precision.

2) *Palette Mode (PLT)*

Palette mode [10], [13]-[15], [34], [36] is designed to address the discrete-tone nature of screen content. Very often, some CUs contain only a few colors. In this case, it is more efficient to signal their pixel values directly than using the prediction- or transform-based representations. When coded in palette mode, a CU can accommodate up to 64 representative colors in its color palette table, as shown in Fig. 3 (b). Pixels in the CU are converted into palette indices, which indicate the mapping from their actual values into the representative colors in the table.

The mapping can be approximate, depending on how these representative colors are determined. Both the palette indices and table have to be communicated to the decoder. Palette indices are coded by first scanning in horizontal or vertical transverse order, followed by run-level coding. The level value(s) can be either signaled explicitly with the copy-left mode or inferred from those in the immediately previous row/column(s) with the copy-above mode. Pixels known as *escape pixels* whose values differ from the representative colors are signaled directly without quantization using the maximal palette index in the case of lossless compression. In lossy compression, they are either coded as one of the representative colors or signaled explicitly after quantization, depending on which choice yields a better rate-distortion trade-off. The representative colors in the palette table can be either signaled explicitly or predicted from a list which contains up to 128 most recently-used colors in the palette tables of the last few coded CUs.

3) *Adaptive Color Transform (ACT)*

ACT [66] is a color space conversion technique. It converts the color space of prediction residuals, be it RGB or Y'CbCr, into the YCoCg-R domain [67], [68] to de-correlate color components. This tool works most effectively with RGB input, in which the three color components are usually highly correlated. The forward and inverse transforms of ACT are given as follows:

$$Forward: \begin{cases} Co &= R-B \\ t &= B+(Co>>1) \\ Cg &= G-t \\ Y &= t+(Cg>>1) \\ Co' &= Co>>n \\ Cg' &= Cg>>n \end{cases}, Inverse: \begin{cases} Co &= Co'<<n \\ Cg &= Cg'<<n \\ t &= Y-(Cg>>1) \\ G &= Cg+t \\ B &= t-(Co>>1) \\ R &= Co+B \end{cases},$$

where n can be 1 or 0. Because the dynamic range of Co and Cg is twice as large as the RGB input, both Co and Cg are truncated by one (n=1) least significant bit to ensure that no arithmetic overflow would occur. In lossless coding, this truncation is disabled (n=0). The application of ACT is adaptive at the TU level, and its use is limited to 4:4:4 content.

4) *Cross-Component Prediction (CCP)*

CCP [69], [70] is another tool used to reduce the inter-component redundancy between the prediction residuals of the three color components. When enabled, it exploits the reconstructed residuals of the first component (usually Y or G component) to form a weighted prediction of the other two. Separate weighting factors can be chosen on a TU basis from {0, ±1, ±2, ±4, ±8}/8 for these two predicted components. Once determined, they are applied uniformly to the residuals of the first component to create prediction signals. Currently, the use of CCP is limited to inter-predicted TUs and intra-predicted TUs with Direct Mode, and all must be in 4:4:4 format.

5) *Transform Skip (TS)*

It has been shown that the spatial transform may not yield benefit when coding some types of screen content. In those cases, simply skipping spatial transform can provide satisfactory gain. TS [11], [12] is a coding option which skips spatial transform without changing the subsequent quantization and entropy coding. As the quantization process remains unchanged, the spatial-domain prediction residuals have to be scaled properly to approximate the dynamic range of transform coefficients [12]. Currently, a TU-level flag is used to signal the use of TS.

6) *Residual DPCM (RDPCM)*

RDPCM [71], [72] performs a pixel-wise spatial DPCM in the residual domain. As Fig. 3 (c) shows, this is achieved by predicting each row or column of residual signals in a TU from the adjacent reconstructed row or column. This becomes possible when TS is enabled. RDPCM is adaptively applied to both inter- and intra-predicted TUs with TS. The prediction direction, horizontal or vertical, is explicitly signaled for inter-predicted TUs and is implicitly aligned with the intra prediction direction for intra-predicted TUs. For the latter, it is enabled only for horizontally or vertically predicted TUs.

7) *Residual Rotation (RR)*

RR [73], [74] aligns the energy distribution of intra-predicted TUs with that of transform blocks for better entropy coding. This is accomplished by rotating the TU with TS by 180 degrees. Recall that when TS is enabled, the quantization and entropy coding remain the same as if the input was a transform block. Usually, transform blocks have higher energy concentrated near the low-frequency coefficients at the top-left corner. However, with transform-skipped intra-picture prediction residuals, the prediction error is usually larger at the bottom-right corner, because samples located there are far from those that are used to generate the prediction. Thus, the rotation makes the resulting energy distribution become more like that of a transform block. In the current design, RR takes place after TS and RDPCM, and is applicable to intra-predicted 4x4 TUs only.

8) *Adaptive Motion Vector Resolution (AMVR)*

AMVR [75] allows motion vectors to switch adaptively at the slice level between quarter-pixel and integer-pixel resolutions. Much screen content contains only integer-pixel motion. It is thus sensible to signal motion vectors in integer-pixel precision in order to reduce motion vector overhead. When enabled, both components of a motion vector predictor are rounded to integer-pixel resolution to ensure that the corresponding motion vector differences are also integer-valued.

D. *Performance and Complexity*

The HEVC-SCC standard demonstrates substantial coding gains over the HEVC-RExt (Main 4:4:4 Profile [76]) and AVC/H.264 (High 4:4:4 Predictive Profile [77]) when coding screen content. Fig. 4 shows the bit-rate savings relative to these prior standards under the common test conditions [78] developed by the JCT-VC. Both lossy and lossless compression scenarios are tested for a wide variety of RGB/Y'CbCr inputs in 4:4:4/4:2:0 formats using all intra (AI), random access (RA), and low-delay B (LB) coding configurations. The "TGM",

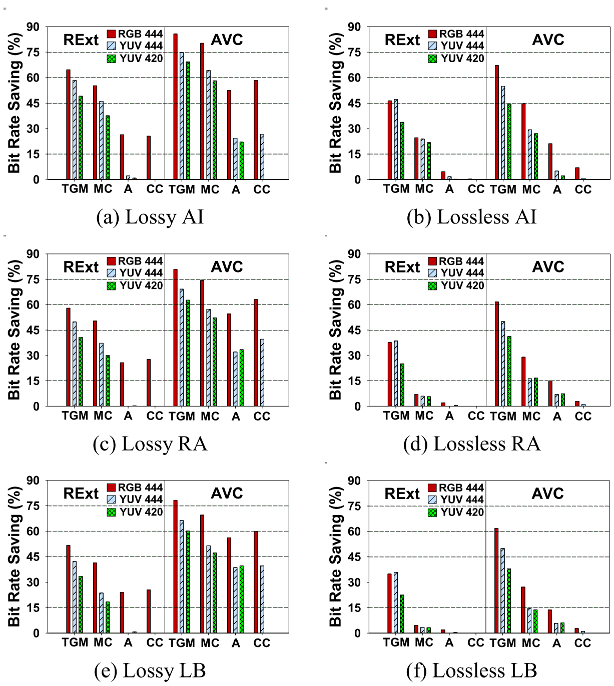

Fig. 4. BD-rate savings achieved by HEVC-SCC relative to HEVC-RExt and AVC/H.264 in percentage terms. Test: HEVC-SCC (SCM-5.2), baseline anchors: HEVC-RExt (HM-16.6) and AVC/H.264 (JM-19). For brevity, Y'CbCr is denoted as YUV.

"MC", "A", and "CC" refer respectively to the four types of content: text and graphics with motion, mixed content (i.e. a mixture of TGM-type content and camera-captured content), animation, and camera-captured content. In the interest of space, the performance for lossy compression are provided for the Y or G component only, based on the BD-rate metric [79].

