



Adaptive intra-refresh for low-delay error-resilient video coding ^{☆,☆☆}



Haoming Chen ^{a,*}, Chen Zhao ^{a,1}, Ming-Ting Sun ^a, Aaron Drake ^{b,2}

^a Department of Electrical Engineering, University of Washington, Seattle, WA 98195, USA

^b T-Mobile USA, Bellevue, WA 12920, USA

ARTICLE INFO

Article history:

Received 23 February 2015

Accepted 30 June 2015

Available online 6 July 2015

Keywords:

Error-resilience
Intra-refresh coding
Low delay
Video coding
IPPP coding
Refresh cycle-size
Refresh order
H.264

ABSTRACT

Low-delay and error-resilient video coding is critical for real-time video communication over wireless networks. Intra-refresh coding, which embeds intra coded regions into inter frames can achieve a relatively smooth bit-rate and terminate the error propagation caused by the transmission loss. In this paper, we proposed a novel linear model for the intra-refresh cycle-size selection adapting to the network packet loss rates and the motions in the video content. We also analyze issues in designing the intra-refresh coding pattern and the refresh order, and propose a strategy which can adapt to different cycle-size and obtain better R–D performance compared with traditional random intra-refresh and vertical-partition intra-refresh. Experimental results show that the linear cycle-size selection model works effectively, where a 3 dB improvement can be achieved compared with a fixed cycle-size. Also, with the proposed intra-refresh order, a 2.0% bitrate reduction is obtained in average compared with the vertical-partition intra-refresh.

© 2015 Elsevier Inc. All rights reserved.

1. Introduction

LOW-delay and error-resilient video coding is critical for real-time video chat applications over wireless networks. On the delay aspect, given a fixed network throughput, the delay induced by the video codec mostly depends on the video encoder buffer size. On the error-resilience aspect, the error induced by the packet loss will propagate to the subsequent frames due to the nature of the predictive coding framework. To terminate an error-propagation, conventional video coding schemes encode an intra-coded frame (I-frame) in each Group-Of-Pictures (GOP) [1]. An I-frame is encoded without any reference to other frames, and so, it is not affected by errors in the previous frames. For a GOP with a size of K frames, in the worst case when the beginning I-frame is lost, error propagation is limited to within the following $K - 1$ Predictive frames (P-frames), assuming for low-delay applications the B-frames (Bi-directional predicted frames) are not used.

However, for the low-delay aspect, the GOP coding structure may not be the best choice. Since an I-frame only exploits the spatial redundancy within itself, it generates much more bits than a P-frame. The resulting bit-stream usually needs to be smoothed by a relatively large encoder buffer to ensure it does not exceed the network transport capability. This rate-smoothing encoder buffer could cause relatively long delay. Since the I-frame appears periodically, this delay is induced not only at the initial stage, but also in the following video transmission.

Note that besides the insertion of I-frame, some other error-resilient schemes including robust entropy coding [2,3], forward error correction (FEC) [4] and unequal error protection [5] are proposed. In [6], some error-resilient tools used in H.263 and MPEG-4 are reviewed. A more recent review of the error-resilient coding tools is in [7]. This review paper focuses on the error-resilient intra-prediction, which is more relevant to our work. Some techniques mentioned in [7] are also discussed in this section.

Intra-refresh (or intra-slice) coding schemes can provide low-delay and good error resilience features. In intra-refresh coding, instead of encoding the whole frame as an I-frame, a subset of Macro Blocks (MBs) in each frame can be forced into intra-coded MBs, so that after a cycle of frames, the whole frame is completely refreshed. This spreading of intra MBs into the P-frames can produce a quite uniform bit-rate. With the relatively uniform bit-rate, the encoder buffer could be avoided or kept to a minimal size to result in a low-delay video codec. Besides offering

^{*} This paper has been recommended for acceptance by Yehoshua Zeevi.

^{**} Part of the work appears at the Asia–Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) 2014.

^{*} Corresponding author.

E-mail addresses: ehmchen@ee.washington.edu (H. Chen), zhaochen@pku.edu.cn (C. Zhao), sun@ee.washington.edu (M.-T. Sun), Aaron.Drake@T-Mobile.com (A. Drake).

¹ Present address: Peking University, Beijing, China.

² The views presented are as individuals and do not necessarily reflect any position of T-Mobile USA.

low-delay, intra-refresh coding schemes also provide good error-resilience performance. Since each frame is completely refreshed after a cycle of frames, the parts of the picture affected by the transmission errors will be constantly refreshed. A vertical partition intra-refresh scheme as shown in Fig. 1 is applied in x264 [8], a popular open-source software, encoding videos into the H.264/MPEG-4 AVC format. In the vertical partition intra-refresh scheme, given the intra-refresh cycle-size of N frames, the whole frame is split into N regions vertically. In the mode decision process, the MBs within each intra-refresh region (shaded area in Fig. 1) are forced to be intra-coded.

To achieve a better performance under different network conditions, a fixed cycle-size intra-refresh scheme is not optimal. In [9], a rate-distortion model is proposed considering the channel (network packet loss rates) and the source (the intra-coded MB percentage) jointly. Based on this model, given the network condition, the optimal number of intra-coded MBs can be derived. However, some empirical and sequence-dependent parameters make the derivation of the optimal intra-coded MB percentage (and thus the number of frames in an intra-refresh cycle) difficult.

Besides the selection of the number of frames in an intra-refresh cycle, intra-refresh coding can be applied to MBs in different grouping patterns, resulting in different rate-distortion performance. One method is to choose the intra-refresh MBs randomly, which is the method used in [9,10] and the default intra-refresh method in the latest JM reference software JM18.6 [11]. In the random intra-refresh, the percentage of MBs refreshed in each frame is set to $1/N$. To avoid duplicated intra-refreshing, refreshed MBs are tracked so that one MB is not refreshed twice within one intra-refresh cycle. In this way, it guarantees that all MBs are refreshed once in a cycle. A method based on rate-distortion (R-D) cost (considering packet loss rate) of each Macroblock was proposed in [12] to improve the coding efficiency. However, the random or R-D-based MB selection may cause dislocation artifacts [13] when errors occur next to the intra-coded parts. Moreover, the scattered distribution of intra-coded MBs decreases the compression efficiency due to the constrained intra prediction: to make the intra-coded region not be affected by the error propagation from the inter-coded region, the intra-prediction pixels cannot come from the neighboring inter-coded region [14]. An attention-based adaptive intra-refresh coding scheme is proposed in [13] which employs an attention area (or region of interest) extraction algorithm and applies the intra-refresh on these grouped attention MBs. This scheme shows good subjective quality of the transmitted video over an error-prone network. However, the scheme does not cover the whole picture and errors out of the attention area may propagate for a long time. An isolated-region-based method is proposed in [15]. In the high packet loss rate scenario, this method works well. However, it has lower coding performance when the packet loss rate is low, since the prediction efficiency is not considered.

