

Isolated Regions in Video Coding

Miska M. Hannuksela, *Member, IEEE*, Ye-Kui Wang, *Member, IEEE*, and Moncef Gabbouj, *Senior Member, IEEE*

Abstract—Different types of prediction are applied in modern video coding. While predictive coding improves compression efficiency, the propagation of transmission errors becomes more likely. In addition, predictive coding brings difficulties to other aspects of video coding, including random access, parallel processing, and scalability. In order to combat the negative effects, video coding schemes introduce mechanisms, such as slices and intracoding, to limit and break the prediction. This paper proposes the use of the isolated regions coding tool that jointly limits in-picture prediction and interprediction on region-of-interest basis. The tool can be used to provide random access points from non-intrapictures and to respond to intrapicture update requests. Furthermore, it can be applied as an error-robust macroblock mode decision method and can be used in combination with unequal error protection. Finally, it enables mixing of scenes, which is useful in coding of masked scene transitions.

Index Terms—Error resilience, isolated regions, random access, video coding.

I. INTRODUCTION

CURRENT video coding standards include ITU-T H.261, ITU-T H.263, ISO/IEC MPEG-1 Part 2, ISO/IEC MPEG-2 Part 2 (a.k.a. ITU-T H.262), and ISO/IEC MPEG-4 Part 2. These standards are based on block-based translational motion compensation and discrete cosine transform (DCT) based residual coding and are herein referred to as conventional video coding standards. The Joint Video Team (JVT) of ITU-T and ISO/IEC recently finalized a new standard based on an earlier ITU-T standardization project called H.26L. The resulting standard is called ITU-T Recommendation H.264 or ISO/IEC International Standard 14496-10 (MPEG-4 Part 10) [1] and is referred to as the Advanced Video Coding (AVC) standard in this paper.

During transmission, many video communication systems undergo transmission errors. Transmission errors can be categorized into bit errors and packet errors. Bit errors are typically caused by imperfections of physical channels, such as radio interference; while, packet errors are typically due to elements in packet-switched networks. For example, a packet router may become congested; i.e., it may get too many packets as input and cannot output them at the same rate. In this situation, its buffers overflow, and some packets get lost as a result.

Packet duplication and packet delivery in different order than transmitted are also possible.

A video communication system includes a transmitter and a receiver. A transmitter includes a source coder and a transport coder. The source coder inputs uncompressed images and outputs coded video stream. The transport coder encapsulates the compressed video according to the transport protocols in use. The receiver performs inverse operations, i.e., transport decoding and source decoding, to obtain a reconstructed video signal. Transmission errors can be controlled in the transport coding layer or in the source coding layer or jointly in both layers. For example, some transport systems enable unequal error protection where part of the transmitted stream is conveyed more reliably than the rest.

Interactive error concealment refers to techniques where the recipient transmits information about corrupted decoded areas and/or transport packets to the transmitter. Many communication systems include a mechanism to convey such feedback information. For example, in ITU-T H.323 and H.324 video conferencing standards, the receiver can request an intra-update of an entire picture or certain macroblocks using the H.245 control protocol. The transmitter typically responds to such a request by coding the requested area in intramode in the next picture to be coded.

Noninteractive error control techniques do not involve interaction between the transmitter and the receiver. Error concealment refers to techniques where the receiver estimates the correct decoded representation of erroneously received data. Forward error control refers to techniques where the transmitter adds such redundant data in the coded stream that helps the receiver conceal transmission errors.

A thorough review of error resilient video coding techniques is given in [2].

Another important aspect in video communication is random access. Random access refers to the ability of the decoder to start decoding a stream at a point other than the beginning of the stream and recover an exact or approximate representation of the decoded pictures. A random access point and a recovery point characterize a random access operation. The random access point is any coded picture where decoding can be initiated. All decoded pictures at or subsequent to a recovery point in output order are correct or approximately correct in content. If the random access point is the same as the recovery point, the random access operation is instantaneous; otherwise, it is gradual.

Random access points enable seek, fast forward, and fast backward operations in locally stored video streams. In video on-demand streaming, servers can respond to seek requests by transmitting data starting from the random access point that is closest to the requested destination of the seek operation.

Manuscript received December 30, 2002; revised August 7, 2003. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Antonio Ortega.

M. M. Hannuksela is with Nokia Research Center, 33721 Tampere, Finland (e-mail: miska.hannuksela@nokia.com).

Y.-K. Wang is with Nokia Mobile Software, 33721 Tampere, Finland (e-mail: ye-kui.wang@nokia.com).

M. Gabbouj is with the Tampere University of Technology, 33101 Tampere, Finland (e-mail: moncef.gabbouj@tut.fi).

Digital Object Identifier 10.1109/TMM.2003.822784

Random access points enable tuning in to a broadcast. In addition, a random access point can be coded as a response to a scene cut in the source sequence or as a response to a fast update intrapicture update request. The proposed isolated regions tool shall prove useful in providing additional random access capability.

This paper is organized as follows. Section II summarizes the types of prediction used in video coding. Applications where prediction needs to be limited or disabled are presented, and a review of methods to limit prediction is given. Section III presents the isolated regions technique, which is based on limiting predictive coding in a specific way. Moreover, the relation of the AVC standard to the isolated region technique is presented in the same section. Section IV demonstrates how isolated regions can be used in random access. Section V applies the isolated regions technique in forward error control; whereas, isolated regions are used in combination with unequal error protection in Section VI. Scene mixing, as presented in Section VII, is yet another application for the isolated regions technique. Finally, Section VIII concludes the paper.

