51. IWK

Internationales Wissenschaftliches Kolloquium
International Scientific Colloquium

PROCEEDINGS 11-15 September 2006

FACULTY OF ELECTRICAL ENGINEERING AND INFORMATION SCIENCE

INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING - DEVICES AND SYSTEMS, MATERIALS AND TECHNOLOGIES FOR THE FUTURE

Startseite / Index:
http://www.db-thueringen.de/servlets/DocumentServlet?id=12391

TECHNISCHE UNIVERSITÄT ILMENAU

Impressum

Herausgeber: Der Rektor der Technischen Universität Ilmenau
Univ.-Prof. Dr. rer. nat. habil. Peter Scharff

Redaktion: Referat Marketing und Studentische Angelegenheiten
Andrea Schneider

Fakultät für Elektrotechnik und Informationstechnik
Susanne Jakob
Dipl.-Ing. Helge Drumm

Redaktionsschluss: 07. Juli 2006

Technische Realisierung (CD-Rom-Ausgabe):
Institut für Medientechnik an der TU Ilmenau
Dipl.-Ing. Christian Weigel
Dipl.-Ing. Marco Albrecht
Dipl.-Ing. Helge Drumm

Technische Realisierung (Online-Ausgabe):
Universitätsbibliothek Ilmenau
ilmedia
Postfach 10 05 65
98684 Ilmenau

Verlag: isle
Verlag ISLE, Betriebsstätte des ISLE e.V.
Werner-von-Siemens-Str. 16
98693 Ilrnenau

ISBN (Druckausgabe): 3-938843-15-2
ISBN (CD-Rom-Ausgabe): 3-938843-16-0

Startseite / Index:
http://www.db-thueringen.de/servlets/DocumentServlet?id=12391

51st Internationales Wissenschaftliches Kolloquium
Technische Universität Ilmenau
September 11 – 15, 2006

Thomas Rathgen, Axel Rehse, Peter Amon, Andreas Hutter

Error Recovery in File Streaming Scenarios Using Dynamic Updates

Introduction

Currently, video transmission over IP networks gets accepted in a wide area of applications such as IPTV or mobile video communication. Wireless network connections are available enabling simple network communication. On the other hand, these connections usually do not offer techniques to ensure a particular level of Quality of Service. These data packets may get lost or may be delayed due to network or router congestions or due to transmission errors.

In recent video coding schemes such as MPEG-4 Visual (Advanced) Simple Profile [4], the video can be coded to be transmitted on a packet basis, so easy resynchronization is possible in case of a packet loss. A decoder may continue decoding the bit stream but needs to interpolate the lost data, which leads to errors in the resulting output picture. These errors propagate in ensuing pictures (Figure 1) due to closed loop coding, until the erroneous picture area is not used as reference any longer.

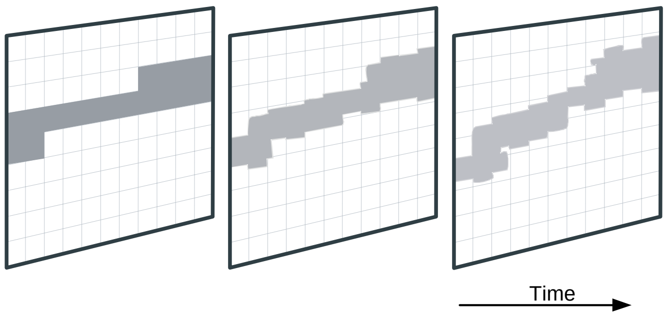

Figure 1 Error propagation due to lost data

The light green area in the left picture of Figure 2 illustrates macro blocks that have to be interpolated due to a packet loss. In this example, the decoder interpolates the lost area by copying affected areas form the decoded picture buffer of the previous picture. The right side shows the output of this example, seven pictures after the transmission error occurred.

Figure 2 Lost macro blocks (a) and error propagation (b)

Error Resilience Tools

MPEG-4 Visual (Advanced) Simple Profile contains a set of error resilience tools to handle such errors. Among others there are tools to prevent error propagation, if parts of the stream could not be decoded correctly.

A simple approach, which is also used in earlier video coding schemes, periodically inserts intra coded pictures (I-pictures). Since I-pictures are not predicted from other pictures, these coded pictures need a significantly higher amount of data. In principle, an I-picture is needed only at the beginning of the coded video sequence. In practice, a shorter I-picture interval is used depending on the motion prediction and transformation accuracy and, especially, depending on usage requirements. To reduce the data rate the longest I-picture interval possible is to be preferred.

Additionally MPEG-4 Visual offers the adaptive intra refresh tool (AIR, MPEG-4 Visual Annex E [4]). AIR generates a refresh map to identify regions containing motion. In these regions a certain amount of macro-blocks is coded using I-blocks in each picture. After a defined number of pictures all macro-blocks in this regions have been refreshed by I-blocks, while this number of pictures may be smaller than the number of pictures between I-pictures. As for inclusion of I-pictures, AIR increases the bit-rate but avoids bit-rate peaks.

The insertion of I-pictures or I-blocks introduces simple but efficient error concealment techniques. In live encoding scenarios such as video conferencing, a decoder could signal errors to the encoder, assuming the network provides a feedback channel, and trigger the insertion of I-pictures or I-blocks. [1] and [2] describe, how to dynamically insert I-blocks to set new references, if a transmission error was detected. The encoder receives a report signaling the frame number and the macro-block positions, which were detected as being lost. This information is used to calculate, which macro-blocks in the next frame to be encoded are directly or indirectly predicted from the lost area. At these positions the encoder uses I-blocks to encode the picture (marked macro-blocks in the right most picture of Figure 3) which enables efficient error recovery. Since I-blocks need more bits to be encoded, the data rate increases, if additional I-blocks are inserted. Alternatively, the quantization may be adjusted to fit a given target bit rate. In this case, a coarser quantization is used, which decreases visual quality.