Some cyclic intra-refresh algorithms that can cover the whole frame are proposed in [16–20]. To guarantee a complete and efficient error recovery, the refreshed region in a cycle cannot predict from an unrefreshed part, otherwise the error may be propagated into the refreshed part and the intra-refresh scheme becomes ineffective. This protection principle for the refreshed region restricts the potential motion compensation candidates and cause a compression efficiency drop, especially when the refresh direction (e.g., left to right in Fig. 1) is opposite to the motion direction in the frames. This restriction affects the bitrate significantly. A bad intra-refresh order could increase the total bitrate as much as 10% compared with a good intra-refresh order [16]. Considering this problem, a motion adaptive intra-refresh scheme is proposed in [16]. In this scheme, every frame is split into 3×4 partitions and the refresh order of these 12 regions is obtained by training

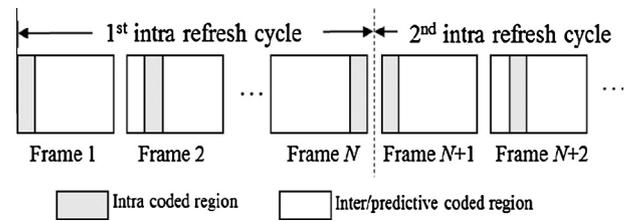


Fig. 1. Vertical partition of intra-refresh regions.

based on the motion in each partition region. This scheme can reduce the motion vector restriction effectively, but to be adaptive to the network packet loss rates, more refresh patterns and different cycle-sizes are needed.

In this paper, we propose a new intra-refresh coding scheme. In the first part, a content-based linear cycle-size selection model is proposed. Given the packet loss rate and the motion information, the best intra-refresh cycle-size is estimated. Compared with [9], our linear model is much simpler and has no empirical and sequence-dependent parameter. Also, since our model is a simple linear model, it gives better insights about how the video content affects the cycle size. It also relates the model parameters with the content and the motion of the video. All parameters in our model are trained from a set of real sequences and verified on a different set of testing sequences. In the second part, the design of intra-refresh region partition and refresh order is analyzed. Based on considerations in intra-refresh coding, we propose a rectangular partition for intra-refresh. Moreover, we modify and simplify the refresh order patterns in [16] so that they can be applied to different cycle-sizes. Experimental results show that the cycle-size model is effective and the proposed intra-refresh order outperforms the vertical cyclic intra-refresh scheme. Compared to the motion-adaptive intra-refresh scheme in [16], the proposed intra-refresh order has the flexibility that it can be applied to different cycle-sizes. The linear model in the first part is partially published in our previous work [21]. The major extension to [21] is the intra-refresh pattern design. In [21], we proposed the cycle number selection model. In this work, based on this model, we propose the intra-refresh pattern design, considering issues affecting the coding performance and the visual artifacts.

The rest of this paper is organized as follows. In Section 2, a linear model is proposed for the adaptive selection of the cycle-size. In Section 3, the design of the intra-refresh partition and the order are analyzed. Section 4 shows the simulation results. Section 5 concludes this paper and describes possible future works.

2. Adaptive content-based cycle-size selection

Fig. 2 compares different refresh cycle-sizes N ($N = 4, 8, 16$, and 32) under different packet loss rates (10^{-4} – 10^{-1}). A smaller cycle is better for recovering from the errors more quickly when the packet loss rate is high, and a large refresh cycle is better for a low packet loss rate since it gives better coding gains with a smaller intra-coded area in each frame. Thus, for optimal performance, the refresh cycle-size N should be adaptive to the packet loss rates.

In this section, we proposed a linear model to predict the optimal intra-refresh rate based on the network packet loss rate and the motion in the video content.

2.1. A linear model for the intra-refresh rate based on the network packet loss rate

In [9], a joint end-to-end distortion model is proposed w.r.t. the intra-refresh rate and the packet loss rate:

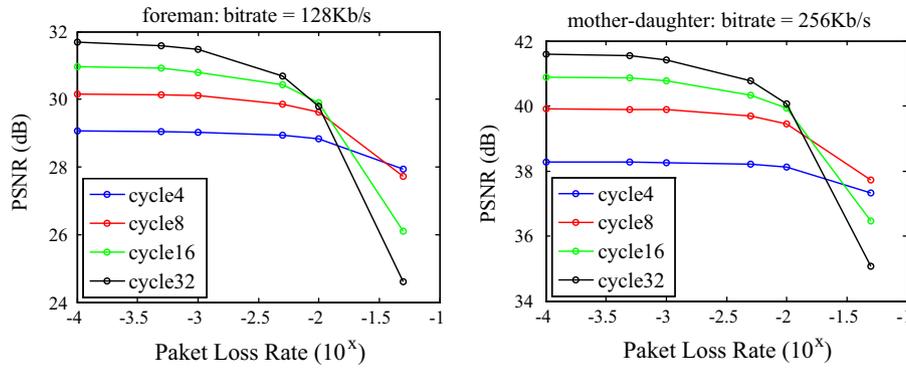


Fig. 2. Simulation results for the test sequences “foreman” and “mother–daughter” encoded by x264 with the vertical partition intra-refresh scheme [8]. The sequences are encoded with four different cycle-sizes (4, 8, 16 and 32). Each random packet loss causes a frame loss. The loss frame is concealed by repeating the previous frame.

$$D_s(R_s, \beta) = D_s(R_s, 0) + \beta(1 - \lambda + \lambda\beta) \times [D_s(R_s, 1) - D_s(R_s, 0)] \quad (1)$$

$$D_c = \frac{a}{(1 - b + b\beta)} \frac{p}{1 - p} E[F_d(n, n - 1)] \quad (2)$$

In Eq. (1), D_s is the source distortion, which is a function of the source bitrate R_s and the intra-refresh rate β (= number of intra-coded MBs/total number of MBs) in one frame. λ is a sequence-dependent parameter. The source distortion is a linear combination of two extreme cases: all MBs inter-coded $D_s(R_s, 0)$ and all MBs intra-coded $D_s(R_s, 1)$, both are measured in MSE (Mean Square Error). In Eq. (2), D_c is the channel distortion caused by the packet loss rate p . Parameter a is a constant, and b is a constant describing the motion randomness of the video scene. $E[F_d(n, n - 1)]$ is the expectation of the difference of the neighboring frames n and $n - 1$, which is also measured by the MSE. The total end-to-end distortion is the sum of the source distortion and the channel distortion:

$$D = D_s + D_c \quad (3)$$

Under a bitrate R_s and the packet loss rate p , the best intra-refresh rate that minimizes the end-to-end distortion can be derived by taking the derivative of D w.r.t. β and setting it to zero, resulting in a cubic equation of β ,

$$A\beta^3 + B\beta^2 + C\beta + D = H \frac{p}{1 - p}, \quad (4)$$

where

$$A = 2\lambda b^2, \quad (5)$$

$$B = b(b + 4\lambda - 5\lambda b), \quad (6)$$

$$C = 2(1 - b)(b + \lambda - 2\lambda b), \quad (7)$$

$$D = (1 - b)^2(1 - \lambda), \text{ and,} \quad (8)$$

$$H = \frac{abE[F_d(n, n - 1)]}{D_s(R_s, 1) - D_s(R_s, 0)}. \quad (9)$$

We observe that the intra-refresh rate β is the inverse of cycle-size N , so β is always much smaller than 1, e.g., $N = 12$ means $\beta = 0.083$. Also, the coefficients A , B , C , and D in Eqs. (5)–(9) are constants. Hence, we propose to omit the higher order items and simplify Eq. (4) into a linear model as follow:

$$\beta = H' \frac{p}{1 - p} + D', \quad (10)$$

where $H' = H/C$ and $D' = -D/C$, which are constants (dependent on the sequences). From Eq. (10) the optimal refresh rate is linear to $p/(1 - p)$.

