Adaptive Downsampling for High-Definition Video Coding

Jie Dong, *Member, IEEE,* and Yan Ye, *Senior Member, IEEE*

***Abstract*—Previous research has shown that downsampling prior to encoding and upsampling after decoding can improve the rate-distortion (R-D) performance compared with directly coding the original video using standard technologies, e.g., JPEG and H.264/AVC, especially at low bit rates. This paper proposes a practical algorithm to find the optimal downsampling ratio that balances the distortions caused by downsampling and coding, thus achieving the overall optimal R-D performance. Given the optimal sampling ratio, dedicated filters for down- and upsampling are also designed. Simulations show this algorithm improves the R-D performance over a wide range of bit rates.**

***Index Terms*—Advanced video coding (AVC), downsampling, H.264, video coding.**

I. Introduction

In RECENT years, digital video content has enjoyed explosive popularity. In addition to professionally generated content, there is a proliferation of user-generated video materials with increased quality and spatio-temporal resolutions. These videos are distributed and transmitted over various networks, such as the Ethernet, 3G networks, and Wi-Fi. As the popularity of HD video keeps increasing, it is projected that video will make up the majority of the world's mobile data traffic in the near future. Even with recent advances in the wireless network technologies, such as 3G and LTE, the bandwidth for video transmission remains very limited. With low bit budget, the state-of-the-art video coding technology H.264/AVC often resorts to heavy compression, which causes severe blockiness, blurriness, and temporal flickering. It is anticipated that even the new generation video coding standard currently being developed, called High Efficiency Video Coding (HEVC), will be unable to provide satisfactory quality for HD videos at such low bit rates.

Previous research has shown that downsampling prior to encoding and upsampling after decoding [see Fig. 1(a)] can improve the rate-distortion (R-D) performance compared with using standard technologies, e.g., JPEG or H.264/AVC, directly on the original input, especially at low bit rates [1]–[7]. In [1] and [2], downsampling and upsampling were performed at the macroblock (MB) level. The so-called critical bit rate was studied, below which an MB is downsampled in the horizontal, vertical, or both directions with a fixed ratio of 2:1 before being coded, and upsampled to the original size after reconstruction. However, performing such operations at the MB level no longer conforms to the coding standards. To maintain conformance to the coding standards, downsampling and upsampling should be performed as preprocessing and postprocessing, respectively. The optimal downsampling ratio for low bit rate image coding was studied in [3], which analytically explained the advantage of downsampling an image prior to JPEG compression and upsampling the JPEG-decoded image, and developed a theoretical model to determine the optimal downsampling ratio. For video coding, the benefits of using this system were reported in [4], where a set of downsampling ratios were tried over a wide range of bit rates. Using the optimal ratio at a given bit rate, it is shown that 30% and 50% bit rate reduction can be achieved for H.264/AVC and MPEG-2, respectively. In [5], a downsampling-based video coding system is proposed for low bit rate applications, in which intraframes and interframes are coded at the original and 2x downsampled resolutions, respectively. After decoding, the reconstructed interframes are restored to the full resolution by an improved exampled-based super-resolution algorithm. Much other related literature in this context focus on the sampling filter design. Following [3], Tsaig *et al.* proposed an image-dependent algorithm to find optimal filters for decimation and interpolation [6]. In [7], an interpolation-dependent image downsampling method was proposed, where the difference between the input image and the output image generated by a specific interpolation method is minimized.

This paper proposes a practical algorithm to find the optimal downsampling ratio for the system in Fig. 1(a), which theoretically can be any rational number represented by A/B ($A \geq B$). The proposed algorithm decomposes the overall distortion into two types: one caused by downsampling and the other by coding, and estimates them separately. Based on these estimates, the proposed algorithm finds the optimal downsampling ratio that makes the best balance of the two distortions, thus achieving the best overall R-D performance. Given the downsampling ratio, the dedicated downsampling and upsampling filters are designed. Experimental results show that this algorithm efficiently improves the R-D performance over a wide range of bit rates.

The remainder of the paper is organized as follows. Section II presents the proposed algorithm to find the optimal ratio for video downsampling. Given the determined optimal

Manuscript received March 26, 2013; revised June 27, 2013; accepted August 5, 2013. Date of publication August 15, 2013; date of current version March 4, 2014. This paper was recommended by Associate Editor W. Zhang.

The authors are with InterDigital Communications, Inc., San Diego, CA 92121 USA (e-mail: jie.dong@interdigital.com; yan.ye@interdigital.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2013.2278146

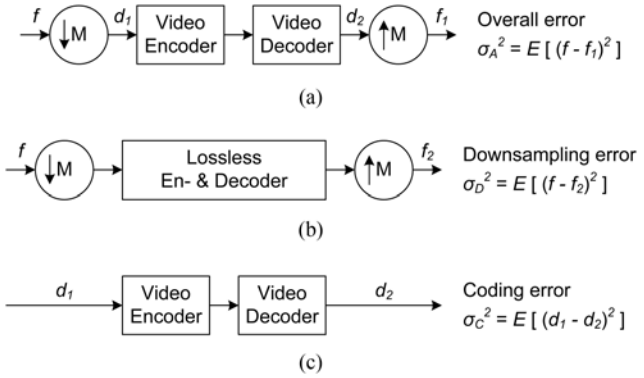


Fig. 1. (a) Coding system with down- and upsampling is decomposed to (b) the sampling part and (c) the coding part.

ratio, Section III shows how the dedicated down- and upsampling filters are designed on-the-fly. Section IV reports the experimental results, followed by the conclusion in Section V.

II. Optimal Ratio for Video Downsampling

A. Problem Statement

In the system shown in Fig. 1(a), the video input, denoted as f, is downsampled with the ratio M to generate d_1; d_1 is coded and decoded to reconstruct d_2; d_2 is upsampled with the ratio M to generate the full-resolution output video f_1. The overall MSE between f and f_1 is called overall error, denoted as σ_A^2. This system can be decomposed to the sampling part and the coding part, as shown in Fig. 1(b) and (c), respectively. In the sampling part, for the input original video f, upsampling with a factor M is applied right after downsampling to generate f_2; that is, the MSE between f and f_2 is caused only by downsampling and is called downsampling error, denoted as σ_D^2. In the coding part, the MSE between the input and output, denoted as d_1 and d_2, is caused only by coding and is denoted as the coding error σ_C^2.