II. PREDICTIVE VIDEO CODING

A. Types of Prediction

Video coding is typically a two-stage process. First, a prediction of the video signal is generated based on previous coded data. Second, the residual between the predicted signal and the source signal is coded. Prediction enables efficient compression, but it causes some complications in error-prone environments, in random access, and in parallel decoding. In the following, we categorize the most commonly used types of prediction and in Sections II-B–E, we describe the applications and means for constrained prediction.

Interprediction, which is also referred to as temporal prediction and motion compensation, removes temporal redundancy. In interprediction, the sources of prediction are previously decoded pictures. H.263, MPEG-4 Part 2, and the AVC standard enable storage of multiple reference pictures for interprediction and selection of the used reference picture on picture segment or macroblock basis.

Intraprediction utilizes the fact that adjacent pixels within the same picture are likely to be correlated. Intraprediction can be performed in spatial or transform domain, i.e., either sample values or transform coefficients can be predicted. Intraprediction is typically exploited in intracoding, where no interprediction is applied.

One outcome of the coding procedure is a set of coding parameters, such as motion vectors and quantized transform coefficients. Many parameters can be entropy-coded more efficiently if they are predicted first from spatially or temporally neighboring parameters. For example, a motion vector is typically predicted from spatially adjacent motion vectors. Prediction of coding parameters and intraprediction are collectively referred to as in-picture prediction in this paper.

B. Applications for Constrained Prediction

While prediction brings high compression efficiency, it causes inconveniences in other aspects such as error resiliency,

random access, parallel processing, and scalability. To compromise between any of these aspects and compression efficiency, constraining prediction is required.

Error Resiliency: If a piece of coded data is hit by a transmission error, the error is visible not only in the decoded area corresponding to the piece of data, but also in spatially neighboring areas that are predicted from the corrupted area. Moreover, all coding parameters predicted from corrupted parameter values are likely to be incorrect. Furthermore, due to interprediction, the artifacts caused by transmission errors propagate in time. Therefore, constraining prediction in a way that transmission errors are as imperceptible as possible is one of the key features in error-prone video communication systems.

Random Access: Random access refers to the ability to start the decoding at any of the random access points of the stream and recover decoded pictures that are correct in content. Frequent random access points are desirable in many applications. For example, random access points allow new recipients to tune in to a video broadcast, and they allow seeking to a desired position in stored video, such as DVD. In order to code a random access point at a specific picture, typically interprediction has to be broken.

Parallel Processing: Parallel processing refers to the process of encoding/decoding different parts of a picture simultaneously. Parallel processing is a desirable feature in multiprocessor architectures. In practice, parts of a picture being coded simultaneously have to be independent, i.e., no prediction from one part to another is allowed.

Scalability: Scalability refers to the capability of a compressed sequence to be decoded at different bit-rates. In scalable video coding prediction is limited in a way that certain parts of the compressed sequence, such as an enhancement layer in layered scalability or a B picture in conventional video coding standards, can be ignored in the decoding process without affecting the decoding of the rest of the compressed sequence. Scalable coded sequences can be used for many purposes. For example, a streaming server may adjust the bit-rate of a prestored coded sequence according to the prevailing network conditions.

C. Means to Limit In-Picture Prediction

Video coding standards allow dividing a coded picture to coded segments or slices. In-picture prediction is typically disabled across slice boundaries. Thus, slices can be regarded as a way to split a coded picture to independently decodable pieces. Coded segments can be categorized into three classes: raster-scan-order slices, rectangular slices, and flexible slices.

A raster-scan-order-slice is a coded segment that consists of consecutive macroblocks in raster scan order. Video packets of MPEG-4 Part 2 and groups of macroblocks (GOBs) starting with a nonempty GOB header in H.263 are examples of raster-scan-order slices.

A rectangular slice is a coded segment that consists of a rectangular area of macroblocks. A rectangular slice may be higher than one macroblock row and narrower than the entire picture width. H.263 includes an optional rectangular slice submode, and H.261 GOBs can also be considered as rectangular slices.

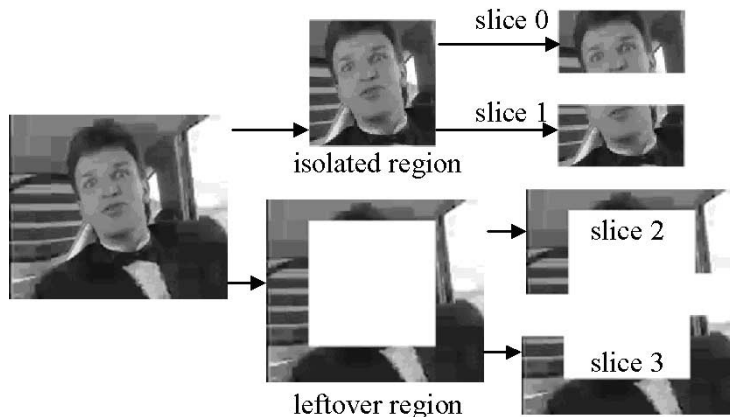


Fig. 1. Example partitioning of a picture to an isolated region and a leftover region and further to slices.

A flexible slice can contain any predefined macroblock locations. The AVC codec allows grouping of macroblocks to more than one slice groups. A slice group can contain any macroblock locations, including nonadjacent macroblock locations. A slice consists of at least one macroblock within a particular slice group in raster scan order.

D. Means to Limit Interprediction

Intracoding of pictures and macroblocks is one way to break interprediction. Reference picture selection can be used to make the chains of interpictures shorter. In addition, interprediction can be limited by restricting the values of motion vectors. A brief review of reference picture selection based methods limiting interprediction has been given in [3].

