NEAREST-NEIGHBOR INTRA PREDICTION FOR SCREEN CONTENT VIDEO CODING

Haoming Chen*,† *, Ankur Saxena** *and Felix Fernandes**

eehmchen@uw.edu; {a.saxena1, felix.f}@samsung.com
*Samsung Information Systems America, 1301 E. Lookout Drive, Richardson, TX - 75082
†Department of Electrical Engineering, University of Washington, Seattle, WA - 98105

ABSTRACT

Screen content video coding is becoming increasingly important in various applications, such as desktop sharing, video conferencing, and remote education. In general, compared to natural camera-captured content, screen content has different characteristics, such as sharp edges. In this paper, we propose a novel intra prediction scheme for screen content video. In the proposed scheme, bilinear interpolation in angular intra prediction in HEVC is *selectively* replaced by nearest-neighbor (NN) interpolation to preserve the sharp edges in screen content video. We present two different variants of NN interpolation. In the first *implicit* pixel-based method, both the encoder, and the decoder determine whether to perform NN interpolation based on the prediction pixels. The second method comprises of the encoder performing a Rate-Distortion search at a block-level, and explicitly signaling a flag to the decoder to indicate when to use the NN interpolation. Both the proposed variants provide significant gains over HEVC, and simulation results show that average gains of 3.3% BD-bitrate are achieved for screen content video. The HEVC proposal of this method was accepted in the core experiments, and would be a technology under consideration in the ongoing Screen Content Coding extension of HEVC scheduled to begin in March 2014.

Index Terms— Intra prediction, screen content coding, nearest-neighbor interpolation, bilinear interpolation, H.265/HEVC.

1. INTRODUCTION

Screen content coding (SCC) is widely used for various applications, such as desktop sharing, video conferencing, and remote education. Even though H.265/HEVC video codec provides significant compression gains over its predecessor H.264/AVC [1], its coding performance can be further improved for SCC as the original version 1 HEVC video codec was designed primarily for natural camera-captured content video. Due to the importance of SCC and its significant different characteristics to camera-captured video content, ITU-T, and ISO/IEC MPEG have jointly issued a Call for Proposal for a SCC extension of HEVC to begin in March 2014 MPEG meeting, and standardization of SCC to happen in 2014/2015. In general, screen content video has different characteristics as compared to natural camera-captured video content, and various compression tools have been proposed for improving the coding efficiency for screen content. For example, screen content has less color intensities, and has very sharp edges. A palette mode [2] was proposed to represent the pixels in screen content with fewer values. A transform skip scheme [3] was also proposed to improve the coding performance for such video sequences. Sample adaptive prediction (SAP) was proposed for horizontal and vertical modes [4, 5] and diagonal modes [6, 7] to improve the prediction accuracy. In addition, screen content also has some repeated patterns, e.g., text letters, and numbers in an article, for example in a Word or Powerpoint file. Intra block copy framework [8] was proposed to utilize this spatial redundancy to further improve the compression efficiency of SCC. The transform skip, sample adaptive prediction and intra block copy have been adopted in the current range extension standardization for H.265/HEVC [9].

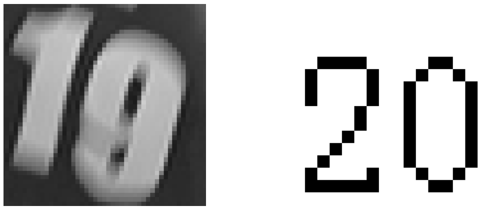


Fig. 1. (a) Natural camera-captured image (from sequence "BasketballDrillText") and (b) Screen content image (from sequence "sc_SlideShow")

In this paper, we focus on another characteristic of the screen content: sharper edges. Due to the nature of the image sensor and the effect of the camera lens, object boundaries in camera-captured content are relatively smooth. However, screen content has sharp edges. A comparison of natural and screen content is shown in Fig. 1. In current angular intra prediction scheme of HEVC, a bilinear filter is used to smooth the reference samples on the boundary, being used as predictors, to achieve a better predictor. However, for screen content video or computer-generated graphics, the content has a lot of sharp edges. In such a case, smoothing the samples actually makes the prediction inaccurate to the intensity values of the pixels being predicted. Hence, we propose a nearest neighbor (NN) intra prediction scheme in intra coding, which can derive better predictors, especially when sharp edges are present.