When M is equal to 1.0, σ_D^2 is zero and the system in Fig. 1(a) reduces to a traditional video codec as in Fig. 1(c). Once the bit budget is given, increasing M reduces resolution for d_1 and also reduces σ_C^2, because more bits on average are allocated to code each pixel in d_1. In other words, a larger M can result in a better reconstruction in the coding part. However, an overly large M may cause too much information loss, represented by σ_D^2, in the sampling part. This information loss occurs prior to the coding process, and could outweigh the benefit of better reconstruction in the coding part. As a consequence, the overall performance of the system in Fig. 1(a) could be even worse than the traditional video codec as in Fig. 1(c). Therefore, the value of M is very critical for balancing σ_D^2 and σ_C^2, such that the best performance of the whole system can be achieved.

Given an input video signal and a bit rate, the downsampling ratio M is optimal, when the overall error σ_A^2 reaches the minimum, as

$$M = \arg\min_M \sigma_A^2 \tag{1}$$

where σ_A^2 is jointly determined by σ_D^2 and σ_C^2. The relationship among σ_A^2, σ_D^2, and σ_C^2 is as below

$$\begin{aligned}\sigma_A^2 &= E\left[(f-f_1)^2\right] \\ &= E\left[(f-f_2+f_2-f_1)^2\right] = E\left[((f-f_2)-(f_1-f_2))^2\right] \\ &= E\left[(f-f_2)^2+(f_1-f_2)^2-2(f-f_2)(f_1-f_2)\right].\end{aligned} \tag{2}$$

As $(f-f_2)$ and (f_2-f_1) are uncorrelated, $E\left[(f-f_2)(f_1-f_2)\right]$ is equal to zero. The difference between f_1 and f_2, i.e., (f_1-f_2), is the upsampled version of (d_1-d_2), and has the same energy as (d_1-d_2), because the upsampling filter gain is equal to M. Therefore, (2) can be rewritten as

$$\sigma_A^2 = E\left[(f-f_2)^2+(d_1-d_2)^2\right] = \sigma_D^2+\sigma_C^2. \tag{3}$$

Based on (3), the conclusion is that the overall error is the summation of the downsampling error and the coding error. Therefore, the optimization problem in (1) is rewritten as

$$M = \arg\min_M \left(\sigma_D^2+\sigma_C^2\right). \tag{4}$$

The estimations of σ_D^2 and σ_C^2 are introduced in Section II-B and II-C, respectively.

Usually, M is a scalar, and applies to both horizontal and vertical directions. Suppose the resolution of the input video f is $W\times H$, and the downsampled video has the resolution $(W/M)\times(H/M)$. For some decoders that can interpolate a downsampled video with nonsquare sample to the full-resolution with the correct picture aspect ratio (PAR), the horizontal and vertical ratios may be different, denoted as M_h and M_v, respectively. Then, the resolution of the downsampled video is $(W/M_h)\times(H/M_v)$. Theoretically, M can be any rational number represented by A/B $(A\geq B)$. In practice, M should satisfy the constraint that W/M, H/M, W/M_h, and H/M_v are all integers.

B. Downsampling Error Estimation

In the sampling part, f is first filtered by an anti-aliasing filter, which is a type of low-pass filters, and then decimated. The output f_2 is a blurred version of f, because f_2 no longer possesses the energy components with frequency components higher than the cut-off frequency of the anti-aliasing filter applied to f. The most straightforward and accurate way to estimate σ_D^2 is to follow the block diagram in Fig. 1(b) and calculate σ_D^2 by definition. However, the complexity of this method is too high. In this paper, σ_D^2 is estimated in the frequency domain by measuring the energy of the high frequency components that exist in f but are lost in f_2.

The energy distribution in the frequency domain is modeled by its power spectral density (PSD). As a video signal can be modeled as a wide-sense stationary random field [8] with auto-correlation $R(\tau_h, \tau_v)$, its PSD can be calculated as the 2-D DTFT of $R(\tau_h, \tau_v)$. In practice, $R(\tau_h, \tau_v)$ is an estimate based on a set of video signals and applying 2-D DTFT to the estimated $R(\tau_h, \tau_v)$ produces an estimated PSD, which may no longer be consistent with the actual PSD.

Here, PSD is estimated by the periodogram of the random field [9]

$$\hat{S}_{xx}(\omega_1, \omega_2) = \frac{1}{WH}\left|\sum_{w=0}^{W-1}\sum_{h=0}^{H-1} x[w,h]e^{-j\omega_1 w - j\omega_2 h}\right|^2 \quad (5)$$

where W and H are the width and height of the sequence f, and $x[w, h]$ is one frame in f. When the entire sequence f consists of consistent content without scene change, $\hat{S}_{xx}(\omega_1, \omega_2)$ calculated based on one typical frame $x[w, h]$, e.g., the first frame, can well represent the energy distribution of the whole sequence f. When the sequence f contains scene changes, $\hat{S}_{xx}(\omega_1, \omega_2)$ can be calculated by averaging the PSDs of frames from different scenes.

Let the ratio M be represented by A/B ($A \geq B$), then the downsampled video has the resolution $(B\mathrm{W}/A) \times (B\mathrm{H}/A)$. In other words, the proportion of the reduced resolution in either direction is equal to (1-B/A). In the frequency domain, the lost components all have high frequency, of which the proportion is also (1-B/A), if the anti-aliasing filter applied to f has a sharp cut-off frequency at $\pm(\mathrm{B}\pi/\mathrm{A})$. In this ideal case, all the high frequency components of f_2 in the bands $\left[-\pi, -\mathrm{B}\pi/\mathrm{A}\right]$ and $\left[\mathrm{B}\pi/\mathrm{A}, \pi\right]$ are lost. We can estimate the PSD of f_2, denoted as $\hat{S}_{yy}(\omega_1, \omega_2)$, from $\hat{S}_{xx}(\omega_1, \omega_2)$ by setting the values in the aforementioned bands to zero

$$\hat{S}_{yy}(\omega_1, \omega_2) = \begin{cases} \hat{S}_{xx}(\omega_1, \omega_2), & \text{if} \omega_1, \omega_2 \in \left[-\frac{B}{A}\pi, \frac{B}{A}\pi\right] \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