E. Types and Limitation of Prediction in the AVC Codec

The syntax of a coded AVC sequence consists of Network Abstraction Layer (NAL) units. A NAL unit is an atomic element that can be framed for transport and parsed independently. Each NAL unit has a specific type, which can be a coded slice, a coded data partition, a sequence parameter set, a picture parameter set, or a supplemental enhancement information (SEI) message among other things. The parameter set concept [4] replaces the use of sequence and picture headers. In contrast to redundant coding of sequence and picture headers for improved error resiliency, the AVC codec enables transmission of sequence and picture parameter sets externally from the rest of a coded sequence using another, more reliable transmission channel or protocol.

Some coding parameters in a NAL unit of one type depend on coding parameters of a NAL unit of another type. In particular, the following dependency hierarchy relates to coded slices: A coded slice consists of a slice header and slice data. A slice header refers to a picture parameter set, and a picture parameter set refers to a sequence parameter set. A picture parameter set contains parameters whose values remain unchanged within a coded picture, whereas the parameters in a sequence parameter set remain unchanged during an entire coded video sequence.

A coded picture consists of at least one coded slice. Coded parameters are not predicted across slice boundaries. Many pa-

rameter values of a slice are adaptively predicted from earlier coding parameters of the same slice.

The AVC codec includes a number of directional pixel-domain intraprediction modes for 4×4 or 16×16 blocks. The border pixels of the neighboring blocks above and on the left are used as prediction sources. A block is not used as a source for intraprediction if it belongs to a different slice than the block to be coded or decoded. The picture parameter set contains a constrained intraprediction flag that is used to control whether neighboring non-intracoded blocks are used for intraprediction.

Interprediction is based on translational motion of blocks. Motion vectors have the accuracy of $1/4$ luma samples. Fractional pixels are interpolated using a two-stage filtering process including a 6-tap and a 2-tap filter. Interprediction can be limited by selecting reference pictures for prediction carefully. Moreover, a particular type of an intrapicture, called an instantaneous decoding refresh (IDR) picture, has been specified. No subsequent picture can refer to pictures that are earlier than the IDR picture in decoding order. Thus each IDR picture forms a random access point.

III. ISOLATED REGIONS

A. Fundamentals of Isolated Regions

The proposed technique isolated regions is based on constraining in-picture prediction and interprediction jointly.

An isolated region in a picture can contain any macroblock locations, and a picture can contain zero or more isolated regions that do not overlap. A leftover region is the area of the picture that is not covered by any isolated region of a picture. When coding an isolated region, in-picture prediction is disabled across its boundaries. A leftover region may be predicted from isolated regions of the same picture.

A coded isolated region can be decoded without the presence of any other isolated or leftover region of the same coded picture. It may be necessary to decode all isolated regions of a picture before the leftover region. An isolated region or a leftover region contains at least one slice. Fig. 1 presents an example where the picture contains one isolated region and a leftover region. Both the isolated region and the leftover region contain two slices.

Pictures, whose isolated regions are predicted from each other, are grouped into an isolated-region picture group. An isolated region can be interpredicted from the corresponding isolated region in other pictures within the same isolated-region picture group, whereas interprediction from other isolated regions or outside the isolated-region picture group is disallowed. A leftover region may be interpredicted from any isolated region. The shape, location, and size of coupled isolated regions may evolve from picture to picture in an isolated-region picture group.

B. Coding of Isolated Regions in the AVC Codec

Coding of isolated regions in the AVC codec is based on slice groups introduced in Section II-C. The mapping of macroblock locations to slice groups is specified in the picture parameter set. The AVC syntax includes efficient methods to code certain slice group patterns, which can be categorized into two types, static and evolving. The static slice groups stay unchanged as long as the picture parameter set is valid, whereas the evolving slice groups can change picture by picture according to the corresponding parameters in the picture parameter set and a slice group change cycle parameter in the slice header. The static slice group patterns include interleaved, checkerboard, rectangular oriented, and freeform. The evolving slice group patterns include horizontal wipe, vertical wipe, box-in, and box-out. The rectangular oriented pattern and the evolving patterns are especially suited for coding of isolated regions and are described more carefully in the following.

For a rectangular oriented slice group pattern, a desired number of rectangles are specified within the picture area. A foreground slice group includes the macroblock locations that are within the corresponding rectangle but excludes the macroblock locations that are already allocated by slice groups specified earlier. A leftover slice group contains the macroblocks that are not covered by the foreground slice groups. The left-hand side picture in Fig. 2 includes two rectangular foreground slice groups (indicated by a white rectangle) and the righthand side picture in Fig. 2 includes three foreground slice groups, two of which are rectangular and the third one, i.e., the screen behind the newsreaders, is composed by excluding the first two rectangles from a bounding rectangle.

An evolving slice group is specified by indicating the scan order of macroblock locations and the change rate of the size of the slice group in number of macroblocks per picture. Each coded picture is associated with a slice group change cycle parameter (conveyed in the slice header). The change cycle multiplied by the change rate indicates the number of macroblocks in the first slice group. The second slice group contains the rest of the macroblock locations. Fig. 3 shows an example of the first five change cycles of the first slice group of the box-out type with a change rate of 12 macroblocks.

In-picture prediction is always disabled across slice group boundaries, because slice group boundaries lie in slice boundaries. Therefore, each slice group is an isolated region or leftover region.

Each slice group has a unique identification number within a picture. Encoders can restrict the motion vectors in a way that they only refer to the decoded macroblocks belonging to slice



Fig. 2. Examples of rectangular oriented isolated regions.



Fig. 3. Example of an evolving isolated region.

groups having the same identification number as the slice group to be encoded. Encoders should take into account the fact that a range of source samples is needed in fractional pixel interpolation and all the source samples should be within a particular slice group.

The AVC codec includes a deblocking loop filter. Loop filtering is applied to each 4×4 block boundary, but loop filtering can be turned off at slice boundaries. If loop filtering is turned off at slice boundaries, perfect reconstructed pictures can be achieved when performing gradual random access. Otherwise, reconstructed pictures would be imperfect in content even after the recovery point. However, in many applications the mismatch is unperceivable and the picture quality is acceptable without turning off the loop filtering at slice boundaries.