To verify that the proposed linear model is sufficiently accurate, we test 21 video sequences as shown in Fig. 3. We encode each sequence at the 1 Mbps bitrate with the cycle-size N from 4 to

40. Note that since the model is independent to the order of the intra-refresh pattern (random, vertical, or motion adaptive in [16]), we use the random intra-refresh pattern. We simulate the packet loss rate p from 0.1% to 20%. Each packet loss scenario is simulated 50 times, the PSNRs of a decoded sequence in each simulation is denoted as $\text{PSNR}(p, N, i)$, where $i = 1, \dots, 50$. For each p , we select the N that maximizes the average PSNR, where the average PSNR³ and the best N are defined as follows:

$$\overline{\text{PSNR}}(p, N) = \frac{1}{50} \sum_{i=1}^{50} \text{PSNR}(p, N, i), \text{ and} \quad (11)$$

$$N^*(p) = \arg \max_N \overline{\text{PSNR}}(p, N). \quad (12)$$

We plot the best β (i.e., $1/N^*(p)$) versus $p/(1 - p)$ in Fig. 3. From the testing results and the fitting lines, we can see that the linear model works well.

2.2. Parameters of the linear model

With this linear model, given the parameters H' and D' we can estimate the best cycle-size easily. However, since C , D , and H depend on many empirical constants, it is difficult to get a close-form solution of H' and D' . To solve this problem, we plot the 21 corresponding fitting lines in Fig. 4. We have following observations on these lines: There is a rough relationship between the slope of the linear model and the motion of video content. For example, the ice and highway sequences have very fast motion and they have the largest slopes; container and bridge-close sequences have very slow motion and have the smallest slopes.

Actually, these observations can be justified from the expressions of C , D and H in Eqs. (7)–(9). For example, H' , which equals to H/C is directly proportional to $E[F_d(n, n - 1)]$ and inversely proportional to $D_s(R_s, 1) - D_s(R_s, 0)$. $E[F_d(n, n - 1)]$ is the difference of neighboring frames, which is large when the motion of the video content is fast. Moreover, for a fast motion video, the error propagates quickly, so a large intra-refresh rate (or a small cycle-size) is needed. For $D_s(R_s, 1) - D_s(R_s, 0)$, the distortion difference between all-intra and all-inter under the same bitrate, if it is large, it means force-intra coded MBs will induce more distortion, so a small intra-refresh rate is preferred, resulting in a smaller slope in the linear model.

Based on the above analysis, we model the slope proportional to $E[F_d(n, n - 1)]/[D_s(R_s, 1) - D_s(R_s, 0)]$. To see the proportion, in Fig. 5,

³ We note that another metric is to measure the average mean square error (MSE) first and then calculate the PSNR.