When the horizontal and vertical directions have different downsampling ratios, i.e., $M_h = A_h/B_h$ and $M_v = A_v/B_v$, $\hat{S}_{yy}(\omega_1, \omega_2)$ is estimated as

$$\hat{S}_{yy}(\omega_1, \omega_2) = \begin{cases} \hat{S}_{xx}(\omega_1, \omega_2), & \text{if } \omega_1 \in \left[-\frac{B_h}{A_h}\pi, \frac{B_h}{A_h}\pi\right] \,\&\omega_2 \in \left[-\frac{B_v}{A_v}\pi, \frac{B_v}{A_v}\pi\right] \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

In practice, estimating $\hat{S}_{yy}(\omega_1, \omega_2)$ as in (6) and (7) is only a close approximation of the true PSD of f_2, because it is impossible for the anti-aliasing filter to have an ideal sharp cut-off frequency response.

After estimating the PSD of f in (5) and f_2 in (6) or (7), one can calculate the downsampling error σ_D^2 by

$$\sigma_D^2 = \frac{1}{WH}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\left[\hat{S}_{xx}(\omega_1, \omega_2) - \hat{S}_{yy}(\omega_1, \omega_2)\right] d\omega_1 d\omega_2. \quad (8)$$

C. Coding Error Estimation

Given the target bit rate R, the coding error σ_C^2 is estimated by a proposed empirical R-D model, which was trained based on a large set of video sequences, including eight CIF, four WVGA, 12 720p, and 12 1080p sequences. The R-D model is shown in

$$\sigma_C^2 = \frac{\beta}{r^\alpha} \quad (9)$$

where r is the average number of bits allocated to each pixel, i.e., bits per pixel (bpp), and can be calculated as

$$r = \frac{R \times M_h \times M_v}{F \times W \times H} \quad (10)$$

where F is the frame rate for the video sequence. This model has two parameters, α and β, whose values vary according to the content, the resolution of the sequence, the encoder implementation and configurations, and so on. Given a certain encoder configured with certain settings, one may encode a given sequence at a range of bit rates $\{R_0, R_1, \ldots, R_{N-1}\}$, and then get a corresponding set of distortions $\{D_0, D_1, \ldots, D_{N-1}\}$. The target bit rates $\{R_0, R_1, \ldots, R_{N-1}\}$ are then normalized to bpp $\{r_0, r_1, \ldots, r_{N-1}\}$ using

$$r_i = \frac{R_i}{F \times W \times H} \quad (11)$$

and the corresponding distortions are normalized to MSE accordingly, denoted as $\{d_0, d_1, \ldots, d_{N-1}\}$. The pairs of normalized bit rate and distortion $[r_i, d_i]$ ($0 \leq i < N$) are plotted as an R-D curve; any commonly used numerical optimization algorithm can be used to fit the R-D curve. The optimal values of α and β are found by solving the problem in

$$\left[\alpha_{opt}, \beta_{opt}\right] = \arg\min_{\alpha,\beta}\sum_{i=0}^{N-1}\left(d_i - \frac{\beta}{r_i^\alpha}\right)^2. \quad (12)$$

Since the down-/upsampling is performed as pre-/postprocessing [Fig. 1(a)], the values of α and β, as well as the downsampling ratio, the determination of which is introduced in Section II-D, are determined once per coded video sequence. Therefore, as defined in the H.264/AVC standard, all the frames in the coded video sequence have the same resolution. If the encoder used in the system as shown in Fig. 1(a) encodes an input video into the bitstream containing only one coded video sequence, then all the determined parameters, i.e., α, β, and the optimal ratio, represent the global optimum for the entire input video. However, if the encoder can detect the scene changes in the input video and encodes each shot into its own coded video sequence, then a set of optimal parameters can be determined for each shot separately.

Fig. 2 shows the accuracy of the proposed R-D model by comparing estimated R-D curve (see the red lines) with the ground truth (see the blue lines). For *Riverbed* and *Harbor*, the curves fit each other almost perfectly. For *Basketball* and *Raven*, the estimated distortion using the proposed model deviates slightly from the actual distortion, but the estimation error is relatively minor.

Fig. 3 gives an example of R-D curve. Given the increment of bpp, denoted as Δr, σ_C^2 is better suppressed at lower bit rates than at high bit rates. This means trading larger downsampling error for smaller coding error at lower bit rates can more effectively improve the overall coding performance, which is also verified in other related work [2]–[4].

D. Optimal Ratio Determination

Given the target bit rate R, bpp increases with respect to the downsampling ratio M (10), and therefore, coding