Compared to HEVC-RExt, HEVC-SCC yields, on average, 40-50% bit-rate reductions in typical 4:4:4 screen content sequences (TGM and MC) using lossy compression and 10-40% using lossless compression. It is also seen that the improvements in the 4:2:0 Y'CbCr case are generally smaller than in the 4:4:4 Y'CbCr counterpart. This is because ACT is only used for 4:4:4 content. In animation and camera-captured sequences, the rate savings are close to zero, except when the input is in RGB space. This suggests that other than ACT, the other tools, including IBC, palette mode and AMVR, may not be as effective for coding these types of content as for coding TGM and MC content. While the finding can be justified for camera-captured content, it does imply that there is still room for improvement in coding animation content. Compared to AVC/H.264, similar trends yet more improvements are observed. The bit rate reductions increase to 65-75% for TGM and MC sequences under lossy compression and 25-55% under lossless compression.

Fig. 5 presents a breakdown of contributions of different SCC tools by showing the bit rate inflation percentage when each single tool is disabled. The baseline anchor is the HEVC-SCC with all tools enabled. Obviously, the larger the inflation, the higher contribution a tool makes to the overall coding performance. This also partly reveals how different tools interact with each other. The results are provided for TGM and MC sequences in 4:4:4 format and with AI and LB coding only, to save space. It is clear that among all the tools, IBC is the most critical one. It can benefit considerably the coding of both TGM and MC sequences. By contrast, the advantages of palette mode and ACT are seen to be more content-dependent. For example, palette mode is more critical to TGM sequences than to MC sequences given that portions of the MC sequences contain camera-captured content not having limited numbers of distinct colors, and ACT is beneficial mostly to RGB inputs. It is noteworthy that turning off CCP shows only a marginal impact on coding performance. This suggests that the competing tool ACT can compensate for most of the loss. This is however not the case when ACT is turned off (in which case CCP is on). In addition, RDPCM is mainly effective for lossless coding. TS, RR and AMVR exhibit modest effects but require only minor changes to the base specification.

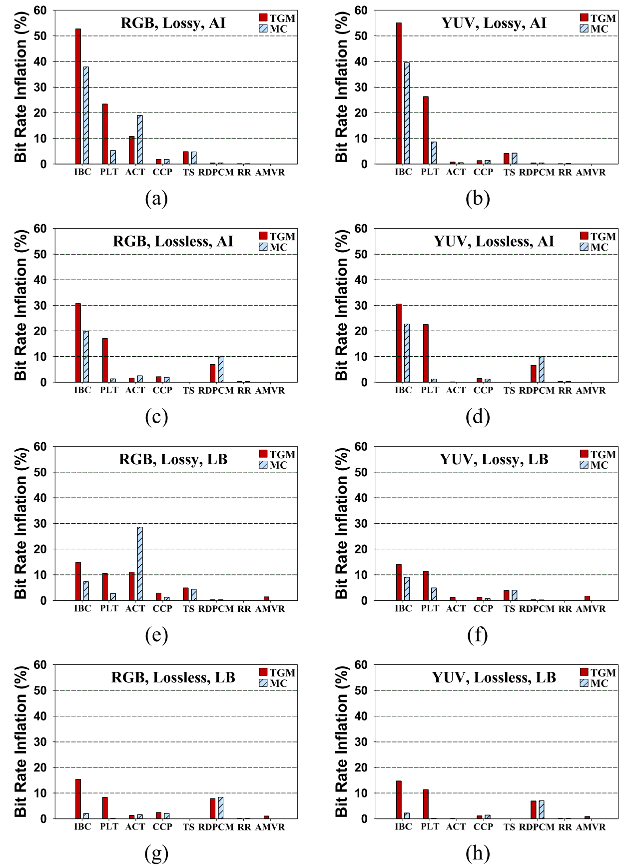

Fig. 5. A breakdown analysis on the contribution of each tool in HEVC-SCC.

In addition to compression efficiency, complexity was also carefully considered during the development of these SCC tools in order to make sure that the decoder would not be burdened by excessive computation, memory access, and syntax parsing. Furthermore, minimizing changes to the existing designs for HEVC/H.265 version 1 or HEVC-RExt were considered. For example, IBC was identified a major source of complexity in terms of memory access. Investigations were made into the smallest allowable size of IBC blocks so that in the worst case, its memory access requirements would not exceed those for motion-compensated prediction. The

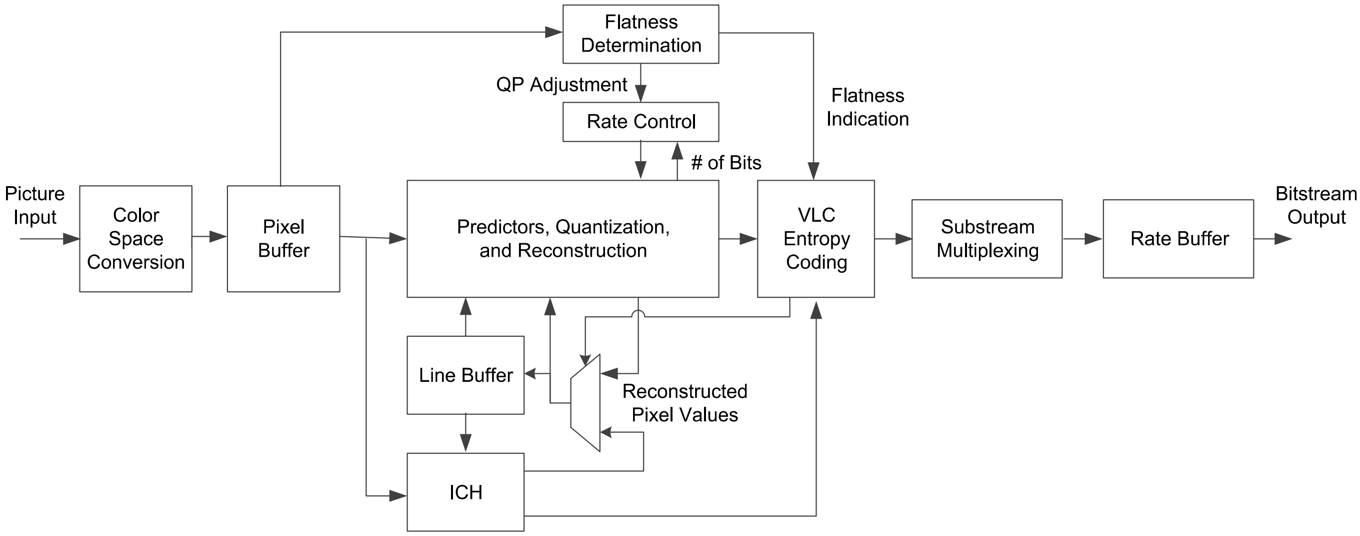

Fig. 6. DSC encoder block diagram [2].

design of palette mode follows a similar philosophy to avoid burdening the syntax parsing. In addition, since both ACT and CCP operate in the residual domain, sample values of the three color components can be reconstructed in parallel. Overall, a slight increase in the decoder complexity as compared to the HEVC/H.265 version 1 is expected.

III. Display Stream Compression

VESA has recently standardized on DSC, a lightweight, visually lossless compression algorithm for coding all types of screen content. In contrast to other standards, the codec is designed to process pictures in raster order in real time, targeting very low-cost ASIC implementations with minimal on-chip memory. DSC targets visually lossless performance at specific constant bit rates; better compression performance is sacrificed in the design in order to minimize the silicon cost and complexity. Visually lossless performance means that users cannot tell that the compression is taking place. There are no particular constraints on the types of images that are visually lossless, and DSC is tested using a wide array of natural and synthetic images and video, including pathologically difficult material such as random noise and multi-component frequency sweep patterns.

A. Background and Motivation

Historically, many types of consumer and mobile devices have employed lightweight compression for the purposes of saving memory and memory bandwidth. These proprietary algorithms may be designed for different levels of compression, quality, or complexity based on the needs of the product that they are included in.