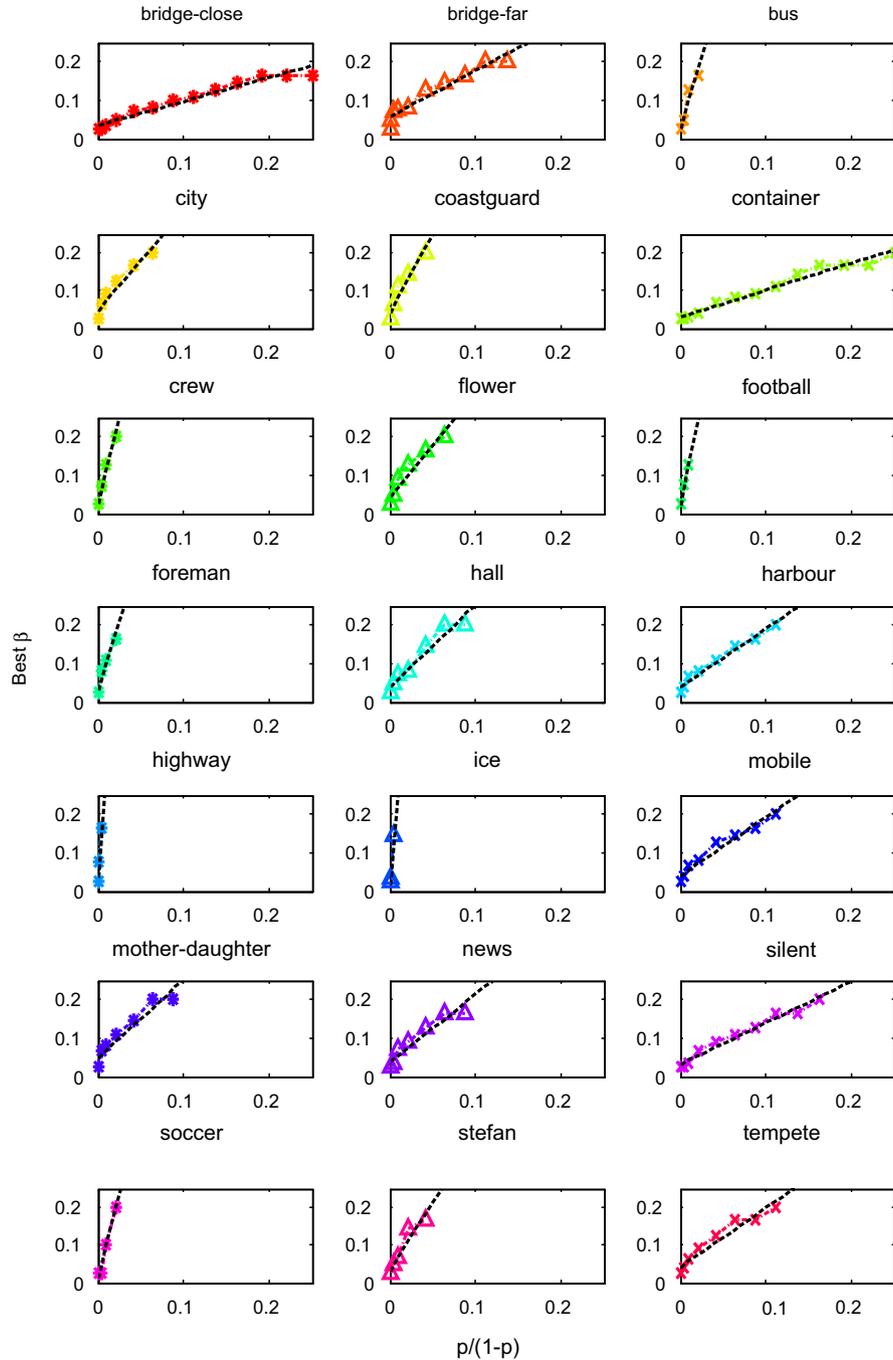


Fig. 3. Testing results on 21 sequences. The x-axis is $p/(1 - p)$, which is in the range of $(0.01, 0.25]$ and the y-axis is the best β , i.e., $1/(\text{Best Cycle-Size})$, which is in the range of $[1/40, 1/4]$. Here the “Best Cycle-Size” is the cycle-size (between 4 and 40) that minimizes the distortion for each packet loss rate. Note that for some sequences (e.g., bus, city, crew...), the best cycle-size reaches the smallest cycle-size limited to 4 ($\beta = 0.25$) when the $p/(1 - p)$ is less than 0.25, and then the line becomes flat. We have removed that flat part to show the linearity clearly. The black dashed line is the fitting linear function. We can see that the linear model can approximate the relationship between β and $p/(1 - p)$ well.

we plot the slope value H' versus $E[F_d(n, n - 1)]/[D_s(R_s, 1) - D_s(R_s, 0)]$. For each sequence, $E[F_d(n, n - 1)]$ is the average MSE of all neighboring frame pairs and the $D_s(R_s, 1) - D_s(R_s, 0)$ is the average MSE difference between two cases: all intra coding and all inter coding structures. Then, we fit the points with a linear model, which results in the following model:

$$H' = c \times E[F_d(n, n - 1)]/[D_s(R_s, 1) - D_s(R_s, 0)] + d. \quad (13)$$

To verify the model, we separate these 21 sequences into two sets: 11 sequences for training the parameters in Eq. (13) and the other 10 sequences for testing the performance of our proposed

cycle-size selection model. From the training part, the parameters $c = 0.3164$ and $d = 1.6625$.

For the offset points (D' in Eq. (10)) of each line in Fig. 4, they have very close positions, so, in the linear model, we just use the average value of the offsets in the training set as the offset of the proposed model.

2.3. Content-based adaptive cycle-size selection

Combining Eqs. (10) and (13), a selection model of cycle-size N is as follow:

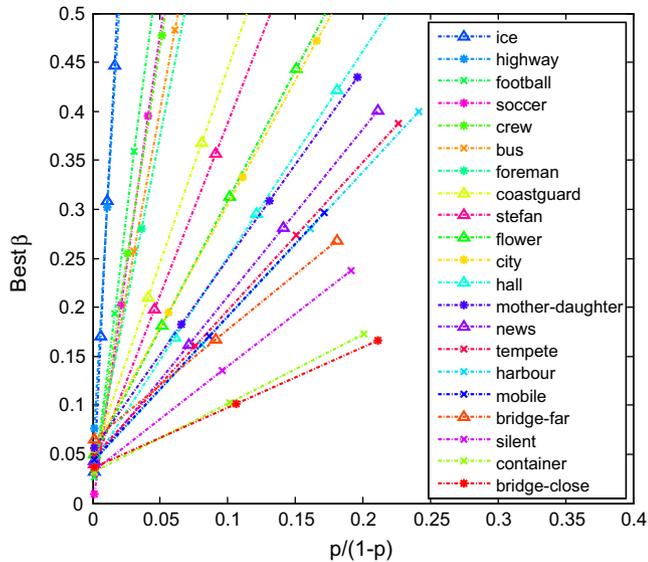


Fig. 4. Fitting lines of the 21 sequences in the same plot. (Best viewed in color.) (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$$\beta = \frac{1}{N} = \left[\frac{0.3164 \times E[F_d(n, n-1)]}{D_s(R_s, 1) - D_s(R_s, 0)} + 1.6625 \right] \times \frac{p}{1-p} + 0.0342 \quad (14)$$

However, in low-delay video coding, we cannot get the $E[F_d(n, n-1)]$, the all-intra MSE $D_s(R_s, 1)$, and all-inter MSE $D_s(R_s, 0)$ of the whole sequence. One method is to estimate these values based on the first N cycles. However, this may not work when the video content changes. Another method, which is more accurate, is to re-calculate these values every several frames and update them adaptively. In this work, for the convenience of comparing with other schemes (random intra-refresh and vertical partition intra-refresh), we use the former method.

3. Design of the MB intra-refresh region partition and order

As mentioned, the distribution and the refreshing order of intra-coded MBs could affect the R-D performance significantly.

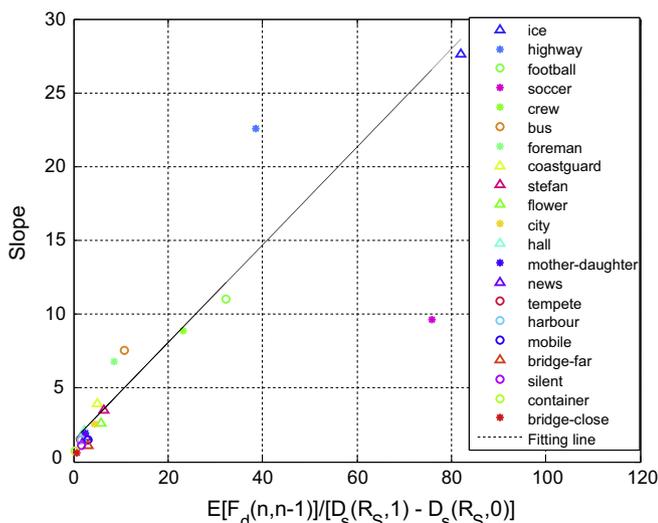


Fig. 5. The slopes of the linear models versus the proportion of $E[F_d(n, n-1)]$ and $D_s(R_s, 1) - D_s(R_s, 0)$. (Best viewed in color.) (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In this section, we discuss the design of intra-refreshing region partition and the refresh order. We consider the dislocation effect, constrained intra prediction, and constrained inter prediction in our designing process.

3.1. Consideration of issues in the intra-refresh pattern design

Some issues related to the design of intra-refresh patterns are briefly summarized as follows.

(1) The dislocation problem

Visual artifacts are caused by the scattered distribution of intra-coded MBs when packet loss occurs. To overcome this problem, in each frame, the forced intra-coded MBs should be grouped together instead of being separated. Some visual results of the dislocation problems are shown in Section 4.

(2) Constrained intra prediction

The constrained intra-prediction requires that the reference samples of the intra prediction cannot come from the neighboring inter-coded blocks. This constraint can protect the intra-refreshed region so that it is not affected by the potentially corrupted inter-coded regions. In [7], it shows that with the constrained intra prediction, the decoded video can achieve up to about 4 dB gain when the packet loss rate is between 2% and 8%. Note that the constrained intra prediction has been included in H.264/AVC standard. Scattered distribution of the intra-coded regions will decrease the compression efficiency due to the constrained intra prediction, which is an error-resilience scheme. The loss of the coding efficiency due to this constrained intra prediction is shown in Fig. 6. We tested 5 CIF sequences and a 5.1% BD-Bitrate increment is induced by the constrained intra-prediction scheme which is significant.