The recovery point SEI message and the motion constrained slice group set SEI message of the AVC standard can be used to indicate that some slice groups are coded as isolated regions with restricted motion vectors. The decoder may utilize the information to achieve faster random access or to save in processing time by ignoring the leftover region.

C. Comparison to Earlier Techniques for Joint In-Picture and Interprediction Limitation

As far as the authors are aware, the closest predecessor of the isolated regions technique is the optional independent segment decoding mode of H.263 (H.263, Annex R). When this optional mode is in use, all slices have to be rectangular. Slice boundaries are treated as picture boundaries, and therefore no spatio-temporal error propagation over slice boundaries occurs. Due to restricted motion prediction, compression efficiency drops compared to normal slice-based operation. The locations of slice boundaries have to remain unchanged within a group of pictures (GOP). This fact hinders the use of the independent segment decoding mode for many of the applications presented in this paper. Furthermore, because the number of macroblocks in a slice is constant within a GOP, the encoder has few means to control the coded size of a slice in bytes. This fact may make the encapsulation of slices to transport packets nonoptimal, because the slice size cannot be adjusted according to an optimal packet size according to prevailing network conditions.

In many applications, such as the case presented in Section VI-C, one rectangular isolated region is sufficient. If

such a scheme were coded with H.263 rectangular slices, five rectangular slices would be needed in contrast to one isolated region and one leftover region. Consequently, both in-picture and interprediction falling into the area of the leftover region would be disallowed unnecessarily across the boundaries of the rectangular slices.

IV. RANDOM ACCESS

A. Gradual Decoding Refresh

Conventionally each intrapicture has been a random access point in a coded sequence. The introduction of multiple reference pictures for interprediction caused that an intrapicture may not be sufficient for random access. For example, a decoded picture before an intrapicture in decoding order may be used as a reference picture for interprediction after the intrapicture in decoding order. Therefore, an IDR picture as specified in the AVC standard or an intrapicture having similar properties to an IDR picture has to be used as a random access point. In this section term IDR picture is not exclusively specific to the AVC standard.

Gradual decoding refresh (GDR) refers to the ability to start the decoding at a non-IDR picture and recover decoded pictures that are correct in content after decoding a certain amount of pictures. That is, GDR can be used to achieve random access from non-intraframes. Some reference pictures for interprediction may not be available between the random access point and the recovery point, and therefore some parts of decoded pictures in the gradual decoding refresh period cannot be reconstructed correctly. However, these parts are not used for prediction at or after the recovery point, which results into error-free decoded pictures starting from the recovery point.

It is obvious that gradual decoding refresh is more cumbersome both for encoders and decoders compared to instantaneous decoding refresh. However, gradual decoding refresh is desirable in error-prone environments thanks to two facts: First, a coded intrapicture is generally considerably larger than a coded non-intraframe. This makes intrapictures more susceptible to errors than non-intraframes, and the errors are likely to propagate in time until the corrupted macroblock locations are intracoded. Second, intracoded macroblocks are used in error-prone environments to stop error propagation (see Section V-A for more details). Thus, it makes sense to combine the intramacroblock coding for random access and for error propagation prevention, for example, in video conferencing and broadcast video applications that operate on error-prone transmission channels. This conclusion is utilized in gradual decoding refresh.

An evolving isolated region can be used to provide gradual decoding refresh. A new evolving isolated region is established in the picture at the random access point, and the macroblocks in the isolated region are intracoded. The shape, size, and location of the isolated region evolve from picture to picture. The isolated region can be interpredicted from the corresponding isolated region in earlier pictures in the gradual decoding refresh period. When the isolated region covers the whole picture area, a picture completely correct in content is obtained when decoding

started from the random access point. This process can also be generalized to include more than one evolving isolated region that eventually cover the entire picture area.

There may be tailored in-band signaling, such as the recovery point SEI message of the AVC standard, to indicate the gradual random access point and the recovery point for the decoder. Furthermore, the recovery point SEI message includes an indication whether an evolving isolated region is used between the random access point and the recovery point to provide gradual decoding refresh.

Gradual decoding refresh using isolated regions can also be applied as a response to intrapicture update request. In applications with a feedback channel, a receiving terminal may request the far-end encoder for an intrapicture refresh if the received pictures are too corrupted. There is another use of an intrapicture refresh request in multipoint video conferencing, in which the multipoint control unit orchestrates a switch of source sequences delivered to recipients by issuing an intrapicture refresh request to a desired source terminal. Conventionally, an encoder responds to an intrapicture refresh request by coding and transmitting an intracoded picture. Due to avoiding of intrapicture coding, improved error resiliency can be achieved by using isolated regions.

B. Simulations

Two sets of simulations were done using the AVC codec.

- 1) *Coding efficiency simulations.* Gradual decoding refresh based on isolated regions was compared to periodic IDR picture coding at a 1-s random access period. Error-free application environment, such as local storage, was assumed, and therefore the coding options yielding the best coding efficiency were selected. The simulations abided the coding efficiency simulation common conditions specified by ITU-T Video Coding Experts Group [5]. A number of QCIF and CIF sequences were coded, and the average bitrate loss of gradual decoding refresh compared to periodic IDR was between 11% and 17%. More results can be obtained from [6].
- 2) *Error resiliency simulations.* The error resiliency performance of gradual decoding refresh was compared with the periodic IDR picture coding. The target was to simulate IP multicast streaming where random access points allow new receivers to start decoding. Random access period of about 1 second was used. Packet loss simulations under loss rates of 0, 3, 5, 10, and 20% were performed according to the conditions specified by ITU-T Video Coding Experts Group [7] with minor modifications as listed in [6]. One set of results is presented in Fig. 4 and more results can be obtained from [6]. It can be seen that gradual decoding refresh performs consistently better compared to periodic IDR in all loss rates. Moreover, the PSNR difference between the cases grows as a function of loss rate. From the simulation results, it can also be seen that using gradual decoding refresh based on isolated regions to respond intraupdate requests has better error resiliency performance than coding intrapictures.