The industry recognized the advantages of sending compressed picture data over display links, and many different proprietary compression algorithms were developed. Both source and sink manufacturers attempted to license their technology for use on the other side of the link. But these proprietary algorithms varied widely in terms of compression, quality, and complexity. Licensing and verifying a plethora of compression algorithms was not a practical solution for component vendors, and vendors preferred a single, high-quality, interoperable solution.

Therefore, in 2012 VESA formed the DSC task group to standardize a low-cost, visually lossless compression algorithm that could be adopted by both sink and source manufacturers. VESA issued a Call for Proposals that targeted visually lossless quality at 12 bits per pixel (bpp)[2] [1]. This target was later reduced to 8 bpp after feedback from mobile device manufacturers. Six proposals were received from industry and academia, and the final specification was published in April 2014. DSC is now widely deployed over many types of display links such as MIPI® DSI, DisplayPort, and Embedded DisplayPort.

A revised DSC specification (1.2) was released in early 2016 in order to better address the requirements for external links between consumer devices. This update includes the addition of native 4:2:2 and 4:2:0 coding as well as better support for Y’CbCr and high bit depths.

B. Algorithm Overview

A DSC encoder block diagram is shown in Fig. 6. RGB sources are converted to YCoCg-R [67], [68] in the color space conversion. A short pixel buffer allows the encoder to make a determination of whether a busy-to-flat transition occurs in the upcoming pixels. The prediction, quantization, and reconstruction block includes three prediction modes and power-of-two quantization with different quantization levels for luma and chroma. The Indexed Color History (ICH) allows reuse of recent reconstructed pixel values. The rate control adjusts the quantization parameter (QP) to provide visually lossless picture quality while managing the rate buffer fullness. The entropy coding operates on 3x1 groups and creates

[2]The ratio of the number of compressed bits of an image to its pixel count, which is commonly used to specify the target compression ratio for an image codec like DSC.

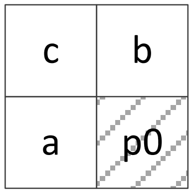

Fig. 7. Pixel constellation for MAP.

component-wise code streams that are multiplexed together using a headerless packet multiplexing scheme called substream multiplexing (SSM).

C. Tool Features

1) Prediction and Quantization

Similar to many contemporary coding algorithms, DSC predicts pixels using a prediction algorithm and encodes the resulting quantized residuals. DSC is a DPCM algorithm, meaning the prediction loop operates on samples directly.

The main prediction method is a modified form of median-adaptive prediction (MAP) [80]. Conceptually, MAP predicts each sample by taking the median of three reasonable predictors (the left adjacent sample **a**, the upper adjacent sample **b**, and a prediction that assumes the sample values all lie in the same plane, **a** + **b** – **c**) as illustrated in Fig. 7.

The MAP prediction that is used in DSC makes two modifications to conventional MAP. First, the reference samples used from the previous line are blended with a horizontally low-pass filtered version of the pixels in the previous line. The blend is controlled by the current quantization level, so in areas where quantization is high, the blend favors the low-pass filtered pixels. This modification helps smooth out quantization artifacts that can be created by MAP. The second modification to MAP involves a slight change to the predicted value for two out of three samples in each group to improve decoder throughput. MAP requires the reconstructed value of the left adjacent pixel in order to make a prediction. Since DSC targets applications where decoder clock speed may be much lower than the pixel rate, two of the three prediction equations include residuals rather than the left reconstructed sample values. The inclusion of residuals means that the encoder is still limited to computing one sample per clock; however, decoders have access to the residuals at the beginning of a clock cycle and can therefore process three sample values simultaneously.

The second prediction type in DSC is called block prediction (BP). The predictor that is used is a reconstructed pixel at a specified offset position to the left of the current pixel. The offset is referred to as the BP vector and applies to a 3x1 group. The BP vector selection and prediction mode decision is made using Sum-of-Absolute-Differences-based (SAD-based) searches of the previous line's reconstructed samples in both encoder and decoder, so no bits are sent to communicate the BP mode.

The last prediction type is midpoint prediction (MPP) and is used to limit the size of residuals by selecting a predictor near the midpoint of the sample range. The use of MPP is explicitly signaled in the bitstream.

Quantization in DSC is done using powers of two with rounding. This can be implemented efficiently using a single add and programmable shift in hardware. The quantization for luma is generally higher than chroma, and the luma and chroma quantization levels are derived from the QP from the rate control.

2) Indexed Color History (ICH)

Another primary coding mode is the ICH mode, which is similar to other palette-based coding methods. The ICH contains a 32-entry register file of recently-coded sample values. On lines other than the first line of a slice, seven of the 32 entries refer to sample values from the previous line. Any of the entries can be directly referenced by transmitting a 5-bit index value in the entropy encoder.

The selection of candidate ICH entries is done using a weighted SAD, where the luma distortion is more heavily weighted than the chroma.

The decision on whether to use ICH mode or prediction is based on a rate-distortion measure that includes a constant lambda cost. The measure definition and lambda value are chosen to minimize the hardware cost required to implement the selection algorithm.

ICH helps performance for many types of content, particularly computer-generated graphics. Subpixel rendering algorithms such as Microsoft ClearType^{TM} [81] enhance the resolution of rendered fonts by activating only some of the display subpixels on edges, which from a full-pixel point of view appear to be unusual colors on the edges of letters. The ICH efficiently and accurately codes such subpixel-rendered text.

3) Entropy Coding and Substream Multiplexing

The entropy coding scheme in DSC was chosen to simplify high-throughput hardware implementations as much as possible while maintaining good coding efficiency. Predicted residuals are coded using a unique entropy coding scheme called Delta-Size Unit Variable-Length Coding (DSU VLC). A set of three quantized residuals for a given component are grouped together into a unit. Each residual has a size (Table IV), which is the number of bits that would be required to code the residual in two's complement. The maximum size of the three residuals is the minimum required size that can be used to send the residual data for a unit.

A unary-coded prefix signals the size of the residual data. For each unit, a size prediction is made based on the sizes of previously coded residuals. If the predicted size is less than the required size, a unary code is sent that indicates how many additional bits are needed. If the predicted size is greater than or equal to the required size, a one-bit unary code representing zero is sent and the residuals are transmitted in fields of the predicted size.

The DSU VLC coding scheme allows straightforward implementations that can decode three residuals in a single clock cycle. DSC implements a headerless multiplexing scheme (Substream Multiplexing, or SSM) which allows low-cost decoder implementations to operate at three pixels

TABEL IV
EXAMPLE RESIDUAL REPRESENTATIONS

Residual	Size (in bits)	Representation
-3	3	101b
-2	2	10b
-1	1	1b
0	0	None
1	2	01b
2	3	010b
3	3	011b

(i.e. 9 residuals) per clock. The SSM multiplexes together the three component-wise entropy code streams into fixed-size packets with no headers. The order of the packets is defined to be an order that is optimal for decoders, which means that encoders need to have a model of the decoder demultiplexing behavior to put the packets in the correct sequence. Although SSM adds some cost to encoders, it saves significant cost on decoders compared to alternatives.

4) Rate Control

The rate control adjusts the QP to maximize subjective picture quality and to ensure that the rate buffer neither overflows nor underflows. It exploits perceptual masking [82] so that busy areas are coded with a higher QP and flat areas are coded with a lower QP, all the while managing the buffer and ensuring the number of bits generated for each slice is correct.

The DSC rate control has several key characteristics. It is designed to update the QP for every 3-pixel group, which can be helpful in adapting to the content, particularly since the quantization is constrained to be a low-cost power-of-two scheme. Because the QP changes frequently, it is not efficient to explicitly encode QP information, so the QP is primarily derived indirectly by encoder and decoder from the buffer fullness and the activity of recently decoded groups. Lastly, since an implicit quantization scheme does not have the ability to predict when content changes from complex to flat, DSC has an explicit flatness syntax that allows signaling such a transition for any group.