The rest of the paper is organized as follows: The proposed NN intra prediction scheme is presented in Sec. 2. The pixel-difference-based selection method for NN intra prediction is presented in Sec. 3. Sec. 4 presents a rate-distortion-optimization (RDO) variant for the NN intra prediction scheme. Experimental results are presented in Sec. 5, followed by conclusions in Sec. 6.

2. PROPOSED NEAREST-NEIGHBOR INTRA PREDICTION

In HEVC [10], up to 35 intra prediction modes are supported, including Planar, DC and 33 angular modes, as shown in Fig. 2. Among the various directional modes, Horizontal, Vertical and three diagonal modes (diagonal down right, diagonal down right and diagonal up right) derive the predictors directly from the integer position reference samples along the prediction direction. For example, prediction for diagonal down left mode for a 4×4 block is shown in

[*] This work was performed when H. Chen was an intern at Samsung.

978-1-4799-5751-4/14/$31.00 ©2014 IEEE

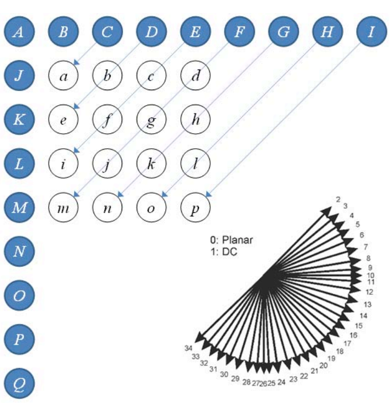
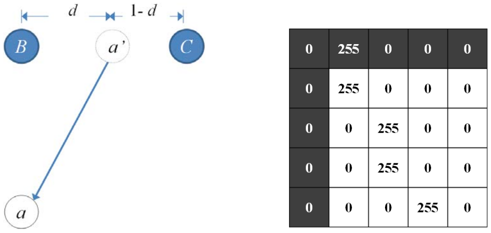


Fig. 2. Right bottom figure shows 35 intra prediction modes and orientations in HEVC. Left top figure is a 4 × 4 block example of the intra prediction process. The coding block has pixels a∼p. Pixels A∼Q are neighboring reconstructed pixels, which are used as the reference pixels. If mode 34 (diagonal down left) is selected, pixels C∼I are used to predict this block.



0	255	0	0	0
0	255	0	0	0
0	0	255	0	0
0	0	255	0	0
0	0	0	255	0

(a) (b)

Fig. 3. (a) Bilinear interpolation in HEVC intra prediction. The pixel a' is the hypothetical predictor position of pixel a. (b) A synthetic image patch with neighboring reference samples (shaded).

Fig. 2. For the oblique non-diagonal modes, integer position reference samples are not available. In this case, bilinear interpolation is applied on the neighboring two samples to obtain the hypothetical prediction position, as shown in Fig. 3a. So the predictor of pixel a is calculated as follows:

$$a' = (1 - d) \times B + d \times C; 0 < d < 1. \quad (1)$$

As described earlier in Sec. 1, for screen content video, the difference in pixel values across an object boundary could be large. In such a case, the bilinear interpolated value a' may not be close to either pixels B and C . Instead, the non-interpolated values can provide an efficient prediction closer to the actual value of the pixel a, and the energy in the residual (or absolute value of the residual $a - a'$) will be smaller. Hence, an NN-interpolated prediction can be an efficient prediction choice, i.e., the reference sample that is nearest to the hypothetical predictor location a' can be used as the predictor, instead of the interpolated value. The predictor is derived by following equation:

$$a' = B \text{ if } d < \frac{1}{2}, \text{otherwise } a' = C. \quad (2)$$

To show the effectiveness of the NN intra prediction scheme, we show a toy example in Fig. 3b where a synthetic 4 × 4 image patch (with neighboring reference samples) is shown. The original values of the 4 × 4 block, and the reconstructed pixels on the boundary are shown. The boundary pixels are used to generated the prediction for the block by using both bilinear interpolation in HEVC, and the proposed NN interpolation. For simplicity, only directional modes 18 to 26 (see Fig. 2) are tested, since other modes are obviously unsuitable for this orientation. Table 1 shows the sum of absolute difference (SAD) between the original block and the prediction block. The best SAD is achieved by the NN interpolation for mode 22 and can be 0. Such a kind of synthetic pattern is actually very common in the screen content video, though it is unlikely to appear in natural camera captured video content. Note that one may argue that there are a lot of edge preserving interpolation techniques in the literature, e.g., bilateral filter [11] and shock filter [12], and using NN interpolation is sub-optimal. However, considering the hardware complexity of the aforementioned techniques, and low computational complexity of NN interpolation, it is an attractive alternative.