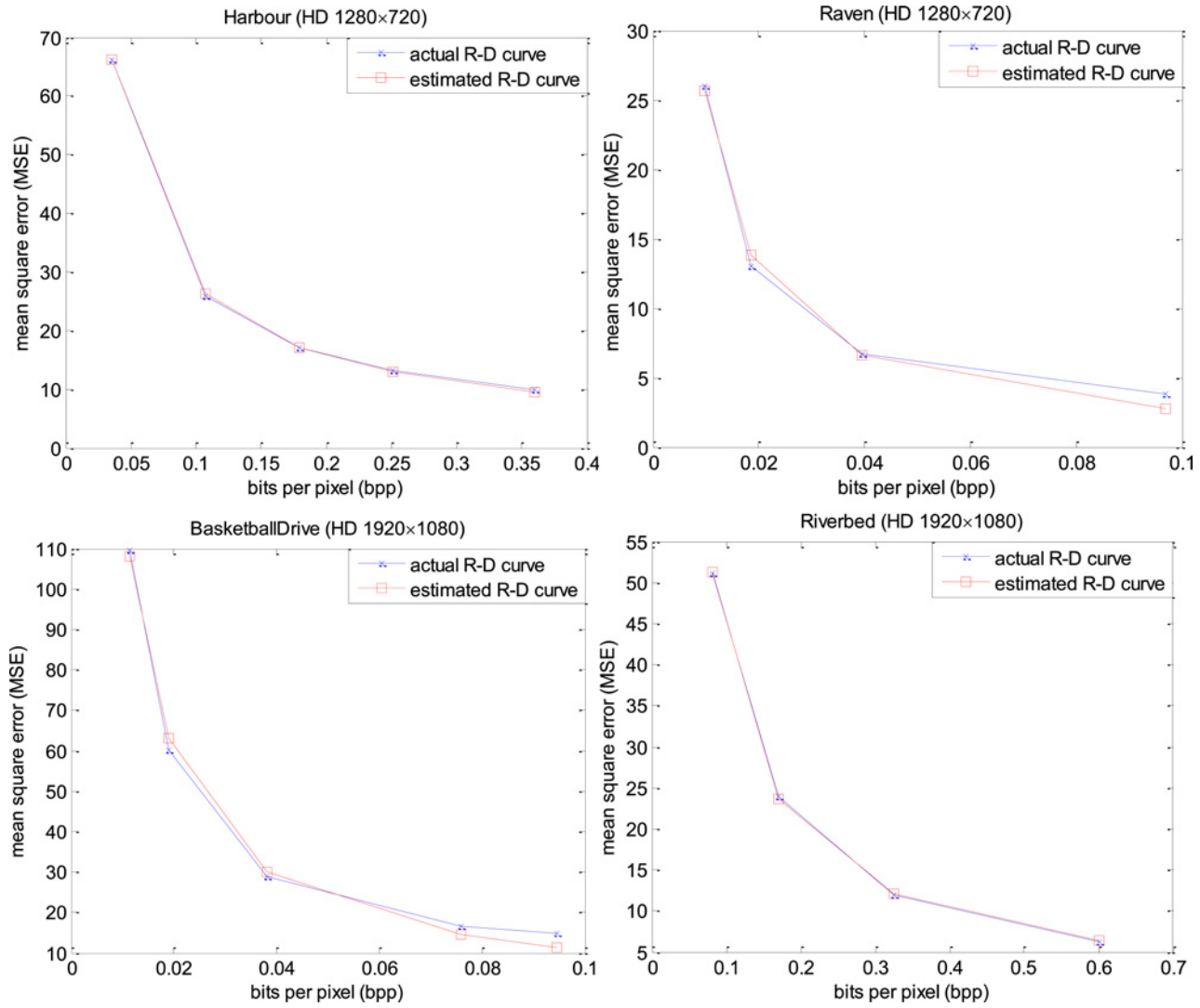


Fig. 2. Comparison of actual and estimated R-D curves.

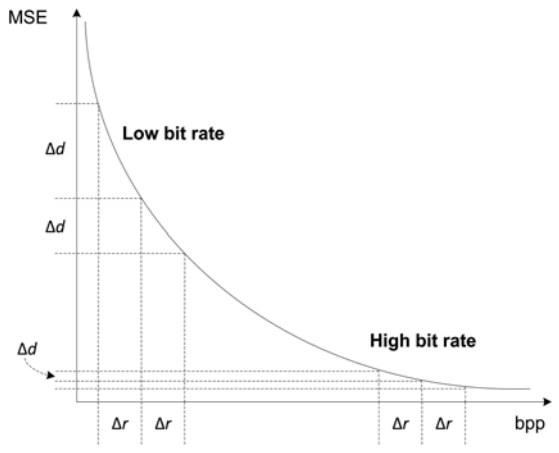


Fig. 3. Example of the R-D curve.

error σ_C^2 monotonically decreases with respect to M (9), as shown in Fig. 4(b). On the other hand, downsampling error σ_D^2 monotonically increases with M, as shown in Fig. 4(a). The optimal ratio balances the two types of distortions, such that the overall error σ_A^2 reaches the minimum. Fig. 4 shows an example of searching for the optimal ratio for the 720p sequence *Harbor*, given the target bit rate of 1 mbit/s. In search step n ($n \geq 1$), the downsampling ratio M is equal to $1+(n\text{-}1)\Delta$, where Δ is an increment of the ratio in each step, and σ_D^2 and σ_C^2 are obtained using the algorithms introduced in Sections II-B and II-C, respectively. In the first few steps, σ_D^2, increasing slowly, is well compensated by better reconstruction in the coding part. Therefore, σ_A^2 decreases. As M becomes large, the increase of σ_D^2 outweighs the decrease of σ_C^2, and σ_A^2 goes up with each search step. The optimal ratio M is determined by finding the global minimum of the overall error σ_A^2 in Fig 2 (c). For videos with different energy distributions in horizontal and vertical directions, using different downsampling ratios (M_h, M_v) in the two directions allows to apply heavier downsampling along the direction with less high frequency energy. This further improves the performance. In this case, the plots in Fig. 4 form convex surfaces, where σ_A^2 reaches the global minimum with the optimal (M_h, M_v).

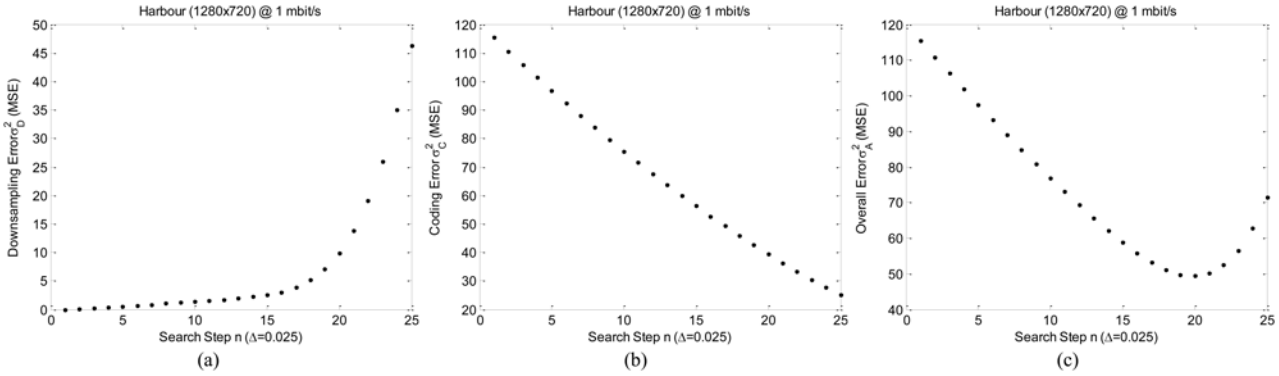


Fig. 4. Example of searching the optimal ratio for *Harbor* at 1 mbit/s. (a) Downsampling error. (b) Coding error. (c) Overall error.