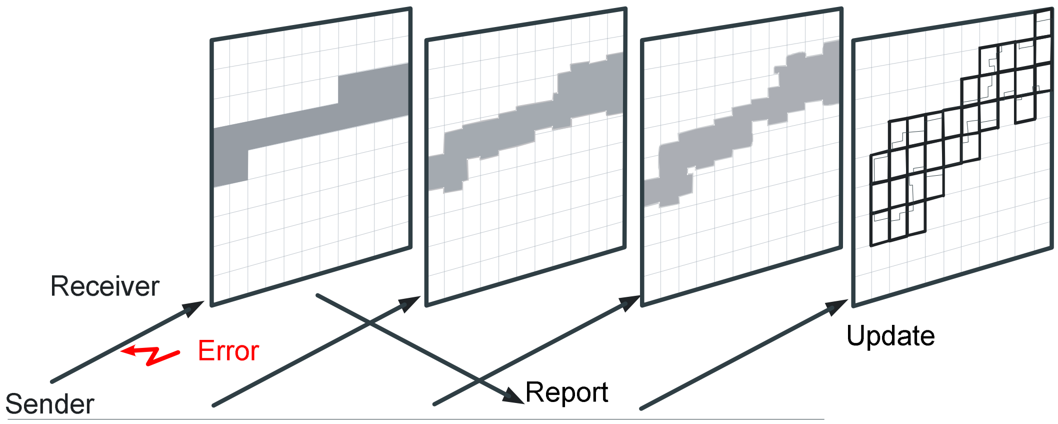

Figure 3 Insertion of I-blocks at affected positions

MPEG-4 Visual defines another tool for networks with feedback channel. The tool NEWPRED also uses reports to explore, which macro-blocks are detected as being lost. This information is used to calculate, which macro-blocks in the next frame to be encoded are directly or indirectly predicted from the lost area. Then the encoder may choose a different reference picture, which is known to be decoded correctly for prediction. This decision might be done on video object or on packet basis. Even if prediction from earlier pictures is less efficient, this tool reduces the amount of data needed to correct missing references in the encoder but requires additional memory in the decoder.

NEWPRED and the I-block insertion mentioned above can be seen as closed loop error control techniques. They need an upstream channel on the network but do not consume additional bandwidth in the error-free case. Additional bandwidth is only needed in case of an error. Nevertheless, no additional delay is added, because the erroneous areas are corrected after a negative report has been received and interpolated pictures are displayed, until the update information is received.

I-block insertion for file streaming

Both tools described above, NEWPRED and the insertion of I-blocks, are designed for live encoding. For use in file streaming scenarios these techniques must be modified to prevent any re-encoding. One approach is the adaptation of the I-block insertion technique for file streaming of MPEG-4 Visual Simple Profile files, where two separate media files are stored on the server, one containing the conventionally coded video stream, the other containing the same video with all frames coded using I-blocks only. Note, that the I-block only file uses the same GOP structure as the original coded stream. As described above, the decoder announces lost macro-blocks (e.g. due to packet losses) to the server. A meta data file on the server contains pre-computed results of a motion vector analysis, so the server can easily compute, which macro-blocks in the next frame to be encoded are directly or indirectly predicted from the lost area. At these positions the server switches to the second stream, causing the inclusion of I-blocks into the output stream (see the insertion switch in Figure 4).

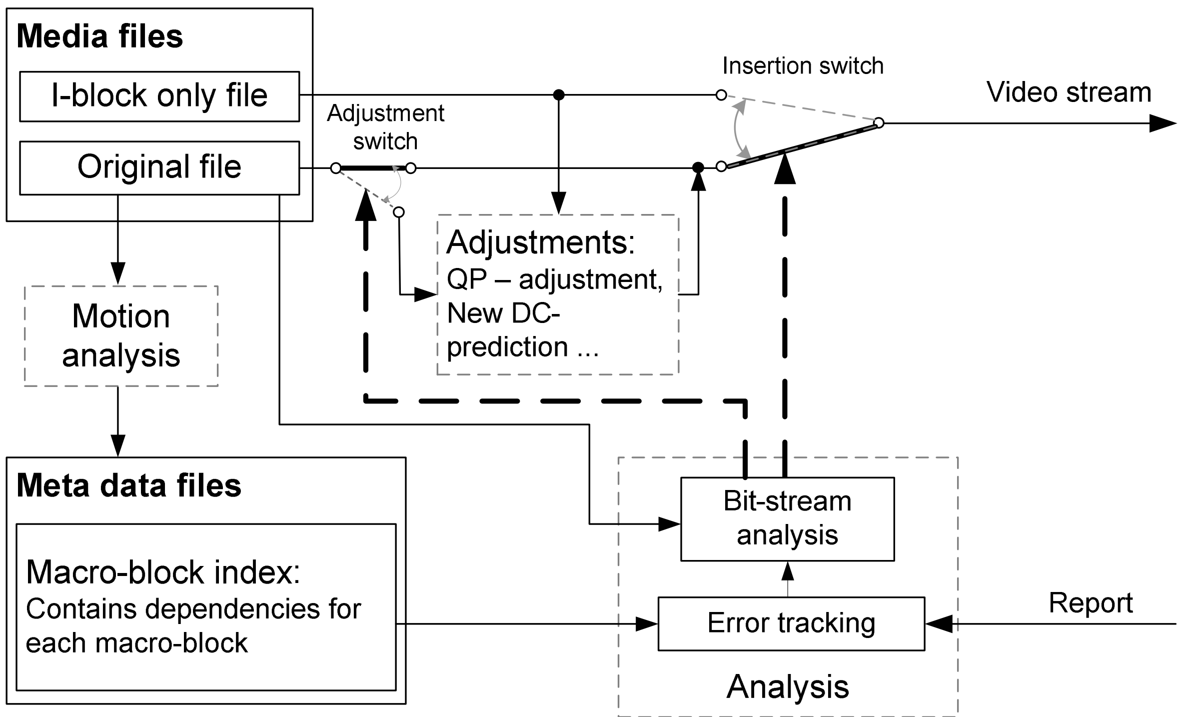

Figure 4 I-block insertion for file streaming – system diagram

Since it is very likely that macro-blocks from the stream containing only I-blocks use a different quantization scale, the current quantization scale needs to be set, before any switching action by including shortened video packet headers (see adjustment box in Figure 4). Furthermore, for surrounding I-blocks form the original coded stream, DC prediction might be adjusted.