D. Performance and Complexity

DSC is designed to be visually lossless for 4:4:4 content at bit rates greater than or equal to 8 bpp and is designed to minimize the required hardware complexity for that performance point. ISO 29170-2 was developed to evaluate the performance of a visually lossless codec, and test results using that methodology have shown DSC to be visually lossless at 8 bpp for 8 bits/component RGB sources [83].

The DSC 1.2 specification was released in January 2016 [84], and it includes new tools for natively coding 4:2:2 and 4:2:0 pictures. Although formal subjective test results for 4:2:2 and 4:2:0 modes are not yet available, preliminary results indicate that 4:2:2 pictures are visually lossless at bit rates greater than or equal to 7 bpp and 4:2:0 pictures are visually lossless at bit rates greater than or equal to 6 bpp. As is the case in other coding standards, compressing higher bit depth sources does not require the bpp to be scaled to maintain equal subjective quality; the compression ratio improves in higher bit depth modes.

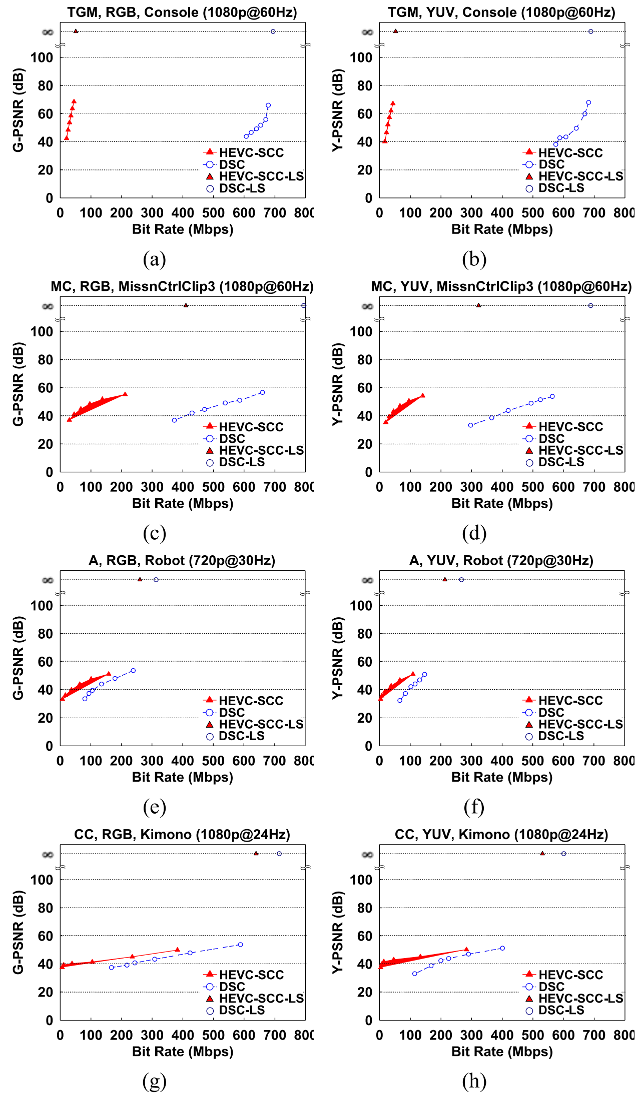

Fig. 8. Rate-distortion performance comparison of HEVC-SCC and DSC for RGB and Y'CbCr (denoted as YUV) inputs in 4:4:4 format. The suffix "-LS" indicates the results with their respective mathematically lossless compression modes.

DSC is optimized for low-cost, real-time implementations in custom hardware. FPGA implementations may be expensive, and CPU implementations may not run in real time. Throughput for 4:4:4 mode is typically 1 pixel/clock for encoders and 3 pixels/clock for decoders. Higher throughput is possible using parallel instances and multiple slices per line. Throughput for the new 4:2:2 and 4:2:0 modes is typically 2 pixels/clock for encoders and 6 pixels/clock for decoders.

IV. COMPARISON OF HEVC-SCC AND DSC

Fig. 8 compares the objective rate-distortion (R-D) performance of HEVC-SCC and DSC for four typical sequences of the types "TGM", "MC", "A", and "CC", respectively. Both codecs are configured to perform fixed QP encoding under the all intra configuration. In the results of HEVC-SCC, the first four R-D points at lower bit rates are produced following the common test conditions [78]. Then, the

QP value is extended further towards the zero end to generate the rest of R-D points with their PSNR generally going beyond 50 dB for visually lossless quality. For DSC, the QP's are chosen to have a comparable quality range to that of HEVC-SCC. In addition, results for mathematically lossless compression are provided for both codecs, where their bit rates are of major concern since the PSNR's are infinite.

As Fig. 8 shows, the compression efficiency of HEVC-SCC is significantly better than that of DSC when coding the TGM and MC sequences. HEVC-SCC achieves mathematically lossless quality at bit rates where the Y- or G-PSNR of DSC is barely 40 dB, below which coding artifacts are likely visible. It is however noteworthy that DSC performs closer to HEVC-SCC in the other two types of tested sequences, especially at high rates. The same observations hold true regardless of the input color space and compression mode.

These results are not to conclude that HEVC-SCC is superior to DSC. Essentially, they target applications of very different nature, having distinct design constraints. For instance, both the encoder and decoder of DSC are intended to be implemented in low-cost hardware for applications requiring small memories for ultra-low end-to-end delay, whereas HEVC-SCC allows for both hardware and software implementations, with low cost and low delay features being generally preferred but not required. Using their current software implementations, we observe that HEVC-SCC (SCM-5.2) has an average encoding runtime that is about 20 to 30 times that of DSC (version 1.48)[3]. These numbers are not to be interpreted as being proportional to their implementation complexity, although they do suggest that the encoding process for DSC may require significantly fewer operations than for HEVC-SCC. To better appreciate and conceive their complexity, the reader is referred to Table V for their design requirements and features.

[3]The average encoding time for producing compressed bitstreams associated with all the R-D points of the eight test sequences in Fig. 8. The runtime measurement is done on a cluster composed of 16 nodes, each containing 16GB of RAM and an Intel i7-860 processor.

TABLE V
REQUIREMENTS AND FEATURES OF HEVC-SCC, DSC AND JPEG XS

	HEVC-SCC	**DSC**	**JPEG XS**[a]
Quality	Up to mathematically lossless	Visually lossless	Visually lossless
Target implementation	Hardware and software	Hardware	Hardware and software
Complexity	Comparable to HEVC Main 10 4:4:4	Light weight compression	Light weight compression
External memory for hardware	Usually required	No	No
Arithmetic data dependency[c]	1 Picture[b]	2 lines of pixels in raster order	32 lines of pixels in raster order
Low end-to-end delay[d]	Optional (via encoder configuration)	Obligatory	Obligatory
Robustness to multiple encoding and decoding cycles	Optional (via lossless compression)	No	Obligatory
Parallel processing features	Slices, dependent slices, tiles, wavefronts	Slices	--
Normative encoder	No	Yes	--

[a]Information is subject to change as JPEG XS was in development at the time of writing.
[b]Intra-picture coding with one slice per picture.
[c]Data dependency inherent in the codec design.
[d]Collective delays for encoding, transmission, and decoding.

V. ONGOING WORK AND FUTURE OUTLOOK

It is possible that both HEVC-SCC and DSC will likely be the leading formats in the marketplace in the coming years. Their publication marks the start of more work in system implementation and application development. In addition, there is plenty of potential scope for improvement. Many open issues that require further research were discovered during the standardization process. This section highlights some ongoing work and an outlook for the future.