To further evaluate the performance of the proposed NN intra prediction on real video sequences, we run a simulation as follows: we simply replace the bilinear interpolation with the proposed Eqn. (2) for all blocks. We encoded various video sequences (including both screen content and natural content video) as all intra frames in Table 2, since the proposed method is for intra coding. Full details about these sequences, and their GOP size, Intra period, frame rate, dimensions etc., are in [9] and [13], and described in more details in Sec. 5. The Bjøntegaard-Delta bitrate (BD-bitrate) [14] comparison of Luma component in this case is shown in Table 2 (method "All Blocks"). From the simulation results, the NN intra prediction saves the BD-bitrate up to 12.9% for the sequence "sc_cg_twist_tunnel" and 2.2% in average for screen content sequences. However, note that there is an 10.2% BD-bitrate increase for the sequence "BasketballDrillText" which is primarily natural camera captured content with a strip of text in the bottom, and significant BD-bitrate increase for almost all natural sequences, which is not desirable. The reason for such a loss is that for natural camera captured content, bilinear interpolation performs efficiently, as the pixel values are smooth.

In theory, one could signal a flag in the Sequence Parameter Set (SPS) or frame-header to indicate that a sequence (or frame) contains screen content, and consequently the entire sequence (or frame) should use NN intra prediction. Unfortunately, such a scheme would fail because video frames typically consist of mixed screen content and camera-captured content.

Hence, applying the NN intra prediction selectively based on the content would be desirable, as it works well on most screen content video. In this paper, we present the following two variants to selectively apply the proposed NN intra prediction, and describe the two variants in the following sections:

1. Pixel difference based selection criteria (at pixel level): As the reconstructed reference pixels are available at both the encoder and the decoder, an implicit criteria which can be derived based on these reference samples to determine whether to use bilinear interpolation or the NN interpolation scheme;
2. Rate-Distortion based selection criteria (at block level): A rate-distortion optimized decision is made for each coding block at the encoder, and a flag is signaled to the decoder side to select between bilinear interpolation or the NN interpolation scheme.

3. PIXEL DIFFERENCE BASED SELECTION VARIANT

In this section, we propose a simple implicit scheme that adaptively applies the NN interpolation or the bilinear interpolation, and is based on the reference pixel differences. Specifically, the proposed algorithm is described as follow:

```
if $|B - C| \geq threshold$ then
    if $(d < 1/2)$ then
        $a' = B$
```

Mode	Bilinear	Nearest Neighbor
18	1530	1530
19	1688	2040
20	1306	1530
21	668	1020
22	**608 (best)**	**0 (best)**
23	894	510
24	1210	1530
25	1402	1530
26	1530	1530

Table 1. Sum of Absolute Differences comparison of bilinear interpolation scheme in HEVC, and the proposed nearest neighbor interpolation scheme on the 4×4 block in Fig. 3b.

```
        else
            $a' = C$
        end if
    else
        retain bilinear interpolation
    end if
```

where pixels B, C, a', and distance d are as shown in Fig. 3a. In our experiments, we used the threshold as the average value of the dynamic range of the pixels. For example, for 8-bit sequences with pixel values from 0 to 255, the threshold is 128. The rationale of this scheme is that when the reference pixel values differ a lot, they typically would belong to sharp edges/regions in screen content video, and NN interpolation will perform better than bilinear interpolation. On the other hand, if the pixel difference is small, then the region is smooth, and belongs to natural camera-captured video content. In such a case, we retain the bilinear interpolation technique in HEVC. In this method, there are no overhead bits to let the decoder know about whether to use NN interpolation or not. However, the threshold decision is typically not rate-distortion optimized. We next present the Rate-Distortion method for using NN intra prediction in the next Section.