Insertion of I-blocks from a different coded stream is expected to perform less accurate as encoder controlled I-block insertion in live encoding schemes, because macro-blocks coded with different parameters are inserted in the decoder loop.

Figure 5 Lost macro blocks (a), error propagation (b) and resulting picture with I-block insertion (c)

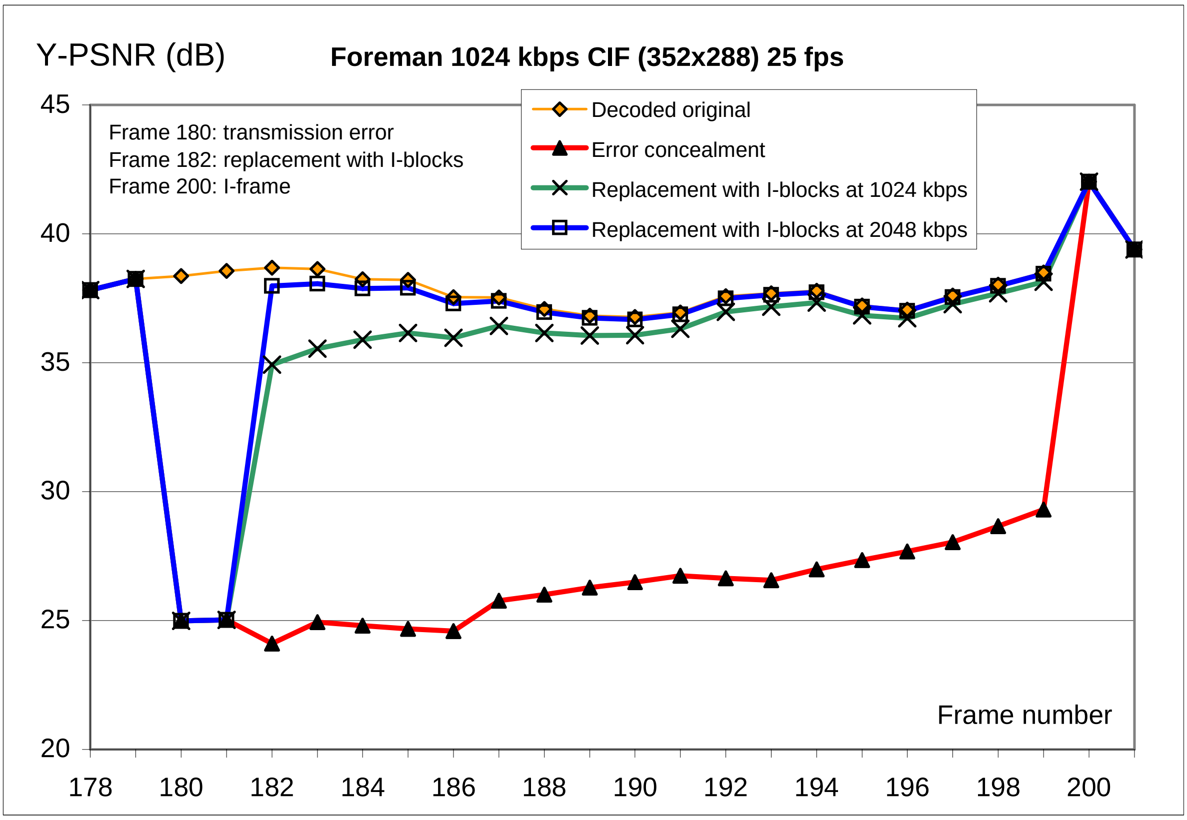

Figure 6 I-block insertion for file streaming – experimental results

Figure 5 and Figure 6 show experimental results of a first implementation, where all macro blocks which depend on the lost area are replaced by I-blocks. In the example an error (see Figure 2) occurs in frame 180 of FOREMAN. Interpolation is done by the decoder by copying from the decoded frame buffer of the previous frame. A report is received immediately and new references are inserted in frame 182. Figure 5 illustrates lost macro blocks (a) and compares the interpolated output of frame 187 (b) with the output after inserting I-blocks at 2048 kbps (c).

Figure 6 depicts the Y-PSNR of different settings for I-insertion for FORMAN (CIF at 1024 kbps) with interpolation only by the decoder (red curve), I-block insertion with I-blocks at 1024 kbps (green curve), I-block insertion with I-blocks at 2048 kbps (blue curve) and error-free sequence (orange curve).