A. Techniques beyond HEVC-SCC

Techniques adopted into the standards often are the tip of the iceberg. During the development of the HEVC-SCC, two techniques, intra line copy (ILC) [30] and intra string copy (ISC) [28], attracted much attention due to their significant compression benefit on top of the HEVC-SCC. The ILC technique extends the notion of IBC by allowing a PU to be further split into horizontal or vertical lines, each associated with a line vector specifying where within the same current picture its prediction signal comes from. In a sense, ILC performs intra-picture copying at much finer granularity than IBC. By relaxing the constraint that a PU has to be split into lines of equal size, ISC is even more flexible, allowing a variable-length string of consecutive pixels in horizontal or vertical scan order to be the basic unit for copying. In particular, the string can be as short as containing only one pixel or as long as the PU or CU size. Pixel copying at finer granularity leads to more compression gain for screen content. This, however, has complexity implications on memory access bandwidth, particularly when the line or string data need to be fetched from external memory. For this reason, these tools were not adopted into the final standard specification. Further study is currently in progress to find a sweet spot between complexity and compression performance [31]. In addition, efforts are being made to harmonize palette coding and ISC [29], [85] for a more general design. Additional techniques that can be used to extend HEVC or similar frameworks include allowing the rotation of blocks for block matching [86], [87] and improving the quality of downsampled chroma components for screen content [88].

In addition to these HEVC/H.265-based techniques, there are also distinct approaches in literature. For example, Yang *et al.* [89] introduce sparse coding for textual blocks. An over-complete dictionary is learnt to acquire a sparse representation for textual blocks via matching pursuit. We expect that there will be additional radical techniques attempting to better leverage the content characteristics outside the constraints of a standard, including studies to improve the coding of screen content that does not benefit from the

HEVC-SCC, such as animation/gaming content whose signal characteristics have not been fully explored yet.

B. Encoding Optimization for HEVC-SCC

Encoding optimization is another potential area for future research. Like prior MPEG and VCEG standards, the HEVC-SCC does not specify how the encoder should determine the coding parameters. As observed in some prior work, encoder optimization tasks require particular considerations with regard to the characteristics of screen content. For example, Zhu *et al.* [26], [90] observe large motion between successive pictures and large displacement between recurrent patterns within the same picture. Therefore, a whole-frame search is desirable for achieving the full potential of motion-compensated prediction and IBC. To avoid excessive search operations, they propose a hash-based search algorithm, which uses a hash table to tabulate possible search locations for coding a block. However, the size of this hash table is comparable to, or even larger than, the decoded picture buffer. Details on the current implementation of hash-based search in HEVC-SCC can be found in [91]. For hardware and software implementations on devices having limited resources, additional studies can be found in literature. Zhang *et al.* [92] further observe that the large motion often occurs on an arbitrarily-sized region basis. They propose a region-based motion detection to speed up the motion estimation process. There are additional studies [93]-[95] on fast search for IBC, which suggest that more work can be done in this area. In addition, Guo *et al.* [96] indicate that rate control for screen content needs to be re-examined with reference to the frequent occurrence of abrupt motion. Somewhat related to this work is [97], which proposes to adapt the quantization parameter to the varying signal characteristics within a screen image. The preponderance and variety of improvements such as these suggest that encoding optimization for HEVC-SCC is a promising direction for continued study.

C. Objective Quality Measurement

Objective quality measurement for predicting the subjective quality of compressed screen content video has been found difficult due to the varied content characteristics. The crude Peak-Signal-to-Nosie-Ratio (PSNR) metric is still widely used for the reasons of familiarity, tractability and lack of convenient alternatives. In [98], Yang *et al.* carry out a study on quality assessment for screen content images, suggesting the limited success of existing image quality assessment models in predicting the human's subjective perception. They make an interesting observation that the quality of the textural part correlates higher with the overall image perception. Shi *et al.* [99] conduct a similar investigation, with an emphasis on the type of distortion caused by compression with HEVC/H.265 or HEVC-SCC. They conclude that the Visual Information Fidelity index [100] can better reflect the subjective quality of compressed screen images, as compared to other state-of-the-art models. Recognizing that mathematically lossless quality may be an excessive requirement for typical applications, Lin *et al.* [101] propose a color-count-dependent PSNR measure to define a sufficient condition for a compressed screen image to be subjective visually lossless. All these studies are still in their early stage. With only few attempts specifically targeting screen content made so far, additional research opportunities exist in this area.

D. Implementations and Applications

The release of the HEVC-SCC and DSC is encouraging more investment on their system implementation and application development. With its hardware-friendly design, there are already a few hardware solutions for DSC on the market [102]. We also expect that the just arrived HEVC-SCC will soon receive full attention in the multimedia design community. In fact, many of its target applications, e.g. cloud gaming [103], wireless displays [104], desktop sharing and collaboration, are increasingly being deployed right now. Many of these applications adopt AVC/H.264 [6] as their compression solution. The emergence of HEVC-SCC may change the landscape for these fast growing applications. For its much improved compression efficiency, many existing compromises may have to be revisited [105]-[107].

E. Ongoing Standardization Activities

More standardization work is underway to address the expanding use of screen content.

1) Screen Content Coding in Audio Video Coding Standard

The Audio Video Coding Standard (AVS) Workgroup of China is moving fast towards adding screen content coding support to their AVS standard portfolio, now also the IEEE 1857 series [108]. In April 2016, they released the first working draft [109] of the AVS SCC extension, along with the reference software and the common test conditions [110], where more test sequences [111] featuring difficult-to-compress elements, such as computer-generated objects with translucent blending and image content rendered with subpixel anti-aliasing techniques, are included. It is worth noting that the first working draft adopts a Universal String Prediction technique based on ISC [29], [84], [112] with efforts made to facilitate low complexity implementations.

2) Future Video Coding

Targeting a potential new standard by the year 2020, the ISO/IEC MPEG and ITU-T VCEG established a Joint Video Exploration Team in October 2015 to study video coding technologies with merits beyond HEVC/H.265. This standard is intended to support the coding of a wide variety of video content including screen content and gaming content in addition to camera-captured content. Several workshops were held to collect input from industry and academia in order to define a set of requirements [113]. In the meantime, the draft reference software, the Joint Exploration Test Model [114], has been made available for research and experiments.

3) Advanced Display Stream Compression

VESA issued a Call for Technology [115] in order to standardize a significantly more complex codec called Advanced Display Stream Compression (ADSC) that is visually lossless at a lower bit-rate than DSC. Since ADSC targets a different complexity versus coding efficiency tradeoff than DSC, transport specifications may utilize either ADSC or

DSC depending on the requirements of the link. Although the specification is still in development, the Call for Technology anticipates a specification release near the end of 2016.

4) JPEG XS

The Joint Photographic Experts Group (JPEG) had initiated the standardization of JPEG XS [116], a low-latency, lightweight image coding system with potential applications in video link compression, frame buffer compression, and real-time video storage. The JPEG-XS aims to achieve visually lossless quality at compression ratios between 2 and 6. Its other notable features include a low combined encoding-decoding latency (e.g. smaller than 32 video lines), low implementation complexity in both hardware (FPGA or ASIC) and software implementations, and robustness to multiple encoding-decoding cycles and transmission errors. These and other requirements are compared side-by-side with those of HEVC-SCC and DSC in Table V. As part of this standardization effort, Willeme *et al.* [117] conduct a preliminary comparison among several existing lightweight compression schemes. A Call for Proposals [118] was issued in March 2016, with much further work being prepared towards publishing an International Standard in mid-2018.

VI. Summary

In this paper, we present an overview of screen content video coding techniques, which includes a review of prior arts and an introduction to two newly completed standards, the HEVC-SCC and DSC. Screen content coding is a challenging task due to dynamic signal characteristics accompanied by the varied level of subjective perception of coding artifacts. Only recently has the study of screen content coding intensified, although much was dedicated to the coding of compound documents and images, which share characteristics similar to screen images. Our survey classifies prior arts into three categories: the layer-based, block-based and line-based methods. Among them, the latter two went mainstream over the last decade as part of HEVC-SCC and DSC, respectively. On top of the HEVC/H.265, the HEVC-SCC introduces several new tools to leverage screen content characteristics for improved compression efficiency. The reported results demonstrate its significant improvements over the HEVC-RExt and AVC/H.264 when coding screen content. DSC targets lightweight, low-latency compression for display links, allowing very low cost implementations. Both codecs involve similar design principles, such as palette-based coding, pixel-based intra prediction and intra-picture copying. Being recently-issued international standards, they are attracting investment on system implementation, application development as well as algorithm optimization and improvement.