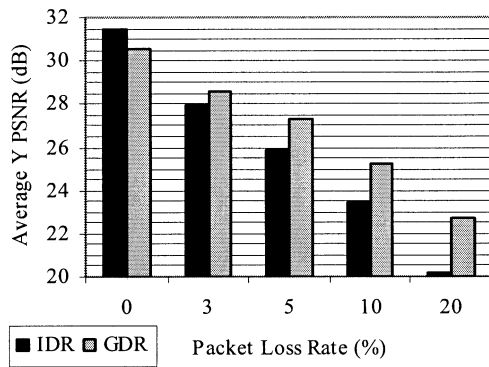


Fig. 4. Comparison of periodic IDR and GDR in terms of average luminance PSNR at different packet loss rates. Sequence: Paris at 384 kbits/s.

V. ERROR-ROBUST INTER/INTRA-MODE DECISION

A. Error-Robust Macroblock Mode Decision

Video encoders have numerous ways to reduce the spatial and temporal propagation of transmission errors and to help decoders concealing transmission errors. One of these methods is to stop temporal error propagation by intramacroblock coding. In applications, where the content is encoded before transmission (e.g., on-demand streaming) or where no feedback about the error or loss locations from the recipients is possible (e.g., live multicast with a huge number of receivers), the encoder has to conclude the rate and locations of intramacroblocks based on expected or measured transmission error or loss rate.

The macroblock mode selection algorithms can be categorized into nonadaptive and adaptive algorithms, and adaptive methods can be further classified to cost-function-based and rate-distortion optimized ones. The family of nonadaptive intrarefresh algorithms includes the circular intrarefresh algorithm that scans the picture area in a predefined order and codes a certain number of intramacroblocks per picture in the predefined scan order. Another example of a nonadaptive algorithm is to code a certain number of macroblocks in intramode at randomly selected macroblock locations. Adaptive macroblock mode decision methods select the intracoded macroblock locations in a way that the content of the pictures is taken into account. For example, a static background area needs not be refreshed in intramode as often as moving objects. Cost-function-based methods, such as [8] and [9], calculate a cost for each macroblock with a certain function that may take into account the amount of prediction error data after motion compensation, for example. A certain number of macroblocks having the highest cost are coded in intramode. Rate-distortion optimized macroblock mode selection algorithms use a Lagrangian cost function that linearly combines terms “rate” and “distortion.” The mode selection of each macroblock is such that the cost is minimized. An estimate of the expected distortion caused by transmission errors and losses is taken into account in the cost function. A number of distortion estimation algorithms have been proposed and one of them, herein referred to as the loss-aware rate-distortion-optimized (LA-RDO) macroblock mode selection algorithm, has been selected into the reference implementation of the AVC codec [10]. The computational complexity of rate-distortion optimized macroblock

mode selection algorithms is typically multifold compared to nonadaptive and cost-function-based algorithms.

B. Isolated Regions in Macroblock Mode Decision

Evolving isolated regions can be used as a nonadaptive macroblock mode selection algorithm. A new evolving isolated region is established at the beginning of an intrarefresh period, i.e., the period of the isolated-region picture group. The intrarefresh period is completed when the isolated region covers the entire picture area. The macroblocks in the isolated region of the first picture in the intrarefresh period are intracoded. The newly added macroblocks in the isolated region of later pictures are intracoded, whereas the other macroblocks in the isolated region can be interpredicted from the corresponding isolated region within the same intrarefresh period.

If the above algorithm has an adaptive change rate for isolated regions or the following modification is applied, the algorithm falls into the category of adaptive macroblock mode selection algorithms: In contrast to coding newly added macroblocks in intramode, the encoder can apply a normal macroblock mode selection algorithm for them. As a result, the newly added macroblocks may be interpredicted from the corresponding isolated region in the same isolated-region picture group or they may be intracoded.

The encoder can select a proper change rate of the isolated region according to the picture size and the assumed transmission error rate. Generally, a good change rate is equivalent to the expected loss rate of macroblocks. For example, for a CIF sequence, if the packet loss rate is 20%, a change rate of about 80 macroblocks per picture is appropriate. However, due to the possible large differences in sequence characteristics and different coding options, a content-adaptive change rate may perform better and is under investigation.

C. Simulations

Four intrarefresh algorithms were compared: conventional circular intrarefresh at a rate of one macroblock row per picture (CIR), the loss-aware rate-distortion-optimized macroblock mode selection of the AVC reference codec (LA-RDO), isolated regions based circular intrarefresh (IREG-CIR), and a combination of LA-RDO and IREG-CIR. Real-time multicast/broadcast to users with different network conditions was assumed. Therefore, the coding options were selected in a way that the strongest error resiliency performance suitable for the worst expected network condition, 20% packet loss rate, was targeted. The coded bitstreams were decoded after packet loss simulation under different loss rates 0, 3, 5, 10, and 20%. Six coded sequences for each intrarefresh algorithm were generated: Foreman QCIF at 64 kbits/s, Foreman QCIF at 144 kbits/s, Hall Monitor QCIF at 32 kbits/s, Irene CIF at 384 kbits/s, Paris CIF at 144 kbits/s, and Paris CIF at 384 kbits/s, referred herein to as sequences 1 to 6, respectively. More details on the simulation conditions can be obtained from [11].