4. RATE-DISTORTION OPTIMIZED VARIANT

In the proposed Rate-Distortion based scheme, we evaluate the bilinear and NN intra prediction on one block, and then select the prediction scheme that results in less R-D cost. For different possible intra prediction modes, the same R-D search is applied, and a globally optimized combination of $\{$intra mode, interpolation$\}$ is selected. Since the intra mode information is already available at both the encoder, and decoder, we only need to encode one extra bit for the interpolation information. We further use the following techniques in the R-D search.

1. Skip the interpolation search for the Planar, DC, Horizontal, Vertical and three diagonal modes, since the integer position prediction is already applied to these modes, and bilinear interpolation and NN interpolation are identical for these modes.
2. As the intra mode information is derived prior to interpolation information in the decoder side, no signaling bits are required for describing the interpolation scheme for the Planar, DC, Horizontal, Vertical and three diagonal modes.
3. For the chroma components, the same prediction is used as corresponding luma component, when the derived mode (same as luma intra mode) is selected. Hence, no additional interpolation bit is needed for chroma.
4. Restricting the proposed intra prediction on only 4×4 blocks, as the additional advantage in terms of compression gains of applying NN intra prediction on larger blocks is marginal as we will show in Sec. 5. Note that, in general, for screen content, when there are sharp edges, 4×4 blocks are usually chosen, and hence it is sufficient to apply NN intra prediction on these smaller blocks only. Note that if an encoder does not support 4×4 blocks, then the prediction search will be required to be performed at the lowest available block size, e.g., 8×8.

5. EXPERIMENTAL RESULTS

We encoded full length sequences (which had 120 to 600 images) and various resolutions varying from 832x480 to 1920x1080 at QP's 22, 27, 32, and 37. The anchor (reference) was HM12.0+RExt4.1 [15], the HEVC reference software for developing SCC, with bilinear interpolation being used as default. Note that the state-of-the-art tools, such as transform skip, sample adaptive prediction and intra block copy are already implemented in it, and also enabled in our experimental tests. The performance of the proposed NN intra prediction scheme is evaluated for the following 3 settings: All Intra (AI), Random Access (RA), and Low-Delay (LB) settings as specified in [9] and [13]. In the AI setting, all the images were encoded as Intra, while RA setting had periodic Intra frames; and the LB settings had only the first frame as Intra. These video sequences are being tested as part of HEVC standardization. Full details about the GOP size, Intra period, coding structure of these video sequences etc. are available in [9] and [13]. Note that we present, here the results for only the evaluation of the proposed intra prediction scheme and retain all other test settings in [9] and [13]. The simulations were run on both natural content video and screen content video sequences, and the results are provided in Table 2. We present the results for following 4 different methods:

(1) NN intra prediction applied on all blocks from 4×4 to 64×64;

(2) NN intra prediction applied only on blocks of size 4×4;

(3) NN intra prediction is selected adaptively on 4×4 blocks based on the difference of the neighboring references, threshold = 128 (variant 1);

(4) NN intra prediction is selected based on the R-D search on the encoder side; one extra flag is included in the bit stream and counted into the total bits (variant 2).

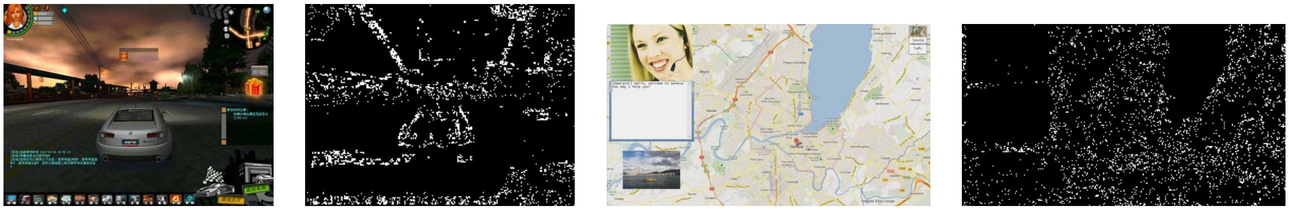
Note that some test sequences are 10-bit sequences, and to compare with the threshold we used value of 128 * 4 = 512 to appropriately normalize to the dynamic range of 10-bit video.