References

[1] *VESA finalizes requirements for display stream compression standard*, Jan. 2013 [Online]. Available: http://www.vesa.org/news/vesa-finalizes-requirements-for-display-stream-compression-standard/

[2] F. Walls and A. MacInnis, "VESA display stream compression: an overview," *SID Symp. Digest Tech. Papers*, vol. 5, no. 1, June 2014, pp. 360-363.

[3] S. Kim, D. Lee, J.-S. Kim and H.-J. Lee, "A block truncation coding algorithm and hardware implementation targeting 1/12 compression for LCD overdrive," *J. Display Technol.*, Oct. 2015.

[4] *Mixed Raster Content (MRC)*, ITU-T Rec. T.44, 1999.

[5] G. K. Wallace, "The JPEG Still Picture Compression Standard," *IEEE Trans. Consum. Electron.*, vol. 38, no. 1, pp. 18-34, Feb. 1992.

[6] T. Wiegand, G. J. Sullivan, G. Bjontegaard and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560-576, July 2003.

[7] G. J. Sullivan, J.-R. Ohm, W.-J. Han and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649-1668, Sep. 2012.

[8] H. Nautsch and J. Ostermann, "Transform coding of compound images using matching pursuit," in *Proc. Picture Coding Symp. (PCS)*, May 2012, pp. 273-276.

[9] S. Hu, R. A. Cohen, A. Vetro and C.-C. J. Kuo, "Screen content coding for HEVC using edge modes," in *Proc. IEEE Int. Conf. Accoust., Speech, Signal Process. (ICASSP)*, May 2013, pp. 1714-1718.

[10] C. Lan, G. Shi and F. Wu, "Compress compound images in H.264/MPGE-4 AVC by exploiting spatial correlation," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 946-957, Apr. 2010.

[11] A. Gabriellini, M. Naccari, M. Mrak and D. Flynn, "Spatial transform skip in the emerging high efficiency video coding standard", in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2012, pp. 185-188.

[12] M. Mrak and J. Xu, "Improving screen content coding in HEVC by transform skipping," in *Proc. European Signal Process. Conf. (EUSIPCO)*, Aug. 2012, pp. 1209-1213.

[13] W. Ding, Y. Lu and F. Wu, "Enable efficient compound image compression in H.264/AVC intra coding," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2007, pp. 337-340.

[14] Z. Pan, H. Shen, Y. Lu, S. Li and N. Yu, "A low-complexity screen compression scheme for interactive screen sharing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 6, pp. 949-960, June 2013.

[15] W. Zhu, W. Ding, J. Xu, Y. Shi and B. Yin, "Screen content coding based on HEVC framework," *IEEE Trans. Multimedia*, vol. 16, no. 5, pp. 1316-1326, Aug. 2014.

[16] T. Lin and P. Hao, "Compound image compression for real-time computer screen image transmission," *IEEE Trans. Image Process.*, vol. 14, no. 7, pp. 993-1005, Aug. 2005.

[17] N. C. Francisco, N. M. M. Rodrigues and E. A. B. da Silva, "Scanned compound document encoding using multiscale recurrent patterns," *IEEE Trans. Image Process.*, vol. 19, no. 10, pp. 2712-2724, Apr. 2010.

[18] T. Lin, P. Zhang, S. Wang and C. Chen, "Mixed chroma sampling-rate high efficiency video coding for full-chroma screen content," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 1, pp. 173-185, Jan. 2013.

[19] T. Lin, K. Zhou, X. Chen and S. Wang, "Arbitrary shape matching for screen content coding," in *Proc. Picture Coding Symp. (PCS)*, Dec. 2013, pp. 369-372.

[20] S. Wang and T. Lin, "A unified LZ and hybrid coding for compound image partial-lossless compression," in *Proc. Int. Congr. Image Signal Process. (CISP)*, Oct. 2009, pp. 99-103.

[21] S. Wang and T. Lin, "United coding for compound image compression," in *Proc. Int. Congr. Image Signal Process. (CISP)*, Oct. 2010, pp. 566-570.

[22] S. Wang and T. Lin, "Compound image compression based on unified LZ and hybrid coding," *IET Image Process.*, vol. 7, no. 5, pp. 484-499, May 2013.

[23] W. Zhu, J. Xu, W. Ding, Y. Shi and B. Yin, "Adaptive LZMA-based coding for screen content," *in Proc. Picture Coding Symp. (PCS)*, Dec. 2013, pp. 373-376.

[24] T. M. Cover and J. A. Thomas, "Lempel-Ziv coding," in *Elements of Information Theory*, 2nd ed., Hoboken, New Jersey, Wiley, 2006, ch. 13, sec. 4, pp. 440-442.

[25] M. Budagavi and D.-K. Kwon, "Intra motion compensation and entropy coding improvements for HEVC screen content coding," in *Proc Picture Coding Symp. (PCS)*, Dec. 2013, pp. 365-368.

[26] W. Zhu, W. Ding, J. Xu, Y. Shi and B. Yin, "2-D dictionary based video coding for screen contents," in *Proc. Data Compression Conf. (DCC)*, Mar. 2014, pp. 43-52.

[27] C.-C. Chen, X. Xu, R.-L. Liao, W.-H. Peng, S. Liu and S. Lei, “Screen content coding using non-square intra block copy for HEVC,” in *Proc IEEE Int. Conf. Multimedia Expo (ICME)*, July 2014, pp. 1-6.

[28] L. Zhao, K. Zhou, S.Wang and T. Lin, “Non-CE3: improvement on intra string copy,” *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-T0139, Feb. 2015.

[29] L. Zhao, T. Lin, K. Zhou, S. Wang and X. Chen, “Pseudo 2D string matching technique for high efficiency screen content coding,” *IEEE Trans. Multimedia,* to be published.

[30] T.-S. Chang, C.-C. Chen, R.-L. Liao, C.-W. Kuo and W.-H. Peng, “Intra line copy for HEVC screen content coding,” in *Proc Asia-Pacific Signal and Inform. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, Dec. 2014, pp. 1-8.

[31] C.-C. Chen and W.-H. Peng, “Intra line copy for HEVC screen content intra-picture prediction,” *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[32] Y. Shen, J. Li, Z. Zhu and Y. Song, “Classification-based adaptive compression method for computer screen image,” in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, July 2012, pp.7-12.

[33] C. Lan, J. Xu and F. Wu, “Compression of compound images by combining several strategies,” in *Proc. IEEE Int. Workshop Multimedia Signal Process. (MMSP)*, Oct. 2011, pp. 1-6.

[34] Z. Ma, W. Wang, M. Xu and H. Yu, “Advanced screen content coding using color table and index map,” *IEEE Trans. Image Process.*, vol. 23, no. 10, pp. 4399-4412, Oct. 2014.

[35] M. Xu, Z. Ma, W. Wang, X. Wang and H. Yu, “Low-complexity dictionary based lossless screen content coding,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 3200-3203.

[36] Y. Xu, W. Huang, W. Wang, F. Duanmu and Z. Ma, “2-D index map coding for HEVC screen content compression,” in *Proc. Data Compression Conf. (DCC)*, Apr. 2015, pp. 263-272.

[37] *Information technology - computer graphics and image processing - Portable Network Graphics (PNG): functional specification*, ISO/IEC 15948:2003 (E), Nov. 2003.

[38] Z. Pan, H. Shen, Y. Lu and S. Li, “Browser-friendly hybrid codec for compound image compression,” in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2011, pp. 101-104.