Fig. 5 presents the average luma PSNR of all the test sequences for each intrarefresh algorithm and each packet loss rate. The simulation results show that the difference in average luma PSNR between IREG-CIR and LA-RDO is

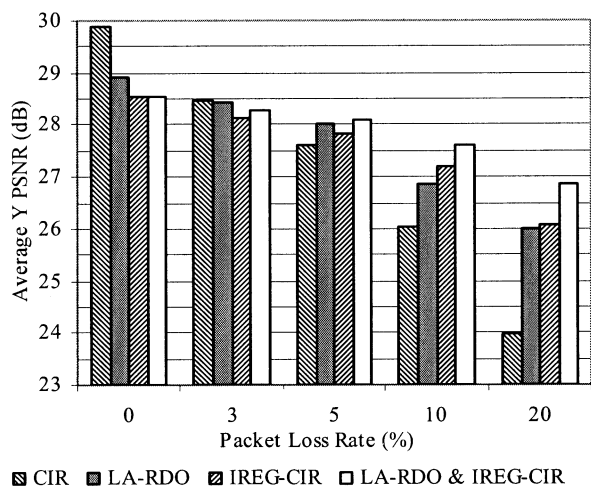


Fig. 5. Comparison of macroblock mode selection algorithms at different packet loss rates. Vertical axis indicates the average luma PSNR of all the test sequences.

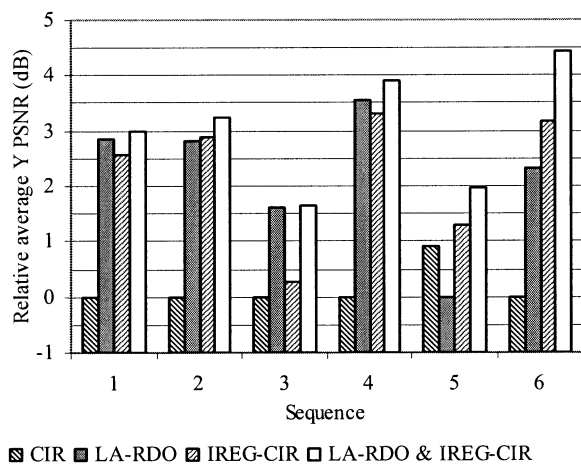


Fig. 6. Comparison of macroblock mode selection algorithms at 20% packet loss rate. Vertical axis indicates the average luma PSNR for a particular sequence and algorithm relative to the worst average luma PSNR for that sequence.

within 0.5 dB regardless of the packet loss rate. In packet loss rates greater than or equal to 5%, the combination of LA-RDO and IREG-CIR outperforms other algorithms, the difference being more than 0.5 dB in the 20% packet loss rate case, to which the bitstreams were optimized. Fig. 6 shows the average luma PSNR for each test sequence in the 20% packet loss rate case. It can be observed that the combination of LA-RDO and IREG-CIR outperforms other algorithms consistently. More detailed simulation results are available in [11].

VI. UNEQUAL ERROR PROTECTION

A. Conventional Coding Tools for Unequal Error Protection

In order to apply unequal error protection, coded video sequences have to be organized in portions of different importance in terms of visual quality. Techniques achieving this goal include data partitioning, scalable coding, and object-based coding.

Data partitioning refers to a technique where subjectively equally important codewords of all macroblocks in a slice are partitioned into a continuous block of data. Typically,

macroblock headers and motion information form one partition and coded prediction error blocks form another partition.

Data partitioning and scalable coding techniques generally treat an entire image equally in spatial domain. However, many images have distinct spatial regions of interest. These regions could have better error protection than other areas in order to obtain a better subjective quality compared to coding and transport schemes that treat all regions equally. Arbitrarily shaped objects [12], as defined in the MPEG-4 Part 2, can be used to extract the regions of interest. However, its high complexity limits its use in real-time encoding.

B. Isolated Regions for Unequal Error Protection

Isolated regions can be used for unequal error protection. The encoder first selects at least one region of interest from the first picture to be encoded using face detection or image analysis techniques, for example. Each region of interest is an isolated region, and the rest of the macroblocks form the leftover region. In the next picture to be encoded, the encoder tracks the same regions of interest as in the previous picture. Each region of interest is coded as an isolated region that is interpredicted only from the corresponding isolated region in the previous reference pictures.

The isolated regions technique allows partitioning pictures spatially and temporally to regions of interest. Each coded isolated region can be further divided into slices and data partitions. Furthermore, the quality of an isolated region can be improved in an enhancement layer, whereas the layer may not provide any quality improvement to the leftover region. Thus, isolated region coding complements data partitioning and scalable coding, and it is an alternative to object-based coding.

C. Simulations

We selected multicast Internet streaming as a target application. A constant rectangular region of interest was selected for each sequence, and smaller quantization steps were used within the region of interest. In one set of sequences the region of interest was coded as an isolated region, and another set of sequences was coded conventionally. The scheme was compared to the conventional codec (version TML8.6 of the AVC public reference software [13]) with and without region-of-interest quantization (abbreviated as Conv + ROI and Conv, respectively). The selection of the quantization step size based on the region of interest was the same in the proposed coding scheme and the Conv + ROI coding scheme.

As interactive error concealment cannot be used in large scale with IP multicast, transport coding level forward error correction (FEC) according to RFC 2733 [14] was used. To be more detailed, we used the so-called parity FEC, where one FEC packet is associated with two media packets and is able to correct the loss of either media packet. Other FEC strengths were not experimented, because we targeted to minimize the delay associated with FEC coding and decoding.