(3) Constrained inter prediction

The constrained inter-prediction requires that the reference samples of the inter prediction cannot come from the unrefreshed regions. As a result, the sub-optimal motion prediction could cause a coding efficiency drop. As mentioned in [16], taking the motion of the frame into consideration can reduce such kind of restriction. Note that video standards only specify those parts necessary for interworking, not those parts that only affect the quality. Since

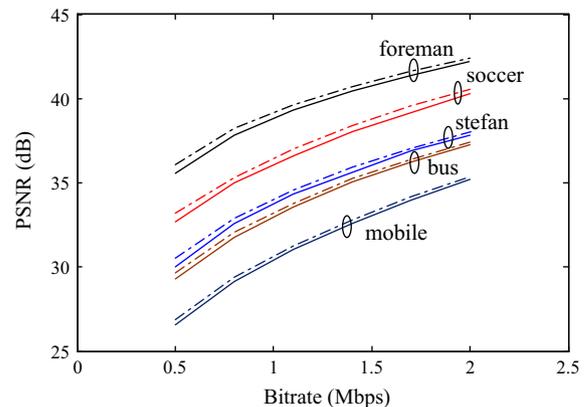


Fig. 6. Comparison of R-D performance of random intra-refresh coding schemes with and without constrained intra prediction in JM 18.6 [11]. For each R-D curve pair of 5 sequences (all are 352×288 , 4:2:0, CIF), the dashed line is the R-D curve coded without the constrained intra prediction, the solid line is the R-D curve coded with constrained intra prediction. The BD-Bitrate calculation [22] shows that constrained intra prediction will increase the bitrate by 5.1% in average.

intra-frame coding cannot refer to other frames, it is part of the H.264/AVC. The constrained inter prediction is to prevent the error from propagating from the un-refreshed part. It only affects the quality, thus, it is not part of the standard.

From the above considerations, we can see the design of intra-refresh coding is twofold: one is to design the partition method, which can reduce the dislocation problem and the constrained intra prediction problem; the other is to design the refresh order of these partitioned regions, so that the constrained motion vectors can be reduced.

3.2. Design of intra-refresh region partition for reducing the intra prediction constraint

To overcome the dislocation problem and to make full use of the intra-coded blocks in the intra prediction, grouping the intra-coded blocks together is desired. Given a number M of MBs to be intra-coded in current frame, they can be grouped in different shapes, e.g., vertical strip, horizontal strip, rectangle (if $M = X \times Y$, where X and Y are integers larger than 1), or other irregular shapes. In intra prediction, a coding block gets its prediction samples from the left and top MBs. However, due to the constrained intra prediction, coding blocks along the top and left boundaries of these grouping shapes can only have partial intra-prediction pixels, so the prediction is less efficient. It should be noted that the longer top and left boundaries, the more blocks have partial prediction pixels. It can be shown that grouping the intra-refreshed MBs into a rectangular shape as close to a square as possible could minimize the total length of the constrained boundaries, and make better use of the intra-coded pixels.

Compared with irregular shapes, grouping intra-refreshed MBs into a rectangle also has an advantage that it is easy to partition one frame into different regions without any overlaps and holes. A straightforward way is to partition a frame into $w \times h$ rectangles, where $w \times h = N$, and w and h are the numbers of partitions along the width and height of the video frame, respectively. When selecting w and h , two factors are taken into consideration:

- (i) For each N , there may exist more than one partitions. Take $N = 18$ as an example, it can be split into 18×1 , 1×18 , 6×3 , 3×6 , 2×9 , and 9×2 . Based on the grouping principle that the grouped rectangle should be as close to a square as possible, we select the partition leading to the smallest difference between N_w/w and N_h/h , where N_w and N_h are the numbers of MBs along the width and height of the video frame, respectively.
- (ii) For some prime numbers N , e.g., 11, it can only be split into $1 \times N$ and $N \times 1$ regions, which conflicts to the grouping principle that the strip shape is less efficient than the rectangle shape close to a square. In this case, we choose the nearest composite number to approximate it, which could be 10 or 12 in this case. The selection of 10 or 12 is based on the same criteria in factor (i).

3.3. Design of intra-refreshing order reducing the inter-prediction constraint

With the whole frame spilt into $w \times h$ rectangles, we design the refresh order so that the constrained inter-prediction effect can be reduced. Since the constrained inter prediction happens on the prediction from the refreshed region in the current frame to the unrefreshed region in the previous frame, we only consider the refresh regions in the current frame. For one refreshed partition region, $P_{n,i,j}$ in the n th frame of one cycle, where $n = 1, \dots, N$, $i = 1, \dots, w$, and $j = 1, \dots, h$, the cost of the constrained inter prediction is approximated by [16]

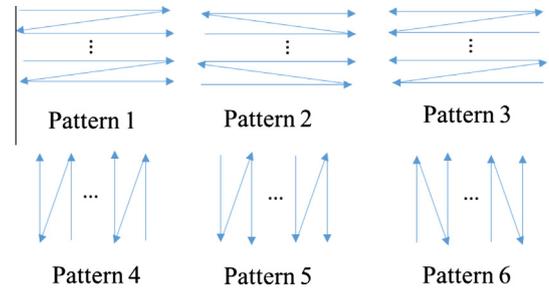


Fig. 7. Six pattern candidates we evaluated. The intra-refresh regions of frames in an intra-refresh cycle are refreshed according to a pattern shown above.

$$L_{n,i,j} = |mv_x| + |mv_y| \quad (15)$$

where mv_x is the horizontal component of the averaging motion vector on region $P_{n,i,j}$ if the horizontal component points to an unrefreshed region and similarly, mv_y is the vertical component of the averaging motion vector on region $P_{n,i,j}$ if this component points to an unrefreshed region. The total cost is the summation of the cost over all regions and all frames within one cycle,

$$L = \sum_n \sum_i \sum_j L_{n,i,j} \quad (16)$$

Since the real motion vectors are not available at the time of selecting the refreshing order of each cycle, the motion vectors of the frame previous to the current cycle is used as an approximation. However, for a cycle-size of N , there are $N!$ different kinds of refreshing orders and the searching space is so huge that we cannot afford to search every possible refreshing order to find the global optimal order leading to the smallest cost as defined in Eq. (16). In [16], 28 different patterns are proposed for the 4×3 partition and it was demonstrated that those patterns can cover most of the camera motions. In our case, since the cycle-size varies, we cannot design a pattern pool specifically for each cycle-size. Based on the simulation results, we observe that using only 6 common patterns, as shown in Fig. 7, can achieve results almost as good as the 28 different patterns used in [16]. The intra-refresh regions of frames in an intra-refresh cycle are refreshed according to one of the six patterns. We also tried other approaches such as intra-refresh a region connected to the previously refreshed region based on the smallest cost as defined in Eq. (16). However, since it could be stuck at local minimum, it is not as good as using one of the six patterns for the whole cycle. Note that these 6 patterns can be extended to different partitions easily. The comparison results are presented in Section 4. It should be noted that since the packet loss position could be random, and the regions are refreshed one by one in an intra-refresh cycle, a packet loss could occur at any position in a cycle. Thus, the optimal order of refreshing a particular MB does not depend on where the packet loss occurs, but only depends on the motion due to the constrained inter prediction.