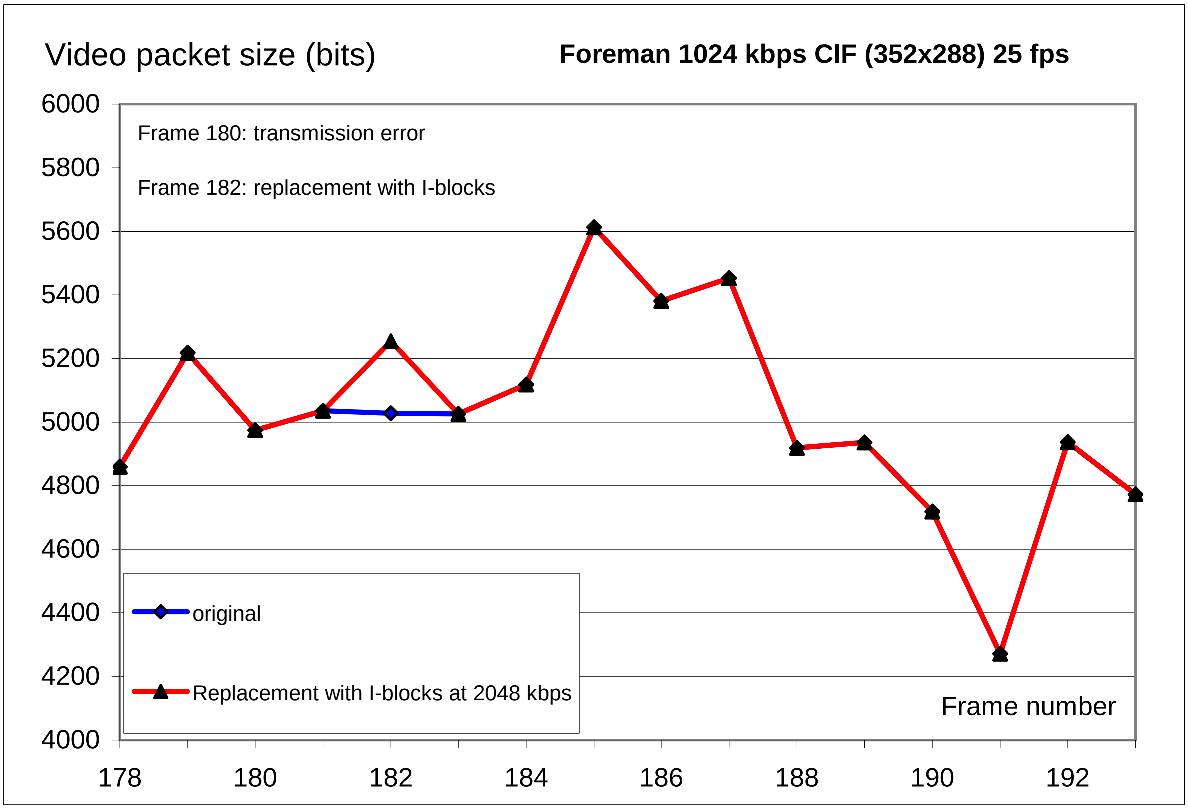

Figure 7 Video packet sizes – experimental results

These results show an improvement of up to 13.88 dB for FOREMAN with a slightly increased bit rate at frame 182 (Figure 7). However, tests with other sequences show quality degradation, if I-blocks were inserted.

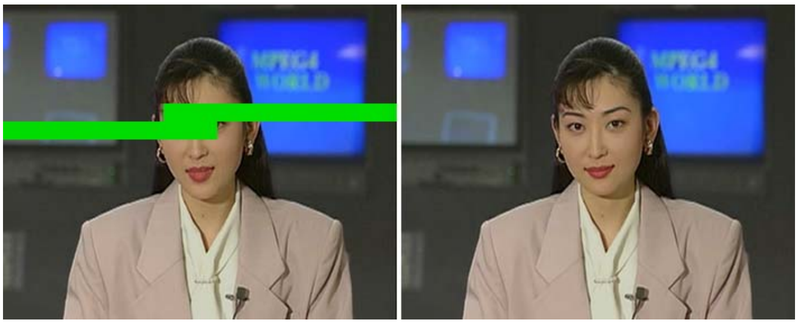

Figure 8 Lost macro-blocks (left) and result of interpolation (right)

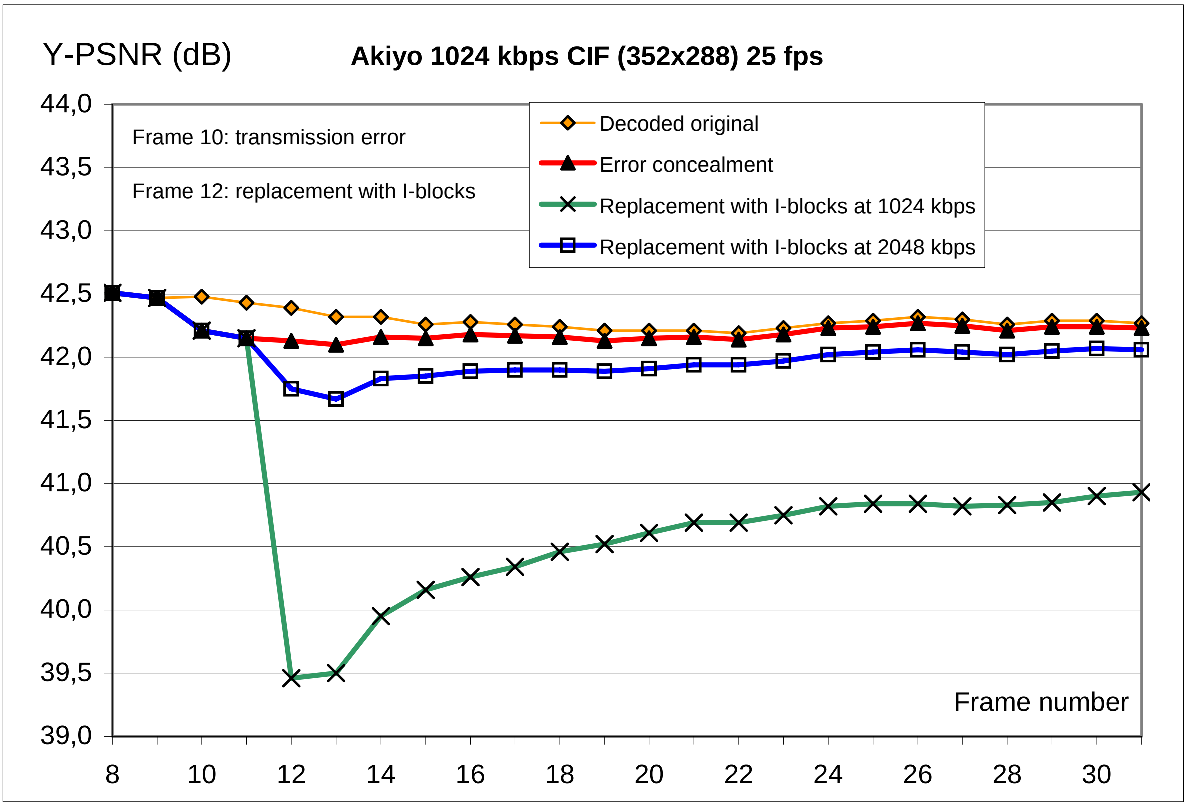

Figure 9 Experimental results for AKIYO

Figure 8 illustrates lost macro-blocks in AKIYO and the result of simple error concealment by the decoder (copying from the last recent decoded picture buffer). As expected visual quality decreases, if affected macro-blocks are replaced by I-blocks (Figure 9) because I-blocks with coarser quantization are inserted.