[39] S. Dikbas and F. Zhai, “Lossless image compression using adjustable fractional line-buffer,” *Signal Process.: Image Commun.*, vol. 25, no. 5, pp. 345-351, June 2010.

[40] Y. Jiang, Y. Li, D. Ban and Y. Xu, “Frame buffer compression without color information loss,” in *Proc. IEEE Int. Conf. Comput. Inform. Technol. (ICCIT)*, Oct. 2012, pp. 12-17.

[41] H.-C. Kuo and Y.-L. Lin, “A hybrid algorithm for effective lossless compression of video display frames,” *IEEE Trans. Multimedia*, vol. 14, no. 3, pp. 500-509, June 2012.

[42] J. Lei, X. Zou, Z. Wu and W. Fan, “Research of an image map encoding algorithm on frame buffer,” in *Proc. Int. Conf. ASIC (ASICON)*, Oct. 2007, pp. 894-897.

[43] Y. Li, Y. Jiang and H. Meng, “Adaptive pixel encoding: an effective algorithm for frame buffer compression,” in *Proc. IEEE Int. Conf. Compu. Inform. Technol. (ICCIT)*, Oct. 2012, pp. 5-11.

[44] Y. Li, W. Wang and G. Zhang, “Hybrid pixel encoding: an effective display frame compression algorithm for HD video decoder,” in *Proc. IEEE Int. Conf. Comput. Sci. Eng. (CSE)*, Dec. 2012, pp.303-309.

[45] T. B. Y. Liew, B. G. Lee and H. Yoo, “A low complexity and lossless frame memory compression for display devices,” *IEEE Trans. Consum. Electron.*, vol. 54, no. 3, pp.1453-1458, Aug. 2008.

[46] H. Shim, N. Chang and M. Pedram, “A compressed frame buffer to reduce display power consumption in mobile systems,” in *Proc. Asia South Pacific Design Automat. Conf. (ASP-DAC)*, Jan. 2004, pp. 819-824.

[47] H.-T. Yang, J.-W. Chen, H.-C. Kuo and Y.-L. Lin, “An effective dictionary-based display frame compressor,” in *Proc. IEEE/ACM/IFIP Workshop Embedded Syst. Real-Time Multimedia (ESTIMedia)*, Oct. 2009, pp. 28-34.

[48] J.-W. Han, M.-C. Hwang and S.-J. Ko, “Vector quantizer based block truncation coding for color image compression in LCD overdrive,” *IEEE Trans. Consum. Electron.*, vol. 54, no. 4, pp. 1839-1845, Nov. 2008.

[49] J. Park and S. Lee, “Structural similarity based image compression for LCD overdrive,” *IEEE Trans. Consum. Electron.*, vol. 58, no. 4, pp.1276-1284, May 2012.

[50] J. Wang and J. W. Chong, “Adaptive multi-level block truncation coding for frame memory reduction in LCD overdrive,” *IEEE Trans. Consum. Electron.*, vol. 56, no. 2, pp. 1130-1136, May 2010.

[51] J. Wang and J. W. Chong, “High performance overdrive using improved motion adaptive codec in LCD,” *IEEE Trans. Consum. Electron.*, vol. 55, no. 1, pp. 20-26, Feb. 2009.

[52] E. J. Delp and O. R. Mitchell, “Image compression using block truncation coding,” *IEEE Trans. Commun.*, vol. 27, no. 9, pp. 1335-1342, Sep. 1979.

[53] J. Wang, K. Min and J. W. Chong, “A hybrid image coding in overdrive for motion blur reduction in LCD,” in *Proc. Int. Comput. Entertainment Comput. (ICEC)*, Sep. 2007, pp. 263-270.

[54] L. Guo, D. Zhou and S. Goto, “A new reference frame recompression algorithm and its VLSI architecture for UHDTV video codec,” *IEEE Trans. Multimedia*, vol. 16, no. 8, pp. 2323-2332, Aug. 2014.

[55] X. Lian, Z. Liu, W. Zhou and Z. Duan, “Lossless frame memory compression using pixel-grain prediction and dynamic order entropy coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 223-235, Aug. 2015.

[56] D. Zhou, L. Guo, J. Zhou and S. Goto, “Reducing power consumption of HEVC codec with lossless reference frame recompression,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 2120-2124.

[57] S. Liu, X. Xu, S. Lei and K. Jou, “Overview of HEVC extensions on screen content coding,” *APSIPA Trans. Signal Inform. Process.*, vol. 4, e10, Sep. 2015.

[58] G. J. Sullivan, J. M. Boyce, Y. Chen, J.-R. Ohm, C. A. Segall and A. Vetro, “Standardized extensions of high efficiency video coding (HEVC),” *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp.1001-1016, Dec. 2013.

[59] J. Xu, R. Joshi and R. A. Cohen, “Overview of the emerging HEVC screen content coding extension”, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 50-62, Sep. 2015.

[60] C. Lan, J. Xu, F. Wu and G. J. Sullivan, “Screen content coding,” *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-B084, July 2010.

[61] “Joint call for proposals for coding of screen content,” *ISO/IEC JTC1/SC29/WG11,* N14175, Jan. 2014.

[62] “Requirements for an extension of HEVC for coding of screen content,” *ISO/IEC JTC1/SC29/WG11,* N14174, Jan. 2014.

[63] T.-S. Chang, R.-L. Liao, C.-C. Chen, W.-H. Peng, H.-M. Hang, C.-L. Lin, F.-D. Jou, C. Pang, J. Sole, L. Guo and M. Karczewicz, “RCE3: results of subtest D.2 on Nx2N/2NxN/NxN intra block copy,” *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-P0180, Jan. 2014.

[64] B. Li and J. Xu, “On WPP with palette mode and intra BC mode,” *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-S0088, Oct. 2014.

[65] K. Rapaka, V. Seregin, C. Pang and M. Karczewicz, “On parallel processing capability of intra block copy,” *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-S0220, Oct. 2014.

[66] L. Zhang, J. Chen, J. Sole, M. Karczewicz, X. Xiu and J. Xu, “Adaptive color-space transform for HEVC screen content coding,” in *Proc. Data Compression Conf.* (DCC), Apr. 2015, pp. 233-242.

[67] H. S. Malvar and G. J. Sullivan, “YCoCg-R: a color space with RGB reversibility and low dynamic range,” *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JVT-I014, July 2003.

[68] H. S. Malvar, G. J. Sullivan, and S. Srinivasan, “Lifting-based reversible color transformations for image compression”, in *Proc. SPIE Appl. Digital Image Process.*, Aug. 2008.

[69] A. Khairat, T. Nguyen, M. Siekmann, D. Marpe and T. Wiegand, “Adaptive cross-component prediction for 4:4:4 high efficiency video coding,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 3734-3738.

[70] W. Pu, W.-S. Kim, J. Chen, J. Sole and M. Karczewicz, “RCE1: descriptions and results for experiments 1, 2, 3, and 4,” *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-O0202, Oct. 2013.

[71] J. Kang, R. Joshi, J. Sole and M. Karcziwicz, “Explicit residual DPCM for screen content coding,” in *Proc. IEEE Int. Symp. Consum. Electron. (ISCE)*, June 2014, pp. 1-2.

[72] M. Naccari, S. G. Blasi, M. Mrak and E. Izquierdo, “Improving inter prediction in HEVC with residual DPCM for lossless screen content coding,” in *Proc. Picture Coding Symp. (PCS)*, Dec. 2013, pp. 361-364.

[73] D. He, J. Wang and G. Martin-Cocher, “Rotation of residual block for transform skipping,” *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-J0093, July 2012.

[74] J. Sole, R. Joshi and M. Karczewicz, “AHG8: residue rotation and significance map context for screen content coding,” *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-M0333, Apr. 2013.

[75] B. Li, J. Xu, G. J. Sullivan, Y. Zhou and B. Lin, “Adaptive motion vector resolution for screen content,” *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-S0085, Oct. 2014.