4. Experimental results

In this section, we first test the proposed design of intra-refresh region partition and the order, and then combine it with the adaptive cycle selection model in Eq. (14). In the experiment, we compare our method with randomized intra-refresh (as default in H.264 reference software), the vertical intra-refresh (as default in x264), and the motion-adaptive intra-refresh [16]. The vertical intra-refresh is effective and is the default method in x264. The motion-adaptive intra-refresh can achieve better coding gain than the previous two. We also compare with the optimal cycle size that can achieve the best RD performance under a certain packet loss rate.

4.1. Experiments on the proposed design of intra-refreshing region partition and order

The experiment is implemented in the latest H.264/AVC reference software JM 18.6 [11]. The test conditions are as shown in Table 1.

Table 2 compares the BD-Bitrate savings (a negative number indicates bits saving) on different sequences in source coding (without packet loss) with $N = 12$ and 36. Since in this case, there is no packet loss, our proposed scheme will select a large cycle size N (in this case 36). The comparison with $N = 12$ is just to show that compared with the motion adaptive intra-refresh in [16], although our scheme uses only six refresh-order patterns, it achieves almost the same R–D performance as that in [16] which uses 28 different patterns at $N = 12$. In this table, we can see that our proposed design also outperforms the traditional vertical intra-refreshing under different cycle-sizes. The results under different packet loss rates are discussed in next part.

4.2. Experiments on adaptive cycle-size selection

In this test, we use Eq. (14) to determine the cycle-size and apply it on the testing set of 10 sequences. All sequences are encoded in 1 Mbps with rate control. The test conditions are listed in Table 3, which are different from the previous test.

The average decoded video PSNRs defined as in Eq. (11) are shown in Fig. 8, where we compare four different cases:

- (1) Optimal cycle-size that maximizes the PSNR of reconstruction video, using our proposed intra-refresh partition and order.

Table 1
Test conditions for intra-refreshing region partition and order.

Profile	Main
RD optimization	Yes
Rate control	No
Packet loss	No
Quantization parameters	22, 27, 32 and 37
Number of reference frames	1

Table 2
BD-Bitrate saving (in%) comparison.

Sequence	Our proposed method with $N = 12$		Our proposed method with $N = 36$	
	v.s. Vertical partition [8] (with $N = 12$)	v.s. Motion adaptive method in [16] (with $N = 12$)	v.s. Vertical partition [8] (with $N = 36$)	v.s. Motion adaptive method in [16] (with $N = 12$)
bridge-close	-1.5	0.1	-1.1	-17.4
bridge-far	-5.4	-0.3	-3.6	-24.7
bus	-0.6	-0.1	0.1	-9.4
city	-4.7	-0.2	-2.3	-18.6
coastguard	-1.4	0.0	-1.9	-8.8
container	-3.2	0.4	-1.6	-24.7
crew	-0.8	0.2	0.2	-4.7
flower	-4.8	0.0	-2.9	-11.3
football	-0.4	0.0	-0.2	-3.5
foreman	-1.5	-0.1	-1.2	-14.0
hall	-2.0	-0.3	-0.4	-21.3
harbour	-1.3	0.3	-0.2	-8.9
highway	-2.5	0.0	-0.8	-9.8
ice	-0.6	-0.3	-0.1	-7.0
mobile	-1.0	0.2	-0.5	-11.7
mother-daughter	-1.6	-0.5	0.1	-22.8
news	-1.9	-0.3	-1.5	-24.5
silent	-0.7	0.3	-0.7	-25.3
soccer	-3.4	-0.4	0.2	-4.1
stefan	-2.1	1.0	0.5	-11.1
tempeste	-1.0	-0.4	-0.2	-9.8
Average	-2.0	0.0	-0.9	-14.0

Table 3
Test conditions for adaptive cycle-size selection.

Profile	Main
RD optimization	Yes
Rate control	Yes (1 Mbps)
Packet loss	Yes (packet loss rate = 0–20%)
Error concealment	Copying the previous image (built-in method in JM18.6)
Number of reference frames	1

- (2) The cycle-size is determined by our proposed selection model Eq. (14), using our proposed intra-refresh partition and order.
- (3) Optimal cycle-size that maximizes the PSNR of reconstruction video, using random intra-refresh coding as default as in JM 18.6.
- (4) Fixed cycle-size of 12 coding with the scheme in [16].

We have following remarks for these results:

- (1) Our proposed linear model for cycle selection can approximate the optimal cycle-size very well.
- (2) Combine the cycle selection model and the proposed intra-refresh partition and order, the performance are even better than the optimal results by using the random intra-refresh scheme, especially when the packet loss rate is high (where a small cycle size is used). It is because when the cycle-size is small, the random intra-refresh coding has lower coding efficiency.
- (3) Compared with [16] with a fixed cycle-size $N = 12$, about 3 dB PSNR gain in average can be achieved especially under a high packet loss rate.

To show the improvement of our proposed method over method in [16] more clearly, we pick up some typical packet loss rates and show the PSNR difference in Table 4. Positive number indicates PSNR gain (quality improvement).