III. Down- and Upsampling Filter Design

Once the optimal ratio is determined, the dedicated down- and upsampling filters are designed on-the-fly. The filters are 2-D separable, and without the loss of generality, only the derivation of horizontal filters is explained here.

Suppose the horizontal sampling ratio is A_h/B_h. In the downsampling step, the rows of the original frames are upsampled to B_h times the width by zero-insertion, filtered by the anti-aliasing filter $f_{d,h}$ in (13) with cut-off frequency $\pm(\pi/A_h)$ and filter gain B_h, and then decimated by a factor of A_h

$$f_{d,h}(n) = \frac{1}{2\pi}\int_{-\frac{\pi}{A_h}}^{\frac{\pi}{A_h}} B_h e^{jn\omega} d\omega = \frac{B_h}{A_h}\mathrm{Sinc}\left(\frac{\pi}{A_h}n\right). \quad (13)$$

In the upsampling step, the rows of the downsampled frames are upsampled to A_h times the downsampled frame width by zero-insertion, filtered by the anti-aliasing filter $f_{u,h}$ in (14) with cut-off frequency $\pm(\pi/A_h)$ and filter gain A_h, and decimated by a factor of B_h

$$f_{u,h}(n) = \frac{1}{2\pi}\int_{-\frac{\pi}{A_h}}^{\frac{\pi}{A_h}} A_h e^{jn\omega} d\omega = \mathrm{Sinc}\left(\frac{\pi}{A_h}n\right). \quad (14)$$

The filters in (13) and (14) have infinite sizes and need to be truncated by appropriate window functions before being used for interpolation. Here, a Gaussian window is used, which is empirically better than other window functions, such as rectangular, triangular, and Hanning windows. As the filter design is based on digital signal processing fundamentals, the detailed description is omitted here. Interested readers are referred to [10] for in-depth discussions.

Since the upsampling is performed during postprocessing, the optimal upsampling ratio that corresponds to the optimal donwsampling ratio determined on the encoder side may be unknown to the decoder. If the optimal upsampling ratio is transmitted to the decoder by out-of-band communication mechanism and used to design the upsampling filter as in (14), the end-to-end R-D performance is maximized. Otherwise, theupsampling ratio could be determined by relevant configurations of the end-user device, such as its display resolution.

TABLE I
Optimal Ratios Obtained by the Proposed Algorithm

HD Sequences	Bit Rate (mbit/s)	Same Ratio for Hor. & Ver.	Different Ratio for Hor.	Different Ratio for Ver.
Harbour (1280×720)	0.5	40/20	40/22	40/13
	1.0	40/21	40/23	40/15
	1.5	40/22	40/24	40/17
	2.5	40/23	40/24	40/18
	4.0	40/24	40/25	40/20
Raven (1280×720)	0.5	40/18	40/18	40/18
	1.0	40/21	40/20	40/22
	1.5	40/22	40/21	40/24
	2.5	40/24	40/22	40/28
	3.5	40/25	40/23	40/30
Cactus (1920×1080)	1.0	60/30	60/33	60/26
	2.4	60/37	60/38	60/34
	4.0	60/42	60/42	60/42
	6.0	60/47	60/45	60/52
	8.0	60/50	60/47	60/60
BasketballDrive (1920×1080)	1.0	60/28	60/26	60/31
	1.7	60/32	60/30	60/35
	3.0	60/40	60/36	60/42
	5.0	60/44	60/42	60/48
	7.0	60/48	60/44	60/55
Riverbed (1920×1080)	4.0	60/24	60/23	60/45
	7.0	60/28	60/18	60/36
	10.0	60/31	60/19	60/39
	14.0	60/33	60/21	60/42
	18.0	60/36	60/23	60/45
WalkingCouple (1920×1080)	4.0	60/34	60/26	60/40
	6.0	60/37	60/28	60/43
	10.0	60/44	60/31	60/50
	14.0	60/48	60/34	60/53
	18.0	60/53	60/36	60/60

IV. Experimental Results

Our simulations are based on six HD video sequences, each coded at five target bit rates (see Table I) with four schemes: 1) H.264/AVC, 2) fixed 2:1 downsampling, 3) the proposed adaptive downsampling (ADS) algorithm with the same horizontal and vertical ratio, and 4) the proposed ADS algorithm