[76] *High Efficiency Video Coding*, ISO/IEC 23008-2 and ITU-T Rec. H.265, Apr. 2015.

[77] *Advanced Video Coding for Generic Audiovisual Services*, ISO/IEC 14496-10 (MPEG-4 AVC) and ITU-T Rec. H.264, Feb. 2012.

[78] H. Yu, R. Cohen, K. Rapaka and J. Xu, "Common test conditions for screen content coding," *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-U1015, June 2015.

[79] G. Bjontegaard, "Improvements of the BD-PSNR model," *ITU-T SG16 Q6*, VCEGAI11, July 2008.

[80] S. Martucci, "Reversible compression of HDTV images using median-adaptive prediction and arithmetic encoding," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 1990, pp. 1310-1313.

[81] J. Platt, B. Keely, B. Hill, B. Dresevic, C. Betrisey, D. Mitchell, J. Blinn and T. Whitted, "Displaced filtering for patterned displays," in *Proc. Soc. Inform. Display Symp.,* May 2000, pp. 296-299.

[82] J. Limb and C. Rubinstein, "On the design of quantizers for DPCM coders: a functional relationship between visibility, probability and masking," *IEEE Trans. Comm*, vol. 26, no. 5, pp. 573-578, May 1978.

[83] D. Hoffman and D. Stolitzka, "A new standard method of subjective assessment of barely visible image artifacts and a new public database," *J. Soc. Inform. Display,* vol. 22, no. 12, pp. 631-643, Dec. 2014.

[84] *VESA updates display stream compression standard to support new applications and richer display content*, Jan. 2016 [Online]. Available: http://www.vesa.org/featured-articles/vesa-updates-display-stream-compression-standard-to-support-new-applications-and-richer-display-content

[85] T. Lin, K. Zhou, L. Zhao and X. Chen, "Non-CE1: enhancement to palette coding by palette with pixel copy (PPC) coding," *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-U0116, June 2015.

[86] K.-H. Ng, L.-M. Po, K.-W. Cheung and K.-M. Wong, "Block-matching translational and rotational motion compensated prediction using interpolated reference frame," *EURASIP J. Advances in Signal Process.*, 2010:385631, Nov. 2010.

[87] Z. Zhang and V. Sze, "Rotate intra block copy for still image coding," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 4102-4106.

[88] T. Vermeir, J. Slowack, R. Van Belle, S. Van Leuven, G. Van Wallendael, J. De Cock, and R. Van de Walle, "Guided chroma reconstruction for screen content coding," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[89] H. Yang, W. Lin and C. Deng, "Learning based screen image compression," in *Proc. IEEE Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2012, pp. 77-82.

[90] W. Zhu, W. Ding, J. Xu, Y. Shi and B. Yin, "Hash-based block matching for screen content coding," *IEEE Trans. Multimedia*, vol. 17, no. 7, pp. 935-944, July 2015.

[91] R. Joshi, J. Xu, R. Cohen, S. Liu and Y. Ye, "Screen content coding test model 5 encoder description (SCM 5)," *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCTVC-V1014*, Oct. 2015.

[92] T. Zhang, X. Guo, Y. Lu, S. Li, S. Ma and D. Zhao, "Arbitrary-size motion detection in screen video coding," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2013, pp. 1943-1947.

[93] D. Kwon and B. Madhukar, "Fast intra block copy (IntraBC) search for HEVC screen content coding," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, June 2014, pp. 9-12.

[94] G. Laroche, G. Malard, T. Poirier, C. Gisquet and P. Onno, "Non-CE2 encoder improvements on IBC search," *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11,* JCTVC-T0050, Jan. 2015.

[95] G. Laroche, T. Poirier, C. Gisquet and P. Onno, "Non-CE2: IBC encoder improvements for SCM2.0," *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCTVC-S0065, Oct. 2014.

[96] Y. Guo, B. Li, S. Sun and J. Xu, "Rate control for screen content coding in HEVC," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2015, pp. 1118-1121.

[97] H. Zhang, O. C. Au, Y. Shi, X. Zhang, K. Tang and Y. Guo, "HEVC-based adaptive quantization for screen content by detecting low contrast edge regions," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2013, pp. 49-52.

[98] H. Yang, Y. Fang, W. Lin and Z. Wang, "Subjective quality assessment of screen content images," in *Proc. Int. Workshop Quality Multimedia Experience (QoMEX)*, Sep. 2014, pp. 257-262.

[99] S. Shi, X. Zhang, S. Wang, R. Xiong and S. Ma, "Study on subjective quality assessment of screen content images," in *Proc. Picture Coding Symp. (PCS)*, June 2015, pp. 75-79.

[100] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 430-444, Feb. 2006.

[101] Y. Lin, S. Wang, K. Zhou and Q. Shen, "AHG8: quantitative measurement for subjectively lossless screen content coding," *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11,* JCTVC-M0348, Apr. 2013.

[102] *Hardent to showcase the industry's first VESA DSC encoder/decoder IPs at CES 2015*, Dec. 2014 [Online]. Available: http://www.hardent.com/electronic-FPGA-design-consulting-services/hardent-to-showcase-the-industrys-first-vesa-dsc-encoderdecoder-ips-at-ces-2015/

[103] *Unleash the potential of your gaming business*, 2015 [Online]. Available: http://www.ubitus.net/en/ugamecloud.html

[104] *Wi-Fi CERTIFIED Miracast*, 2016 [Online]. Available: http://www.wi-fi.org/discover-wi-fi/wi-fi -certified-miracast

[105] T. Lin, K. Zhou and S. Wang, "Cloudlet-screen computing: a client-server architecture with top graphics performance," *Int. J. Ad Hoc and Ubiquitous Comput.*, vol. 13, no. 1, pp. 96-108, 2013.

[106] Y. Lu, S. Li and H. Shen, "Virtualized screen: a third element for cloud-mobile convergence," *IEEE Trans. Multimedia*, vol. 18, no. 2, pp. 4-11, Feb. 2011.

[107] X. Nan, X. Guo, Y. Lu, Y. He, L. Guan, S. Li and B. Guo, "A novel cloud gaming framework using joint video and graphics streaming," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, July 2014, pp. 1-6.

[108] *Standard for Advanced Audio and Video Coding*, IEEE 1857, 2013.

[109] "Screen and mixed content coding working draft 1," AVS N2283, Apr. 2016.

[110] "Common test conditions for screen and mixed content coding," AVS N2282, Apr. 2016.

[111] J. Guo, L. Zhao, T. Lin, and H. Yu, "Response to B1002 call for test materials: five test sequences for screen content video coding," *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11,* JVET-C0044, May 2016.

[112] L. Zhao, K. Zhou, J. Guo, S. Wang, T. Lin, "Universal string matching for ultra high quality and ultra high efficiency SCC," *ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11,* JVET-B0048, Feb. 2016.

[113] "Requirements for a future video coding standard v1," *ISO/IEC JTC1/SC29/WG11*, N15340, June 2015.

[114] "Algorithm description of joint exploration test model 1 (JEM 1)," *ISO/IEC JTC1/SC29/WG11*, N15790, Oct. 2015.

[115] *VESA issues call for technology: advanced display stream compression*, Jan. 2015 [Online]. Available: http://www.vesa.org/news/vesa-issues-call-for-technology-advanced-display-stream-compression/

[116] *JPEG committee initiates standardization of JPEG XS, a low-latency lightweight image coding system*, July 2015 [Online]. Available: http://jpeg.org/items/20150709_press.html

[117] A. Willeme, A. Descampe, S. Lugan, B. Macq, "Quality and error robustness assessment of low-latency lightweight intra-frame codecs," in *Proc. Data Compression Conf. (DCC)*, Mar. 2016.

[118] *JPEG XS - Call for Proposals for a low-latency lightweight image coding system*, Mar. 2016 [Online]. Available: https://jpeg.org/items/20160311_cfp_xs.html