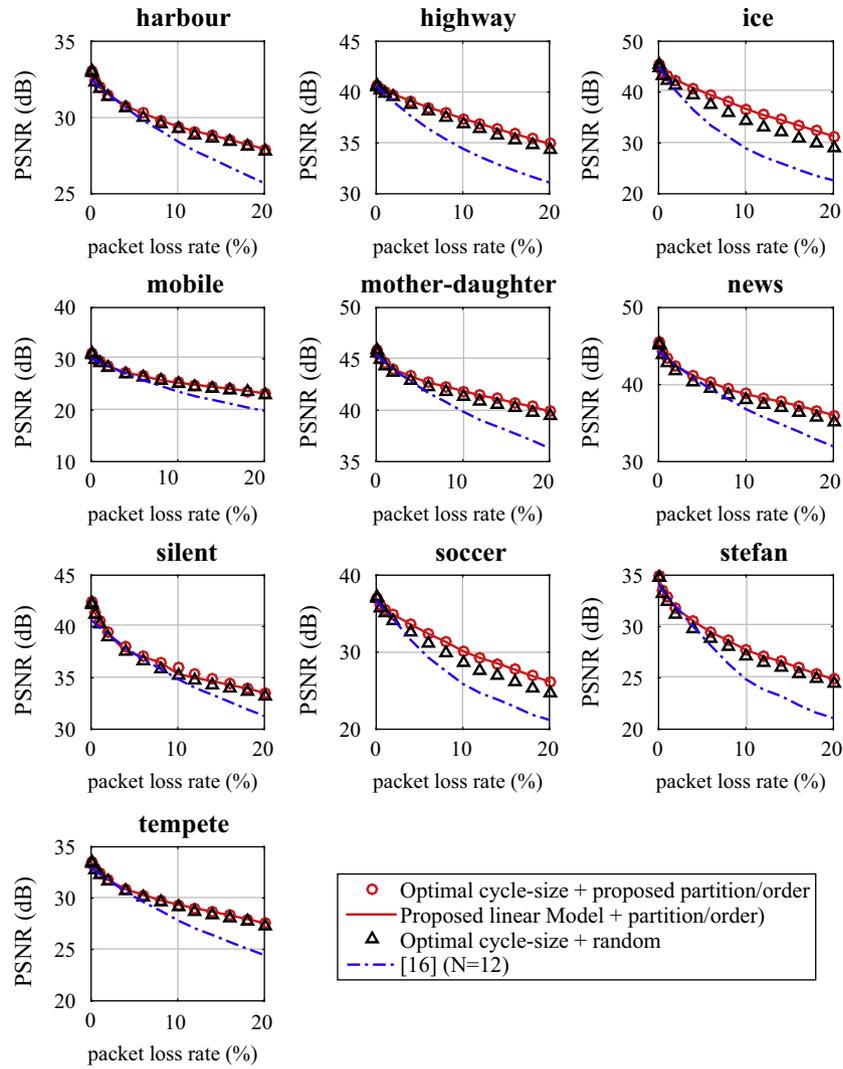


Fig. 8. Comparison of the proposed cycle-size selection algorithm with our intra-refresh partition and order, optimal cycle-size with our intra-refresh partition and order, the scheme in [16] and the optimal cycle-size with random intra-refresh. The PSNR is for the decoded video sequences. The lost frame is concealed with copying the previous decoded frame.

Table 4
PSNR (in db) comparison for some packet loss rates versus [16].

Sequences	$p = 0.1\%$	$p = 1\%$	$p = 10\%$	$p = 20\%$
harbour	0.45	0.01	0.98	2.25
highway	0.00	0.26	2.96	3.82
ice	0.04	0.65	7.77	8.66
mobile	0.62	0.11	1.71	3.42
mother–daughter	0.30	−0.01	1.99	3.59
news	0.69	0.09	1.96	4.06
silent	1.37	0.52	0.49	2.27
soccer	0.12	−0.03	4.27	4.97
stefan	0.52	−0.20	2.86	3.80
tempete	0.46	0.00	1.53	3.10
Average	0.46	0.14	2.65	3.99

Beside the PSNR, we also compare the decoded video quality between our proposed method and [16] with the SSIM metric [23]. The results are shown in Table 5.

In practice, the receiver may need to count the packet losses and feedback to the encoder. This process will induce latency between the estimated packet loss rate and the actual packet loss rate. To investigate the performance of our proposed method in

this scenario, we perform simulations to investigate the PSNR changes with inaccurate packet loss rate estimations due to the latency of the feedback from the decoder. In this simulation, a real packet loss rate (1.0%) is used, while the estimated packet loss rate used to derive the intra-refresh cycle size varies within a range (0.1–4.0%). The results are in Fig. 9. We have following two remarks:

- (1) For most sequences, the best PSNR appears at the matching point (estimated packet loss rate = actual packet loss rate = 1%). It can further confirm the effectiveness of our proposed linear model.
- (2) When the estimated packet loss rate deviates from the actual packet loss rate. The PSNR may degrade, but the quality is still relatively good.

4.3. Visual results

In Figs. 10 and 11, some visual results of the same frame are compared with different coding methods to show the resultant dislocation artifacts. It is clear that our proposed method have a good error-resilience performance. Compared with the scattered

Table 5
SSIM comparison for some packet loss rate versus [16].

Sequences	$p = 0.1\%$		$p = 1.0\%$	
	[16]	Proposed	[16]	Proposed
harbour	0.953	0.958	0.947	0.947
highway	0.954	0.954	0.950	0.951
ice	0.984	0.984	0.975	0.980
mobile	0.939	0.945	0.927	0.928
mother–daughter	0.986	0.986	0.984	0.983
news	0.985	0.987	0.984	0.984
silent	0.975	0.981	0.973	0.976
soccer	0.918	0.921	0.900	0.892
stefan	0.972	0.974	0.957	0.953
tempete	0.955	0.959	0.950	0.950
Average	0.962	0.965	0.954	0.954
	$p = 10.0\%$		$p = 20.0\%$	
	[16]	Proposed	[16]	Proposed
harbour	0.895	0.909	0.838	0.880
highway	0.922	0.936	0.895	0.920
ice	0.896	0.946	0.826	0.910
mobile	0.802	0.845	0.664	0.781
mother–daughter	0.962	0.973	0.938	0.961
news	0.967	0.971	0.947	0.962
silent	0.948	0.941	0.923	0.927
soccer	0.744	0.807	0.637	0.733
stefan	0.802	0.870	0.696	0.801
tempete	0.886	0.909	0.806	0.874
Average	0.882	0.911	0.817	0.875

random intra-refresh coding, dislocation problem is avoided. It should be noted that the MB region groupings in the vertical partition intra-refresh scheme can be considered as a special case of our rectangle region grouping, and so will also not show the

dislocation errors. However, due to its partition, some vertical stripe artifacts could be observed, and it has lower coding efficiency (shown in Table 2).

5. Conclusion and future work

In this paper, we present an efficient intra-refresh cycle-size selection model depending on the network packet loss rate and the motion in the video content. We also present a design of the intra-refresh partition and order which is based on the dislocation error, constrained intra-prediction, and constrained inter-prediction. To select the best cycle-size adaptively according to the packet loss rate, we propose a linear model, where no heuristic sequence-specified parameters are needed. In the intra-refresh partition and order design, we discuss issues in intra-refresh coding, and propose a universal strategy for the intra-refresh partition and order design for different cycle sizes. Experimental results confirm the effectiveness of the proposed algorithms.

In the current simulations, one intra-refresh cycle size is used for the whole video sequence, which does not consider the scene change in the video. One of our future works is to extend this scheme so that the parameters can be updated more frequently and the cycle size selection can be adaptive to the scene changes. Another future work is to extend this method in the newly H.265/HEVC standard [24]. The H.265/HEVC uses a quad-tree coding structure, where the coding unit size varies from 8×8 to 64×64 . A straightforward way to adapt our low-delay error-resilient method in HEVC is to apply the intra-refreshing for an LCU (largest coding unit), which is similar to the MB. More work will be done to investigate the intra-refresh in H.265/HEVC.