Error estimation and insertion control

Experiments yield the assumption that the I-block insertion technique described above can be improved by incorporating analyzing and controlling mechanisms. Figure 10 demonstrates, how a decoder could interpolate lost macro-blocks of CARPHONE QCIF at 250 kbps (sequence PSNR 37 dB) by copying from the decoded picture buffer.

Figure 10 Lost macro-blocks (left), interpolated picture (middle) and amplified difference decoded original vs. interpolated picture (right)

Figure 11 Decoded original (left), interpolated picture (middle) and amplified difference decoded original vs. interpolated picture (right); each 10 frames after the transmission error

Figure 12 Decoded original (left), output with inserted I-blocks (middle) and amplified difference decoded original vs. output with inserted I-blocks interpolated picture (right); each 10 frames after the transmission error

Figure 11 shows the output 10 frames after the error occurred, while Figure 12 illustrates the output, if all affected macro-blocks are replaced by I-blocks at 512 kbps (sequence PSNR 33 dB). Comparing the right pictures of Figure 11 and Figure 12 reveals that insertion of I-blocks may decrease the visual quality, e.g. the erroneous area may expand.

Finally, for this example, Figure 13 shows, which macro-blocks are optimally to be replaced by I-blocks to minimize the error for a given total bit rate.

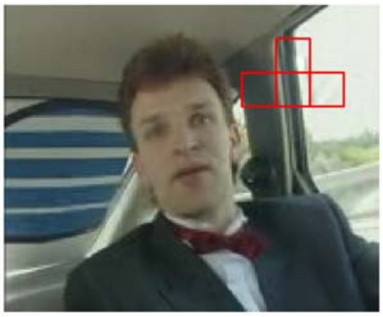

Figure 13 Optimal block selection

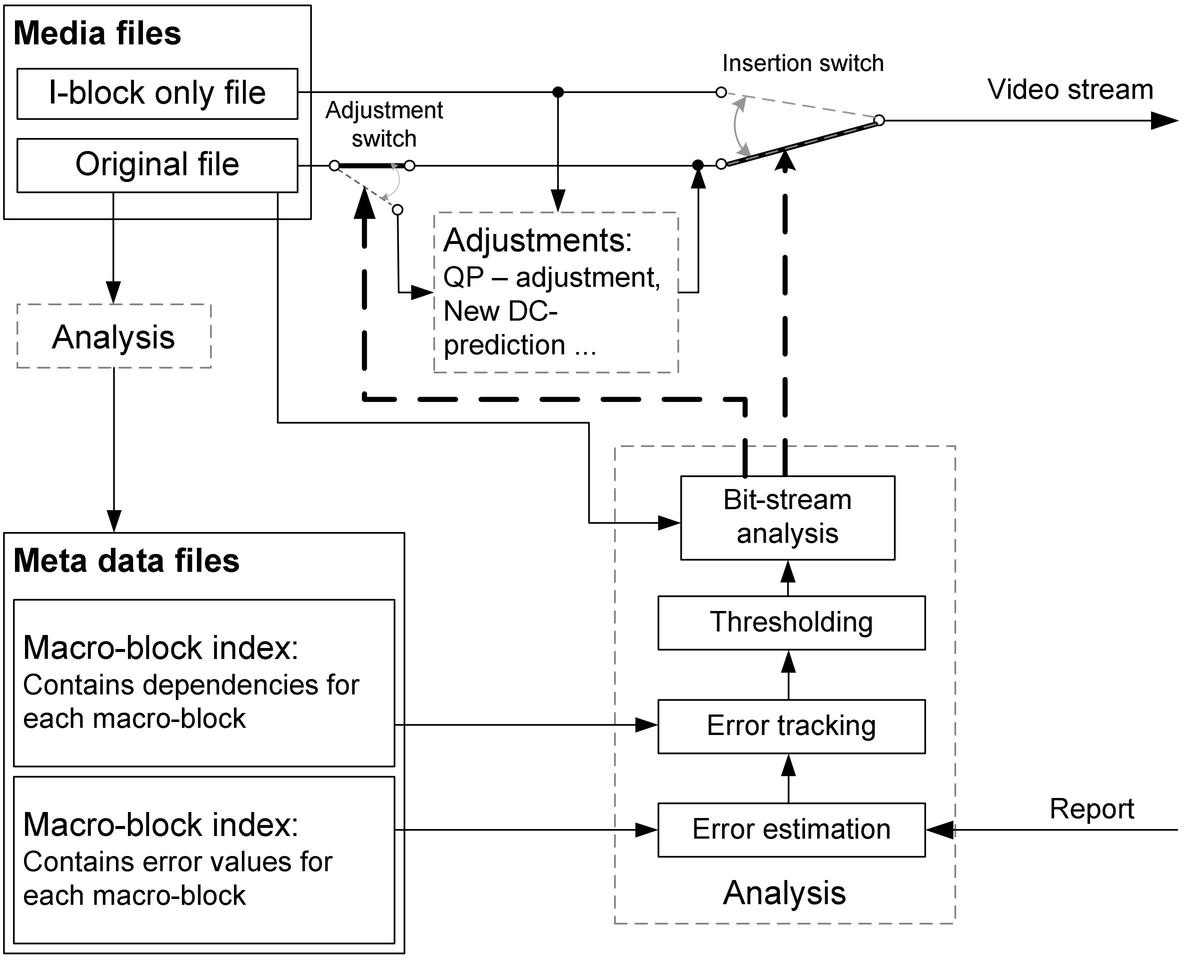

Figure 14 I-block insertion for file streaming – extended system diagram

The system in shown in Figure 4 is extended by additional analysis and estimations tools (Figure 14), which decide, whether a certain macro-blocks is to be replaced by an

I-block. All meta information used for error estimation and error tracking is stored in separate meta data files; so again all calculations do not require much computational power.