In Table 2, the results for the 4 above methods are presented and the BD-rate is shown for Luma component. From the results, we have the following remarks:

1. Restricting the proposed intra prediction on 4×4 achieves most of the gains in screen content video and reduces the loss in natural content video. So, we apply the "selective" intra prediction for only 4×4 blocks.
2. The threshold based pixel difference criteria works well on almost all sequences, and there are some slight losses on the sequence "EBUWaterRocksClose" since we have not optimized the threshold for each sequence separately.
3. The Rate-Distortion based scheme performs well on all sequences, and coding gains of 3.3% (AI), 2.8%(RA), and 2.7% (LB) on average are obtained. Further, there is no loss on camera-captured content.

Category	Sequences	(1) All blocks	(2) Only 4×4 blocks	(3) Pixel difference variant on only 4×4 blocks			(4) R-D based variant on only 4×4 blocks		
		AI	AI	AI	RA	LB	AI	RA	LB
Screen Content	ChinaSpeed	0.3	-0.7	-1.1	-0.6	-0.3	-2.1	-1.2	-0.8
	SlideEditing	-1.0	-0.8	-0.3	-0.3	0.2	-1.1	-1.1	-0.7
	sc_cad_waveform	-3.1	-2.8	-0.6	-1.3	-5.4	-3.1	-3.6	-7.6
	sc_pcb_layout	-7.1	-6.8	-1.0	-1.2	-0.2	-7.4	-6.5	-4.3
	sc_ppt_doc_xls	-2.9	-2.9	-1.4	-1.6	-0.9	-3.4	-3.1	-2.8
	sc_programming	1.6	-0.0	-0.5	0.1	-0.3	-1.2	-0.6	-0.4
	sc_SlideShow	4.7	2.3	0.1	-0.1	-0.3	-0.6	-0.6	-0.7
	sc_web_browing	-0.5	-1.3	-0.2	-0.2	-0.8	-1.4	-1.3	-0.5
	sc_wordEditing	-1.1	-1.0	-0.4	-0.6	-5.4	-1.5	-1.1	-0.5
	sc_twist_tunnel	-12.9	-11.6	-5.4	-5.2	-5.0	-11.5	-8.7	-8.6
	Average	**-2.2**	**-2.6**	**-1.1**	**-1.1**	**-1.3**	**-3.3**	**-2.8**	**-2.7**
Mixture of Screen and Natural Content	BasketballDrillText	10.2	4.4	0.3	0.0	0.0	-0.1	0.0	0.0
	sc_map	3.0	1.1	0.1	-0.1	0.1	-0.2	-0.3	0.0
	sc_vc_doc_sharing	-1.9	-1.7	-1.2	-1.1	-0.9	-2.4	-2.2	-0.4
	Average	**3.8**	**1.2**	**-0.3**	**-0.4**	**-0.3**	**-0.9**	**-0.8**	**-0.2**
Natural Content	Kimono	3.3	0.5	0.0	0.0	0.0	0.0	0.0	0.0
	ParkScene	2.0	0.7	0.0	0.0	0.1	0.0	0.1	0.1
	EBUHorse	3.0	1.4	0.1	0.1	0.1	0.0	-0.1	0.1
	EBUWaterRocksClose	3.2	1.9	0.5	0.5	0.5	0.0	0.0	0.0
	EBURainFruits	2.6	0.8	0.0	0.0	0.0	0.0	0.0	0.0
	Average	**2.8**	**1.1**	**0.1**	**0.1**	**0.1**	**0.0**	**0.0**	**0.0**

Table 2. BD-bitrate savings for Luma component (in %'s) when the proposed nearest neighbor intra prediction applied under difference schemes. Note that negative BD-Rate means compression gain. (Shaded number $\geq$ 1.0, which means a large loss)



(a) ChinaSpeed (b) Hit ratio (c) sc_map (d) Hit ratio

Fig. 4. First frames of two video sequences: ChinaSpeed and sc_map, and their corresponding hit ratio illustration. The white blocks denote the 4×4 blocks using NN interpolation, and the black blocks use bilinear interpolation, or are IntraBC blocks (Figure best viewed in color.)

5.1. Remarks

The additional complexity of the proposed NN intra prediction scheme at the decoder is almost negligible, as the decoder has to only check whether or not to do the NN interpolation based on the pixel difference, or a flag in the R-D variant. At the encoder, the complexity of the pixel difference based variant is again negligible, while the R-D search incurs about 8-9 % increase in the run-times for the All Intra setting. For the RA, and LB settings, the increase in encoder run-times is around 1 % or less.