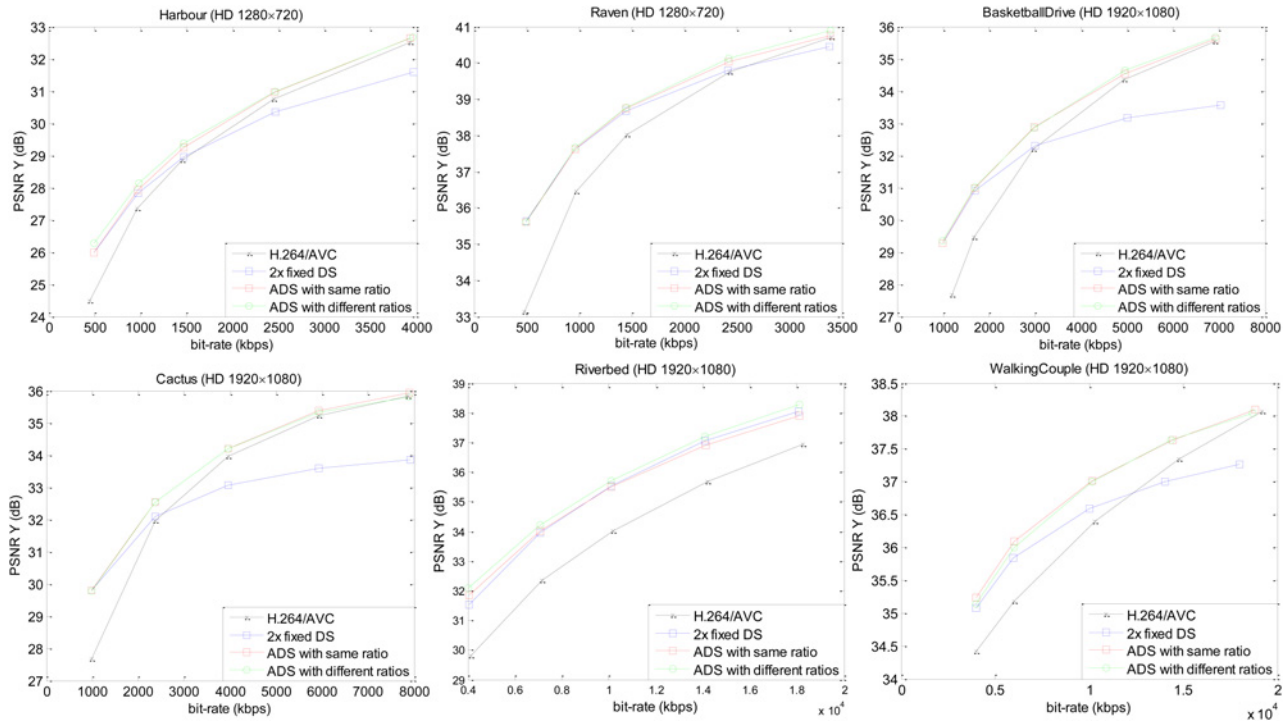


Fig. 5. Operational R-D curves provided by H.264/AVC, fixed 2:1 downsampling, proposed ADS with same ratio in each dimension, and proposed ADS with different ratios in each dimension.

with different horizontal and vertical ratios. The highest bit rate of the five target rates is selected for each test sequence such that it is either higher than the rate for which the proposed ADS algorithm can no longer outperform the regular H.264/AVC coding scheme, or 18 mbit/s, whichever is lower. Based on the highest bit rate thus selected, four additional bit rates are selected to be able to plot the R-D curves with five rate points. To show the robustness of the algorithm, the test sequences are selected to have diverse content, including rapid and irregular motions (*Cactus*, *BasketballDrive*, and *Riverbed*), sharp edges and rich textures (*Harbor* and *WalkingCouple*), and smooth areas (*Raven*). For schemes 3) and 4), the following steps were performed:

1) We calculated the optimal downsampling ratio based on the input video and the target bit rate, as presented in Section II.
2) We downsampled the input video using the filter introduced in Section III.
3) We used the x264 implementation of H.264/AVC [11] (the default x264 settings are used) to encode and decode the downsampled video.
4) We upsampled the decoded video using the filter introduced in Section III.
5) We calculated the PSNR values by comparing the upsampled decoded video [f_1 in Fig. 1(a)] to the original video [f in Fig. 1 (a)].

Fig. 5 shows the operational R-D curves for the six test sequences coded by four schemes: H.264/AVC (black line), fixed 2:1 downsampling (blue line), ADS with the same ratio for the horizontal and vertical directions (red line), and ADS with different optimal ratios for the two directions (green line). For all the sequences, the two ADS schemes significantly outperform H.264/AVC, not only at low bit rate as shown by the previous research, but also at relatively high bit rates. The gains over the regular H.264/AVC scheme are up to 2.5 dB. Compared with the fixed 2:1 downsampling scheme, which outperforms the regular H.264/AVC coding at low bit rate but can incur significant performance loss at medium to high bit rates (mostly due to overly aggressive downsampling), the proposed ADS schemes can consistently outperform the regular coding over a much wider range of bit rates. And as we expected, Fig. 5 shows that, as the optimal ratio calculated by the proposed ADS schemes approaches 1:1, the R-D performance of the proposed ADS schemes approaches that of the regular coding at high bit rates. We also applied the ADS schemes to CIF and WQVGA videos, which have less spatial correlation among pixels. We found that, for these lower resolution sequences, even mild downsampling ratios could cause too much downsampling error to be compensated by better reconstruction during coding, and that the optimal ratio eventually selected by the ADS algorithm was very close or even equal to 1:1. Therefore, for lower resolution videos, ADS cannot improve the performance as significantly as for HD videos.

As shown in Fig. 5, among the six test sequences, *Riverbed* is the only one for which the ADS scheme outperforms the regular coding consistently over a wide range of bit rates, and for which no tendency of narrower performance gap is observed at higher bit rates. After a deeper study of *Riverbed*, we found that *Riverbed* has very few high frequency components, and therefore is resistant to downsampling. The