Encapsulation into RTP packets was done as follows. In the proposed coding scheme, intrapictures were encapsulated into five packets. There were two packets for the isolated region: one packet contained odd macroblock rows and another packet



Fig. 7. Results of the unequal error protection simulations: example snapshots of 20% packet loss rate. From left to right, the used codecs are Conv, Conv + ROI, and the proposed codec.

contained even macroblock rows. This slice interleaving mechanism, introduced in [15], was used to obtain a better error concealment result. One parity FEC packet was generated for the two foreground packets according to RFC 2733. The leftover region was packetized into another two packets using slice interleaving method. Two consecutive interpictures consisted a group, and for each such group there were two isolated region packets, one parity FEC packet for the isolated region packets, and two leftover region packets. An isolated region packet contained data from two pictures: macroblocks from even rows of a certain frame and macroblocks from odd rows of the next frame or vice versa. When subpicture coding was not in use, there were three packets for each intra- and interframe: two packets for the entire picture (slice interleaving applied), and one parity FEC packet for the two packets.

Intramacroblock refresh was tailored for the worst expected case (20% packet loss rate), and packet losses were simulated with the obtained packet stream at 0, 3, 5, 10, and 20% packet loss rates. See [16] and [17] for further details on the simulation conditions.

The experiments were done using the Carphone, Hall, Coastguard, Foreman, News, and Irene sequences, with different frame rates and bit-rates. We present only part of the results due to lack of space, more results can be obtained from [16] and [17].

Fig. 7 shows some example snapshots of Foreman at 64 kbps and Carphone at 64 kbps in 20% packet loss rate. It can be seen that in both sequences the proposed subpicture coding scheme with gradual bit allocation maintains the best subjective image quality. In fact, the overall PSNR in the proposed coding drops a little compared to conventional coding cases. However, since errors in the background are far less noticeable than errors in the foreground, the overall subjective quality is improved.

VII. SCENE MIXING

A. Applications

There are a couple of situations where mixing of multiple source pictures into the same coded picture, termed as scene mixing herein, is necessary. The cases can be roughly categorized into masked scene transitions and constant scene mixing. Masked scene transitions are such that one scene spatially uncovers from the other scene or from black in a gradual manner,

and all pictures are mixed and displayed at full intensity. Examples of constant scene mixing include the so-called picture-in-picture scheme, where a picture from one source is included in the picture area originating from another source. For instance, a news broadcast may include a newsreader and a small screen besides her showing video material of a news topic. Furthermore, in video conferencing or surveillance, pictures from multiple cameras may have to be tiled to the same coded picture.

B. Problems

Conventionally, scene mixing is done as follows. First, source pictures are composed from the original pictures of different scenes. Then, the source pictures are coded as if they were normal pictures. The conventional coding approach is not optimal at least due to the following reasons.

- Boundaries of slices do not follow the original source picture boundaries. Thus, in-picture prediction is not likely to succeed well if the source for prediction is from a different scene than the block to be coded.
- It is likely that there is a sharp edge between the original source pictures. If a loop filter is applied, it smoothes the edge unnecessarily.

C. Scene Mixing Based on Isolated Regions

A masked scene transition can be coded with an evolving isolated region. Picture content from one scene is covered by one region and picture content from another scene of the transition is covered by another region. The boundary between the regions moves from picture to picture according to the transition effect.

Constant scene mixing can be implemented as follows: An isolated region covers each original source picture, and the entire picture area excluding the isolated regions forms the leftover region.

As a result of covering each original source picture by an isolated region, each slice contains data from one original picture only. Consequently, in-picture prediction within a slice is likely to succeed well, whereas in-picture prediction and loop filtering in particular is disallowed across the boundaries of source pictures. The disadvantage of the technique compared to conventional coding is that scenes can be mixed along macroblock boundaries only. However, in most cases, especially when the picture sizes are large, the disadvantage does not cause perceivable quality degradations compared to conventional coding.

VIII. CONCLUSION

A novel technique called isolated regions is proposed in this paper. The technique is based on constraining in-picture and interprediction jointly. It provides an elegant solution for many applications, such as gradual decoding refresh, error resiliency and recovery, region-of-interest coding and unequal error protection, picture in picture functionality, and coding of masked video scene transitions. With gradual decoding refresh based on the technique, random access, media channel switching for receivers, and allowing newcomers for multicast streaming is as easy as conventional intrapicture coding with smoother bit-rate

and high error resiliency. Future research directions include investigating proper ways to apply the isolated regions technique in other video coding standards than the AVC standard and investigating adaptive region evolution algorithms for further improved error resilience.