There are some features, which can be used for error estimation. Various decoder strategies exist to interpolate lost data (most of which are non-normative), in these experiments the decoder interpolates lost data by copying from the decoded picture buffer. If the interpolation strategy of a decoder is known, the error can be calculated while encoding the bit stream by calculating MSE for each macro block in both streams. While it is expected to get best result when using MSE (in case the interpolation strategy is know), the following simple approximation approaches to estimate the error were considered also. The estimated error for each macro-block has been rated, the insertion decision has been made by thresholding this rating:

- If an I-block has been lost then a great error must be expected. Therefore the error rating is equal to the maximum value.
- If Inter-coded macro-blocks are lost, then motion information and texture information is missing.
    - Blocks which are *not-coded* (for a given bit rate, there is no difference to the previous frame to be coded) contain no information. The more *coded* blocks are contained in a macro-block the more texture information is lost and the higher is the rating.

      $$F_n \approx 2^{nc-1} \cdot k_Q; \text{ with } nc > 0,\ nc \text{ - number of coded blocks in a macro-block,}$$
      $$k_Q \text{ - correction factor for quantization scale}$$

      If a macro-block gets lost with four *coded* blocks the MSE is approximately twice the MSE for a lost macro-block with three *coded* blocks. The correction factor k_Q represents an approximation of the MSE, if a macro-block with one *coded* block is lost. This feature especially might be used for low bit-rate coding.
    - Due to the entropy coding a higher number of bits in a block contains more information. The higher the number of bits of a macro-block the higher is the rating.
    - The higher the value of a motion vector of a macro-block the greater is the error caused by motion vector interpolation in the decoder (in this case, interpolation is done by setting the value to 0, which means copying from the previous picture buffer). The rating increases with the quadratic value of the motion vector length.

The estimated error also depends on the texture structure, which is represented by the correction factor k_v.

$$F_l \approx l^2 \cdot k_v; \text{ with } l^2 = x^2 + y^2 \text{ - motion vector value,}$$
$$k_v \text{ - correction factor for picture structure}$$

- Additionally, improved error tracking may be performed. If a macro-block of a previously coded picture is used for prediction, then this macro-block is marked. This is repeated in ensuing pictures, so macro-blocks might be marked by mistake, if just parts of the macro-block are used for prediction. To address this problem a macro block is rated based on the number of pixels used for reference and this may be performed on block or sub-block basis (Figure 15).

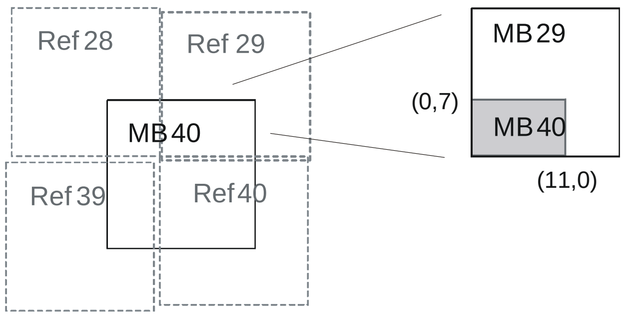

Figure 15 Dependencies between macro-blocks

$$F_m \approx \frac{i_x \cdot i_y}{i_{x\max} \cdot i_{y\max}}; \text{ with } i_x \text{ and } i_y \text{ identifying the referenced area,}$$
$$i_{x\max} \cdot i_{y\max} = 256 \text{ - total number of pixels per macro-block}$$

Special attention must be paid, if motion compensation is used with a sub-pixel value. In this case, the interpolation filter reduces high frequency components and the rating is reduced by 10 % in each step this applies.

- For ensuing pictures the area criterion F_m can be used for calculating the error rating as illustrated in Figure 16. Besides F_m also an error value for the lost macro-block is used (e.g. F_l or F_n).