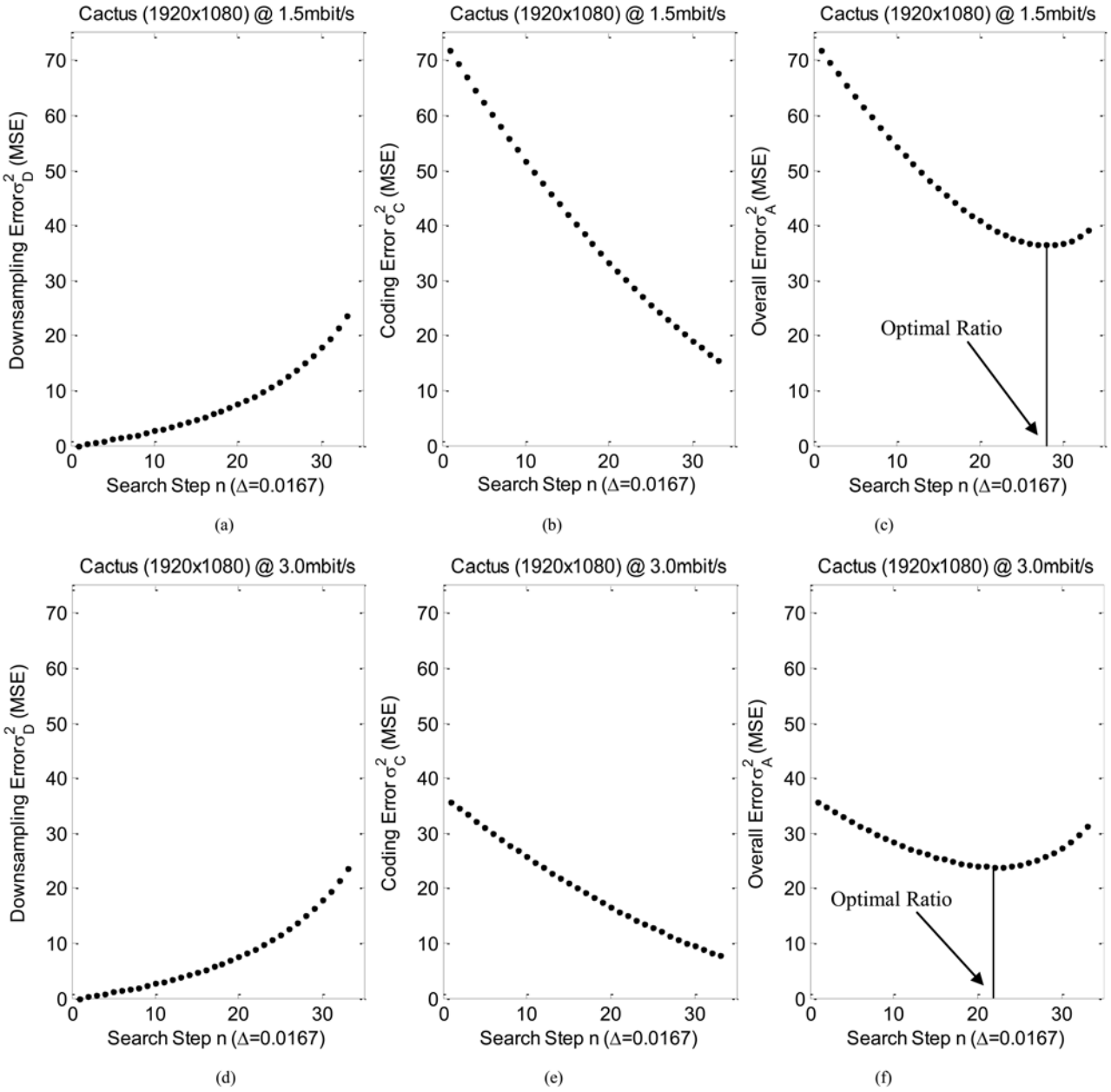


Fig. 6. Search for the optimal ratio for *Cactus*. (a) Downsampling error at 1.5 mbit/s. (b) Coding error at 1.5 mbit/s. (c) Overall error at 1.5 mbit/s. (d) Downsampling error at 3.0 mbit/s. (e) Coding error at 3.0 mbit/s. (f) Overall error at 3.0 mbit/s.

downsampling error of *Riverbed* in the form of MSE is only 4.3 when the downsampling ratio is 2:1 for each dimension, and increases very gradually until the ratio increases to 6:1. Therefore, *Riverbed* prefers more severe downsampling, where much better reconstruction during coding can be achieved, and significantly outweighs the mild downsampling error.

Table I shows the optimal ratios selected by the proposed algorithm to generate the results in Fig. 5. As can be seen, the optimal ratio depends on the bit rate and the video content. The proposed algorithm selects larger downsampling ratios at lower bit rates than it does at higher bit rates. The reason is explained here. Given a fixed bpp increment of Δr, or according to (10), equivalently a fixed increment of downsampling ratio of Δ, the coding error σ_C^2 decreases faster (that is, at steeper slope) at lower bit rates than at high bit rates. This behavior of the σ_C^2 curve can be observed in Fig. 3, as well as by comparing Fig. 6(b) and (e). Though the shape of σ_C^2 curve depends on the target bit rate, the shape of the σ_D^2 curve, as shown in Fig. 6(a) and (d), is not influenced by change in bit rates. Instead, it is determined only by the video content. Therefore, when adding σ_D^2 and σ_C^2 together to calculate the overall error σ_A^2 [see Fig. 6(c) and (f)], one needs more steps to reach the minimum of σ_A^2 at lower bit rate [Fig. 6(c) with a target rate of 1.5 mbit/s] than at higher bit rate [Fig. 6(f) with a target rate of 3mbit/s]. Consequently, the optimal downsampling ratio is larger for target rate of 1.5 mbit/s than for target rate of 3 mbit/s.

Compared with ADS with the same horizontal and vertical ratio, ADS with different ratios jointly optimizes the ratios for the two directions and, therefore, can further improve the

Fig. 7. Images cropped from 1080*p* sequence *BasketballDrive*. (a) By ADS with the same horizontal and vertical ratio (PSNR Y 31.90 dB). (b) By H.264/AVC (PSNR Y 30.73 dB). (c) Original image.

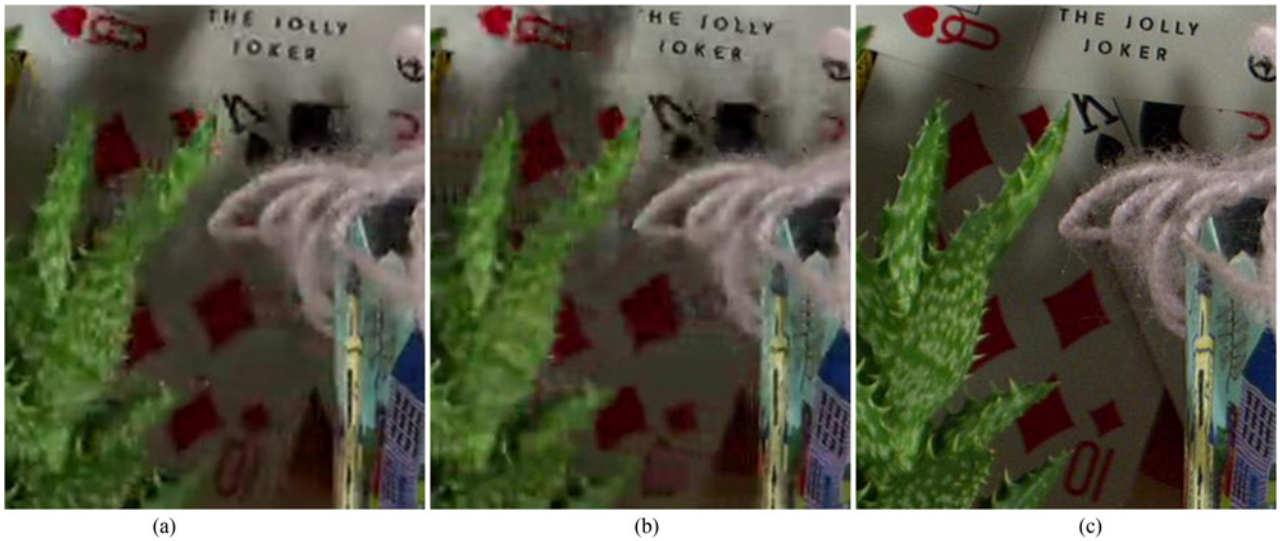


Fig. 8. Images cropped from 1080*p* sequence *Cactus*. (a) By ADS with the same horizontal and vertical ratio (PSNR Y 30.70 dB). (b) By H.264/AVC (PSNR Y 29.55 dB). (c) Original image.