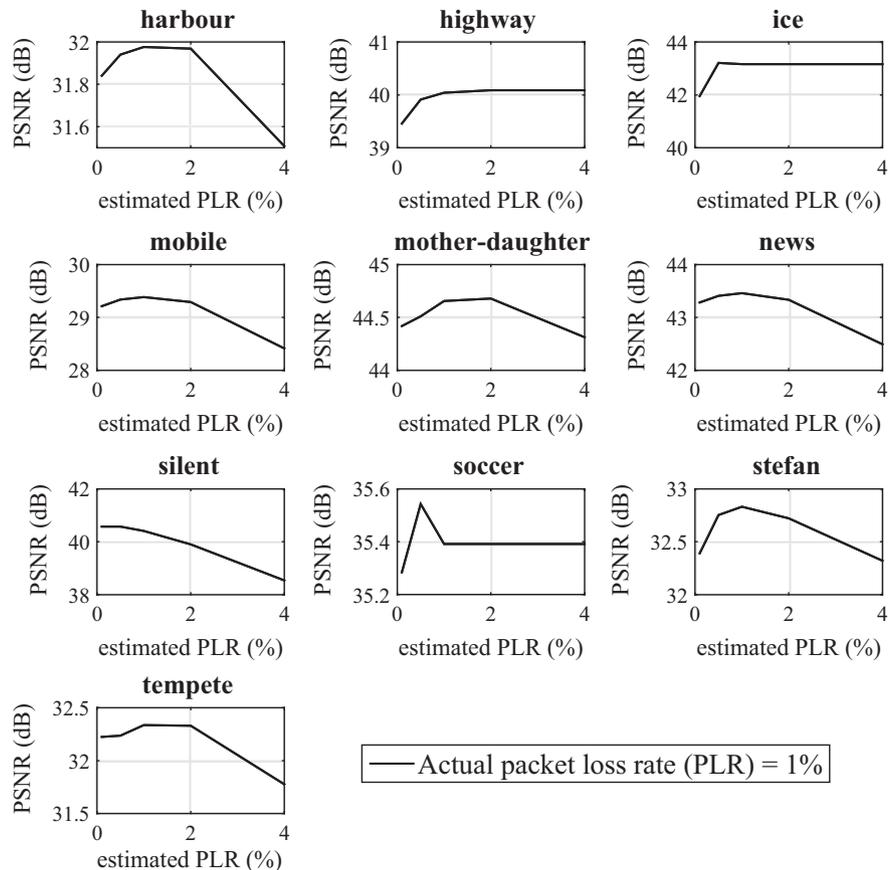


Fig. 9. Decoded PSNR for different estimated packet loss rates from 0.1% to 4.0% when the actual packet loss rate is 1%.



Fig. 10. One reconstructed frame from “foreman” with transmission error. In different methods, the lost frames are the same.



Fig. 11. One reconstructed frame from “soccer” with transmission error. In different methods, the lost frames are the same.

References

- [1] ISO/IEC 13818-2 and ITU-T Rec. H-262, Information Technology – Generic Coding of Moving Pictures and Associated Audio Information: Video, 1996.
- [2] Shao-Shuai Gao, Guo-Fang Tu, Robust H.263+ video transmission using partial backward decodable bit stream (PBDDBS), *IEEE Trans. Circuits Syst. Video Technol.* 13 (2) (2003) 182–187.
- [3] S. Kumar, Error resiliency measure for RVLC codes, *IEEE Signal Process. Lett.* 13 (2) (2006) 84–87.
- [4] R.H. Morelos-Zaragoza, *The Art of Error Correcting Coding*, John Wiley & Sons, 2006.
- [5] Tong Gan, Kai-Kuang Ma, Weighted unequal error protection for transmitting scalable object-oriented images over packet-erasure networks, *IEEE Trans. Image Process.* 14 (2) (2005) 189–199.
- [6] Y. Wang, S. Wenger, J. Wen, A.K. Katsaggelos, Error resilient video coding techniques, *IEEE Signal Process. Mag.* 17 (4) (2000) 61–82.
- [7] M. Fleury, I.A. Ali, M. Ghanbari, Video intra coding for compression and error resilience: a review, *Recent Patents Signal Process.* 4 (1) (2014) 32–43.
- [8] VideoLAN-x264, The Best H.264/AVC Encoder, <<http://www.videolan.org/developers/x264.html>> (accessed: 12.01.12).
- [9] Z. He, J. Cai, C.W. Chen, Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding, *IEEE Trans. Circuits Syst. Video Technol.* 12 (6) (2002) 511–523.
- [10] G. Côté, F. Kossentini, Optimal intra coding of blocks for robust video communication over the Internet, *Signal Process. Image Commun.* 15 (1) (1999) 25–34.
- [11] H.264/AVC Reference Software jm18.6. <<http://iphome.hhi.de/suehring/tml/download/jm18.6.zip>> (accessed: 02.06.14).
- [12] R. Zhang, S.L. Regunathan, K. Rose, Video coding with optimal inter/intra-mode switching for packet loss resilience, *IEEE J. Sel. Areas Commun.* 18 (6) (2000) 966–976.
- [13] Q. Chen, Z. Chen, X. Gu, C. Wang, Attention-based adaptive intra refresh for error-prone video transmission, *IEEE Commun. Mag.* 45 (1) (2007) 52–60.
- [14] Y. Dhondt, S. Mys, K. Vermeersch, R. Van de Walle, Constrained inter prediction: removing dependencies between different data partitions, in: *Advanced Concepts for Intelligent Vision Systems*, Springer, Berlin, Heidelberg, 2007, pp. 720–731.
- [15] M.M. Hannuksela, Y.-K. Wang, M. Gabbouj, Isolated regions in video coding, *IEEE Trans. Multimed.* 6 (2) (2004) 259–267.
- [16] R.M. Schreier, A. Rothermel, Motion adaptive intra refresh for the H.264 video coding standard, *IEEE Trans. Consum. Electron.* 52 (1) (2006) 249–253.
- [17] R. Schreier, A. Rothermel, Motion adaptive intra refresh for low-delay video coding, in: *Proceedings of the International Conference on Consumer Electronics (ICCE 2006)*, 2006.
- [18] P. Nunes, D.L. Soares, F. Pereira, Automatic and adaptive network-aware macroblock intra refresh for error-resilient H.264/AVC video coding, *ICIP 2009* (2009) 3073–3076.
- [19] I. Ali, M. Fleury, M. Ghanbari, Content-aware intra-refresh for video streaming over lossy links, in: *International Conference on Consumer Electronics (ICCE 2012)*, 2012.
- [20] I. Ali, S. Moiron, M. Ghanbari, M. Fleury, Prioritized packetization for video with intra-refresh macroblock line”, in: *International Conference on Multimedia Expo (ICME 2011)*, 2011.
- [21] H. Chen, C. Zhao, M.-T. Sun, A. Drake, Adaptive intra-refresh for low-delay error-resilient video coding, in: *Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2014 Asia-Pacific, 2014, pp. 1–4.
- [22] G. Bjontegaard, Calculation of average PSNR differences between rd curves, in: *VCEG-M33 ITU-T Q6/16*, 2001.
- [23] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, *IEEE Trans. Image Process.* 13 (4) (2004) 600–612.
- [24] G.J. Sullivan, J.R. Ohm, W.J. Han, T. Wiegand, Overview of the high efficiency video coding (HEVC) standard, *IEEE Trans. Circuits Syst. Video Technol.* 22 (2012) 1649–1668.