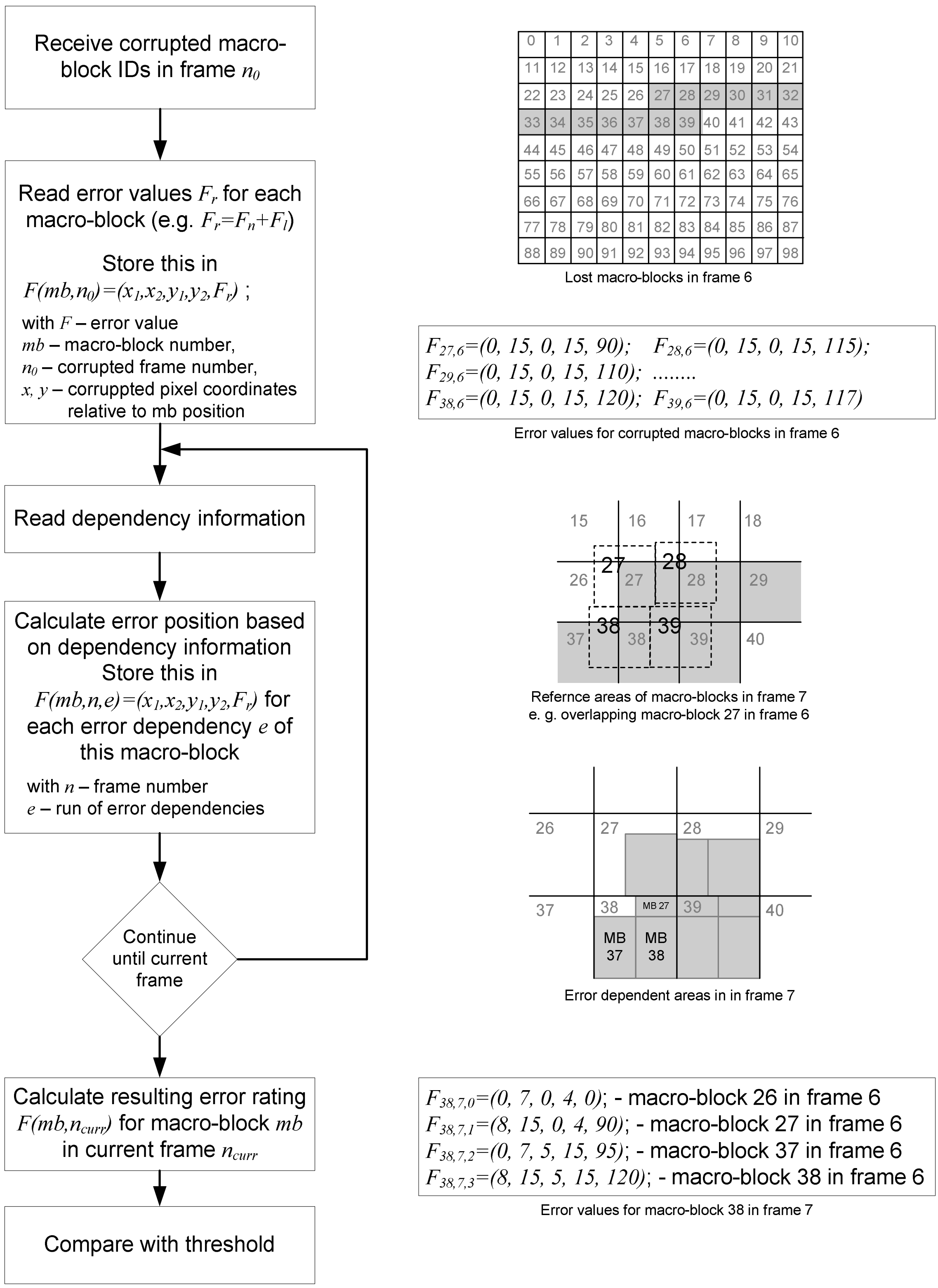

Figure 16 Calculating the error rating for ensuing pictures

Calculating the error rating of the current picture based on the initial error rating and the reference area information of the process illustrated in Figure 16 is done as follows:

$$F(mb,n) = \sum_{e=0}^{nre_{n_0}} \frac{\mathrm{P}_{ref}(mb,n,e) \cdot F_r(n,e)}{256}$$

$$P_{ref}(mb,n,e) = \sum_{x=x_{ref}}^{x_{ref}+15} \sum_{y=y_{ref}}^{y_{ref}+15} f_{err}(x,y) \ ;$$

$$\text{with } f_{err}(x,y) = \begin{cases} 1 & \text{if pixel at position(x,y) directly or indireclty depends on error } e_{mb} \\ 0 & \text{otherwise} \end{cases}$$

$$F_r(n,e) = F_r(n-1,e) \cdot k_{filter} \ ;$$

$$\text{with } k_{filter} = \begin{cases} [0,1) & \text{if filter is used during decoding} \\ 1 & \text{otherwise} \end{cases}$$

n - current frame number

mb - current macro-block number

e - run of erroneous macro-blocks

n_0 - erroneous frame

e_{mb} - erroneous macro-blocks in n_0

nre_{n_0} - number of erroneous macro-blocks in n_0

P_{ref} - number of affected pixels in reference area for mb in n

F_r - rated error value

$F_r(n_0, e)$ - rated error value of erroneous macro-blocks in n_0

- Also, the number of affected pixels in the macro-block to be replaced may be used as a simple criterion for error estimation.

Figure 17 shows experimental results for CARPHONE. Figure 18 shows the same results but the difference of each error control approach to the decoded original. If advanced selection was performed, each criterion performs better than the 'replace all affected positions' approach.

Furthermore, mean square error (MSE) based selection for frame 11 gives best results on this frame. On the other hand, the proposed process in Figure 16 comes close to the MSE based selection. For pre-encoded files this enables simple error rating if no additional encoder information is available or if the interpolation strategy of the decoder is not known.