overall R-D performance. For some cases (e.g., *Raven*), the improvement is negligible, because the energy distributions in the two directions are similar and the horizontal and vertical ratios selected by the ADS algorithm are also similar. For other cases (e.g., *Harbor* and *Riverbed*), the improvements are more remarkable. For example, in *Harbor*, vertical edges (masts) dominate the content. Since correlation in the vertical direction is higher than that in the horizontal direction, a bigger downsampling ratio can be applied in the vertical direction without causing much information loss. In contrast, *Riverbed* is dominated by horizontal textures (waves); in this case, a bigger downsampling ratio can be applied to the horizontal direction. Note that for some cases (*WalkingCouple* and *Cactus*), ADS with different ratios slightly underperforms ADS with the same ratio. That is likely due to the fact that, instead of brute-force search, we employed a fast search algorithm to find the optimal downsampling ratio in each dimension. As a result, the search results may have minor error compared with the actual optimum.

In addition to the improvement in objective performance, visual quality is also improved significantly by ADS. When ADS is applied, coding artifacts usually caused by heavy compression, such as blockiness and blurriness, are greatly alleviated. Figs. 7–9 show three sets of images cropped from the reconstructed sequences coded at 1.7 mbit/s: the first set is coded using ADS with the same horizontal and vertical ratio, the second set is coded by H.264/AVC without downsampling and upsampling, and the third set is the original images for comparison. The PSNR values of the videos in Figs. 7–9 range between 30 dB to 32 dB. As can be seen, the contours of the player's face and arm (see Fig. 7) and edges of the numbers and letters in the playing cards (see Fig. 8) are much better

Fig. 9. Images cropped from 1080*p* sequence *WalkingCouple*. (a) By ADS with the same horizontal and vertical ratio (PSNR Y 31.81 dB). (b) By H.264/AVC (PSNR Y 30.83 dB). (c) Original image.

preserved by using ADS. In Fig. 9, the image coded with ADS looks much smoother, compared with the blocky one coded by H.264/AVC directly.

V. Conclusion

This paper proposes the ADS algorithm. For a given sequence and target bit rate, it finds the optimal sampling ratio that minimizes the overall distortion by balancing the downsampling distortion and the coding distortion, thereby achieving the best R-D performance for the overall system. Simulations show the proposed ADS algorithm improves the coding efficiency of HD video over a wide range of bit rates. The proposed ADS algorithm also significantly improves the visual quality of the reconstructed video signal.

References

[1] W. Lin and L. Dong, "Adaptive downsampling to improve image compression at low bit rates," *IEEE Trans. Image Process.*, vol. 15, no. 9, pp. 2513–2521, Sep. 2006.

[2] V.-A. Nguyen, Y.-P. Tan, and W. Lin, "Adaptive downsampling/upsampling for better video compression at low bit rate," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2008, pp. 1624–1627.

[3] M. Bruckstein, M. Elad, and R. Kimmel, "Down-scaling for better transform compression," *IEEE Trans. Image Process.*, vol. 12, no. 9, pp. 1132–1144, Sep. 2003.

[4] A. Segall, M. Elad, P. Milanfar, R. Webb, and C. Fogg, "Improved high-definition video by encoding at an intermediate resolution," *Visual Comm. Image Process.*, pp. 1007–1018, Jan. 2004.

[5] M. Shen, P. Xue, and C. Wang, "Down-sampling based video coding using super-resolution technique," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 6, pp. 755–765, Jun. 2011.

[6] Y. Tsaig, M. Elad, and P. Milanfar, "Variable projection for near-optimal filtering in low bit-rate block coders," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 1, pp. 154–160, Jan. 2005.

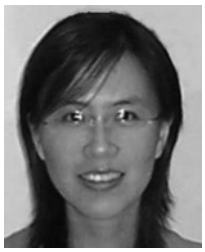
[7] Y. Zhang, D. Zhao, J. Zhang, R. Xiong, and W. Gao, "Interpolation-dependent image downsampling," *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3291–3296, Nov. 2011.

[8] W. K. Pratt, *Digital Image Processing*. New York, NY, USA: Wiley, 2001.

[9] A. Papoulis and S. U. Pillai, *Probability, Random Variables, and Stochastic Processes*, 4th ed. New York, NY, USA: McGraw-Hill, 2002.

[10] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing,* 3rd ed. Englewood Cliffs, NJ, USA: Prentice Hall, 2009.

[11] [Online]. Available: http://www.videolan.org/developers/x264.html



Jie Dong (S'07–M'10) received the B.E. and M.E. degrees in information engineering from Zhejiang University, Hangzhou, China, in 2002 and 2005, respectively, and the Ph.D. degree in electronic engineering from the Chinese University of Hong Kong, Shatin, Hong Kong, in 2009.

In 2010, she was a Post-Doctoral Research Fellow with the Chinese University of Hong Kong. In 2011, she joined InterDigital Communications, Inc., San Diego, CA, USA, as a Staff Engineer. She has been engaged in the standardization effort of HEVC and its extensions since 2011. Her research interests include HEVC and processing.

Yan Ye (M'08–SM'13) received the B.S. and M.S. degrees in electrical engineering from University of Science and Technology of China, Hefei, Anhui, China, in 1994 and 1997, respectively, and the Ph.D. degree from the Electrical and Computer Engineering Department, University of California, San Diego, CA, USA, in 2002.

She is currently with Innovation Laboratories, InterDigital Communications, Inc., San Diego, CA, USA. Previously, she was with Image Technology Research, Dolby Laboratories, Inc., Burbank, CA, and Multimedia Research and Development and Standards, Qualcomm, Inc., San Diego, CA. She has been involved in the development of various video coding standards, including the HEVC standard and its scalable extensions, the key technology area of ITU-T/VCEG, and the scalable extensions of H.264/AVC. Her research interests include video coding, processing, and streaming.