REFERENCES

- [1] Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264 \ ISO/IEC 14496-10 AVC). presented at Joint Video Team Doc. JVT-G050r1. [Online]. Available: ftp://ftp.imtc-files.org/jvt-experts/2003_05_Geneva/JVT-G050r1.zip
- [2] Y. Wang, S. Wenger, J. Wen, and A. K. Katsagelos, "Error resilient video coding techniques," *IEEE Signal Processing Mag.*, vol. 17, pp. 61–82, Jul. 2000.
- [3] M. M. Hannuksela, "Simple packet loss recovery method for video streaming," in *Proc. Int. Packet Video Workshop PV2001*, Apr. 2001.
- [4] T. Stockhammer, M. M. Hannuksela, and S. Wenger, "H.26L/JVT coding network abstraction layer and IP-based transport," in *Proc. IEEE Int. Conf. Image Processing*, Sept. 2002.
- [5] G. Sullivan and G. Bjontegaard. Recommended simulation common conditions for H.26L coding efficiency experiments on low-resolution progressive-scan source material. presented at ITU-T Video Coding Experts Group Doc. VCEG-N81. [Online]. Available: ftp://standard.pictel.com/video-site/0109_San/VCEG-N81.doc
- [6] Y.-K. Wang and M. M. Hannuksela. Gradual decoder refresh using isolated regions. presented at Joint Video Team Doc. JVT-C074. [Online]. Available: ftp://ftp.imtc-files.org/jvt-experts/2002_05_Fairfax/JVT-C074.doc
- [7] S. Wenger. Common conditions for wire-line, low delay IP/UDP/RTP packet loss resilient testing. presented at ITU-T Video Coding Experts Group Doc. VCEG-N79. [Online]. Available: ftp://standard.pictel.com/video-site/0109_San/VCEG-N79r1.doc
- [8] *Annex E, Features Supported, by the Algorithm*, ISO/IEC Int. Std. 14496-2:2001.
- [9] J. Y. Liao and J. D. Villasenor, "Adaptive intra update for video coding over noisy channels," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 1996.
- [10] T. Stockhammer and S. Wenger, "Standard-compliant enhancements of JVT coded video," in *Proc. 2002 Tyrrhenian Int. Workshop on Digital Communications (IWDC 2002)*, Sept. 2002.
- [11] Y.-K. Wang and M. M. Hannuksela. Error-robust video coding using isolated regions. presented at Joint Video Team Doc. JVT-C073. [Online]. Available: ftp://ftp.imtc-files.org/jvt-experts/2002_05_Fairfax/JVT-C073.doc
- [12] N. Brady, "MPEG-4 standardized methods for the compression of arbitrarily shaped video objects," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 1170–1189, Dec. 1999.
- [13] K. Sühring. H.264/AVC Ref. Software. [Online]. Available: <http://bs.hhi.de/~suehring/tml/>
- [14] J. Rosenberg and H. Schulzrinne. An RTP payload format for generic forward error correction. presented at IETF Internet Draft RFC 2733. [Online]. Available: <ftp://ftp.ietf.org/rfc/rfc2733.txt>
- [15] S. Wenger and G. Côté, "Using RFC2429 and H.263+ at low to medium bit-rates for low-latency applications," in *Proc. Int. Packet Video Workshop*, Apr. 1999.
- [16] Y.-K. Wang and M. M. Hannuksela. Results of the core experiment for sub-picture coding. presented at Joint Video Team Doc. JVT-B040. [Online]. Available: ftp://standard.pictel.com/video-site/0201_Gen/JVT-B040.doc
- [17] M. M. Hannuksela, Y.-K. Wang, and M. Gabbouj, "Sub-picture: ROI coding and unequal error protection," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Sept. 2002, pp. 537–540.

Miska M. Hannuksela (M'03) received the M.S. degree in engineering from Tampere University of Technology, Tampere, Finland, in 1997.

He is currently a Research Manager in the Visual Communications Laboratory, of Nokia Research Center, Tampere. From 1996 to 1999, he was a Research Engineer in the area of mobile video communications at the Nokia Research Center. From 2000 to 2003, he was a Project Team Leader and a specialist in various mobile multimedia research and product projects at Nokia Mobile Phones. He has co-authored more than 80 technical contributions to these standardization groups. His research interests include video error resilience, scalable video coding, and video communication systems.

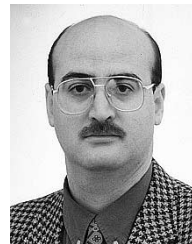
Mr. Hannuksela has been an active participant in the ITU-T Video Coding Experts Group since 1999 and in the Joint Video Team of ITU-T and ISO/IEC since its foundation in 2001.



Ye-Kui Wang (M'02) received the B.S. degree in industrial automation in 1995 from the Beijing Institute of Technology, Beijing, China, and the Ph.D. degree in electrical engineering in 2001 from the Graduate School at Beijing, University of Science and Technology of China.

He is currently a Senior Design Engineer with Nokia Mobile Software, Tampere, Finland. From 2001 to 2002, he was a Senior Researcher with the Tampere International Center for Signal Processing, Tampere University of Technology. He has co-authored over 40 technical contributions to JVT, VCEG, and MPEG, and 18 academic papers. His research interests mainly focus on video coding and communications.

Dr. Wang has been an active participant in the Joint Video Team of ITU-T VCEG and ISO/IEC MPEG.



Moncef Gabbouj (M'85–SM'95) received the B.S. degree in electrical engineering in 1985 from Oklahoma State University, Stillwater, and the M.S. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, in 1986 and 1989, respectively.

He is currently a Professor and Head of the Institute of Signal Processing, Tampere University of Technology, Tampere, Finland. From 1995 to 1998, he was a Professor with the Department of Information Technology, Pori School of Technology and Economics, Pori, Finland, and, during 1997 and 1998, he was on sabbatical leave with the Academy of Finland. His research interests include nonlinear signal and image processing and analysis, content-based analysis and retrieval and video coding. He was co-guest editor of the *European Journal of Applied Signal Processing*, special issues on Multimedia Interactive Services (April and June 2002) and *Signal Processing*, special issue on nonlinear digital signal processing (August 1994). He is co-author of over 200 publications.

Dr. Gabbouj is the Chairman of the IEEE-EURASIP NSIP (Nonlinear Signal and Image Processing) Board. He is currently the Technical Committee Chairman of the EC COST 211quat. He served as associate editor of the *IEEE TRANSACTIONS ON IMAGE PROCESSING*. He is the chairman of the IEEE Finland Section and past chair of the IEEE Circuits and Systems (CAS) Society, TC DSP, and the IEEE Signal Processing/CAS Finland Chapter. He was also the TPC Chair of EUSIPCO 2000 and the DSP track chair of the 1996 IEEE ISCAS and the program chair of NORSIG'96. He is also member of EURASIP AdCom. He was co-recipient of the Myril B. Reed Best Paper Award from the 32nd Midwest Symposium on Circuits and Systems and co-recipient of the NORSIG 94 Best Paper Award from the 1994 Nordic Signal Processing Symposium. He was the prime investigator in several EU research and educational projects and Auditor of a number of ACTS and IST projects on multimedia security, augmented and virtual reality, image and video signal processing.