Next, to illustrate that the NN intra prediction scheme is suitable in screen content video, we present the hit ratio information when it is used for two video sequences in Fig. 4. The white blocks correspond to the blocks selected as using NN intra prediction in the R-D search, and the black for bilinear interpolation, and IntraBC (Intra block copy) coding blocks. A large portion of the frames, especially the boundary, use the NN interpolation. Note that even in Intra frames, with the advent of IntraBC blocks [8], the whole block can be predicted from a different block in the frame (similar to motion estimation for Inter frames), and therefore the number of blocks using intra prediction are less. When we disabled IntraBC, the compression gains of NN intra prediction scheme increased further significantly, but we omit them due to space constraints.

6. CONCLUSION

In this paper, we have presented a nearest neighbor (NN) intra prediction scheme for screen content video coding. Our contribution is twofold. First, we demonstrate that the NN intra prediction can achieve better coding efficiency when coding screen content video, especially in the presence of sharp edges. Second, we describe two variants of when to selectively use NN interpolation or bilinear interpolation schemes: pixel-difference based, and Rate-Distortion search based. Simulation results show significant gains for the proposed NN intra prediction scheme. Future work includes finding more effective implicit criteria to selectively use the NN intra prediction scheme, and using information from already coded neighboring blocks about the interpolation scheme being used in them so as to utilize the correlation during the coding of current block.

7. REFERENCES

[1] J.-R. Ohm, G. J Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, “Comparison of the coding efficiency of video coding standards including High Efficiency Video Coding (HEVC),” *IEEE Trans. on Circuits and Syst. for Video Tech.*, vol. 22, no. 12, pp. 1669–1684, 2012.

[2] C. Lan, X. Peng, J. Xu, and F. Wu, “Intra and inter coding tools for screen contents,” *JCTVC-E145, Geneva, Switzerland*, March 2011.

[3] C. Lan, J. Xu, G. J. Sullivan, and F. Wu, “Intra transform skipping,” in *JCTVC-I0408, Geneva, Switzerland*, April 2012.

[4] M. Zhou, “Sample-based angular prediction (SAP) for HEVC lossless coding,” *JCTVC-G093, Geneva, Switzerland*, Nov. 2011.

[5] R. Joshi, J. Sole, and M. Karczewicz, “Residual DPCM for visually lossless coding,” *JCTVC-M0351, Incheon, Korea*, April 2013.

[6] H. Chen, A. Saxena, and F. Fernandes, “On sample adaptive intra prediction for oblique modes in lossless coding,” *JCTVC-N0176, Vienna, Austria*, July 2013.

[7] A. Saxena, H. Chen, and F. Fernandes, “On sample adaptive intra prediction for oblique modes in lossy coding,” *JCTVC-N0177, Vienna, Austria*, July 2013.

[8] M. Budagavi and D.-K. Kwon, “Video coding using intra motion compensation,” *JCTVC-M0350, Incheon, Korea*, April 2013.

[9] D. Flynn, K. Sharman, and C. Rosewarne, “Common test conditions and software reference configurations for HEVC range extensions,” *JCTVC-N1006, Vienna, Austria*, July 2013.

[10] G. J. Sullivan, J. Ohm, Woo-Jin Han, and T. Wiegand, “Overview of the High Efficiency Video Coding (HEVC) Standard,” *IEEE Trans. on Circuits and Syst. for Video Tech.*, vol. 22, no. 12, pp. 1649–1668, 2012.

[11] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” in *Computer Vision, 1998. Sixth International Conference on*. IEEE, 1998, pp. 839–846.

[12] S. Osher and L. Rudin, “Feature-oriented image enhancement using shock filters,” *SIAM Journal on Numerical Analysis*, vol. 27, no. 4, pp. 919–940, 1990.

[13] A. Saxena, D. Kwon, M. Naccari, and C. Pang, “HEVC Range Extensions Core Experiment 3 (RCE3): Intra Prediction techniques,” *JCTVC-N1123, Vienna, Austria*, July 2013.

[14] G. Bjøntegard, “Calculation of average psnr differences between rd-curves,” *ITU-T VCEG-M33*, 2001.

[15] JCTVC, “Hevc test model (hm),” `http://hevc.kw.bbc.co.uk/git/w/jctvc-tmuc.git/commit/15824d963c1a928b75170b784599eebf8b38a1b1`, 2013, [Online; accessed 4-Nov-2013].