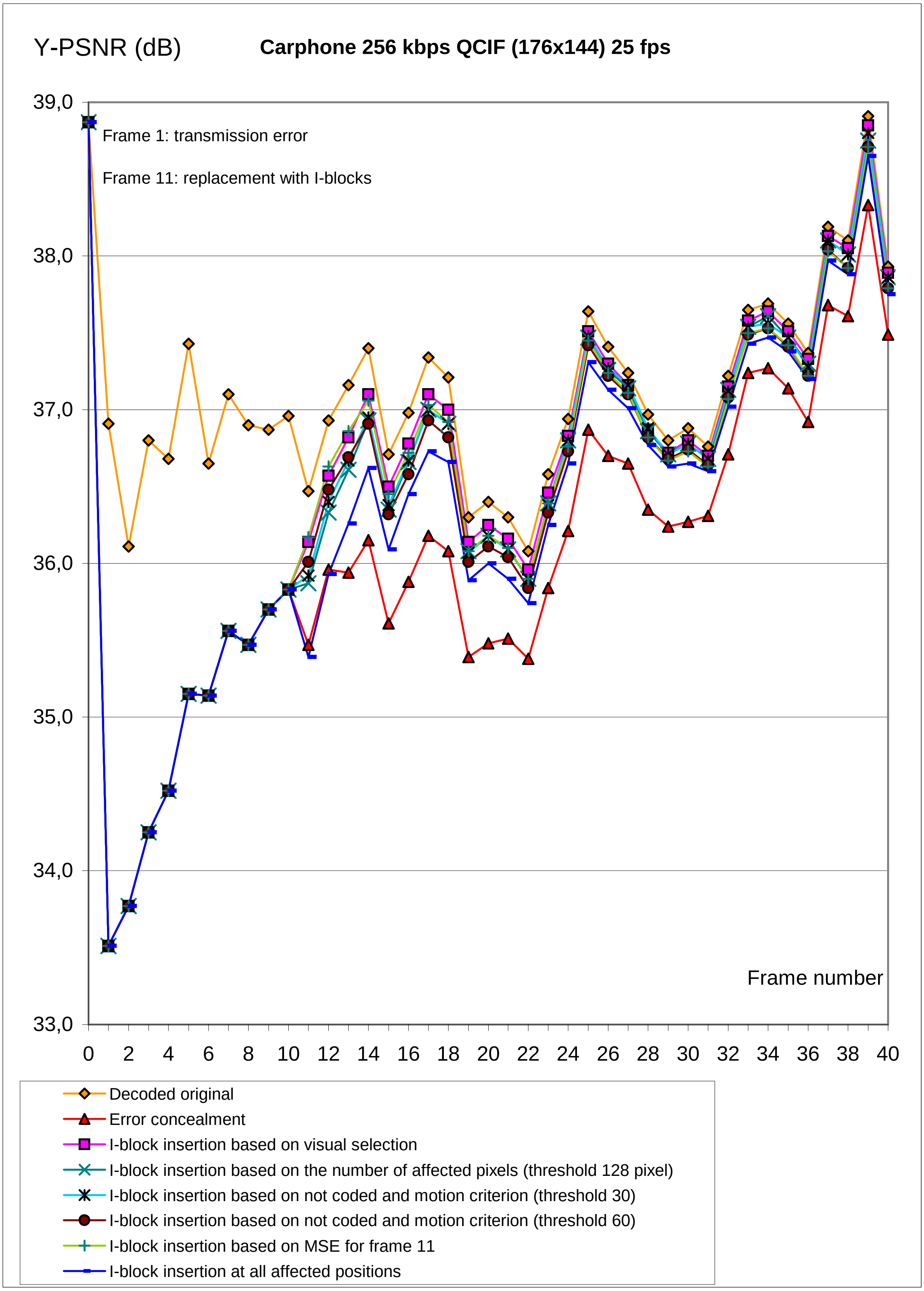

Figure 17 PSNR for different rating criteria

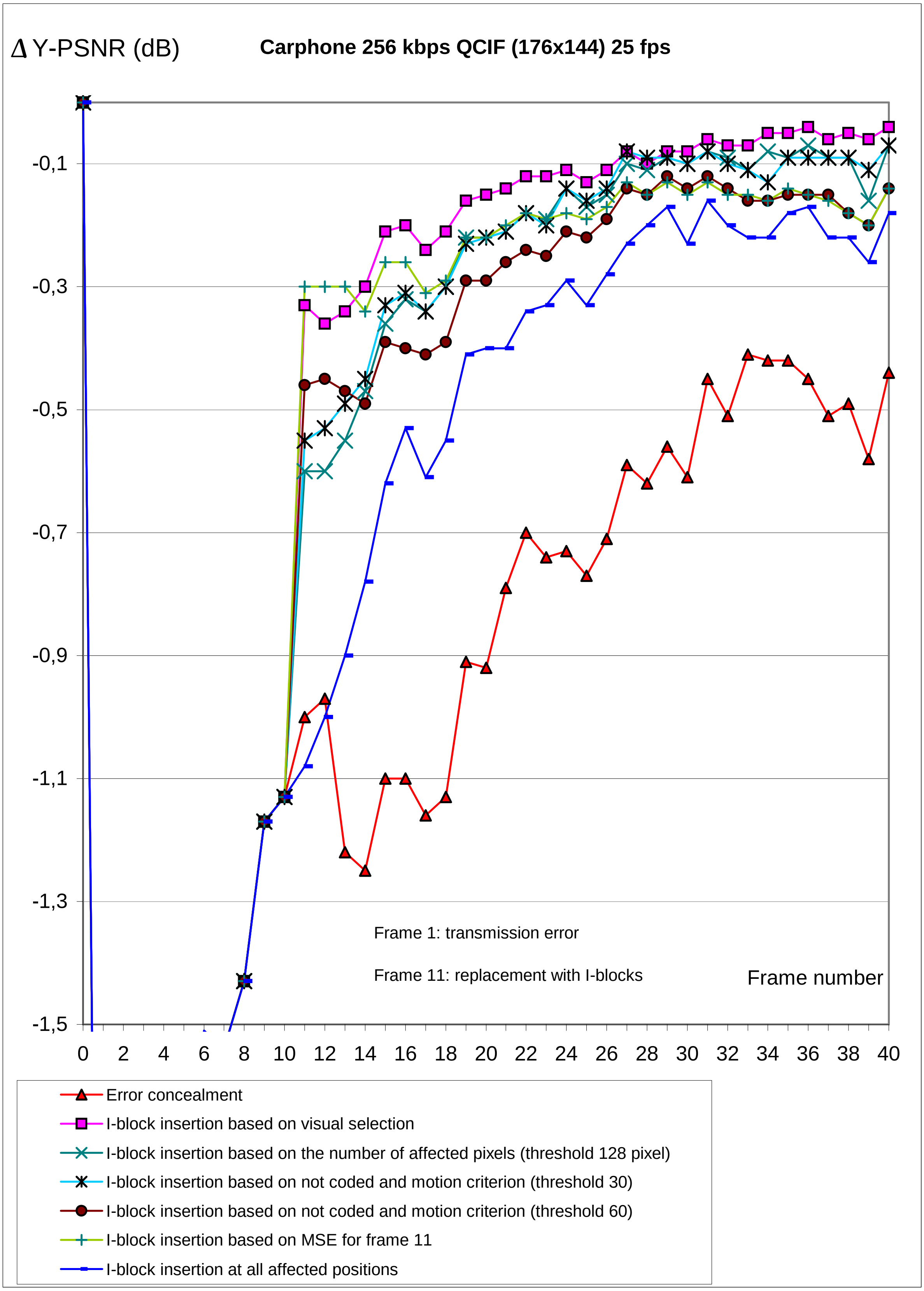

Figure 18 Δ PSNR (different error control techniques vs. decoded original)

For higher bit rates the quality improvement increases as shown in Figure 19 and Figure 20. Again, all approaches to control I-block insertion with advanced error rating techniques perform better than replacing all affected macro blocks.

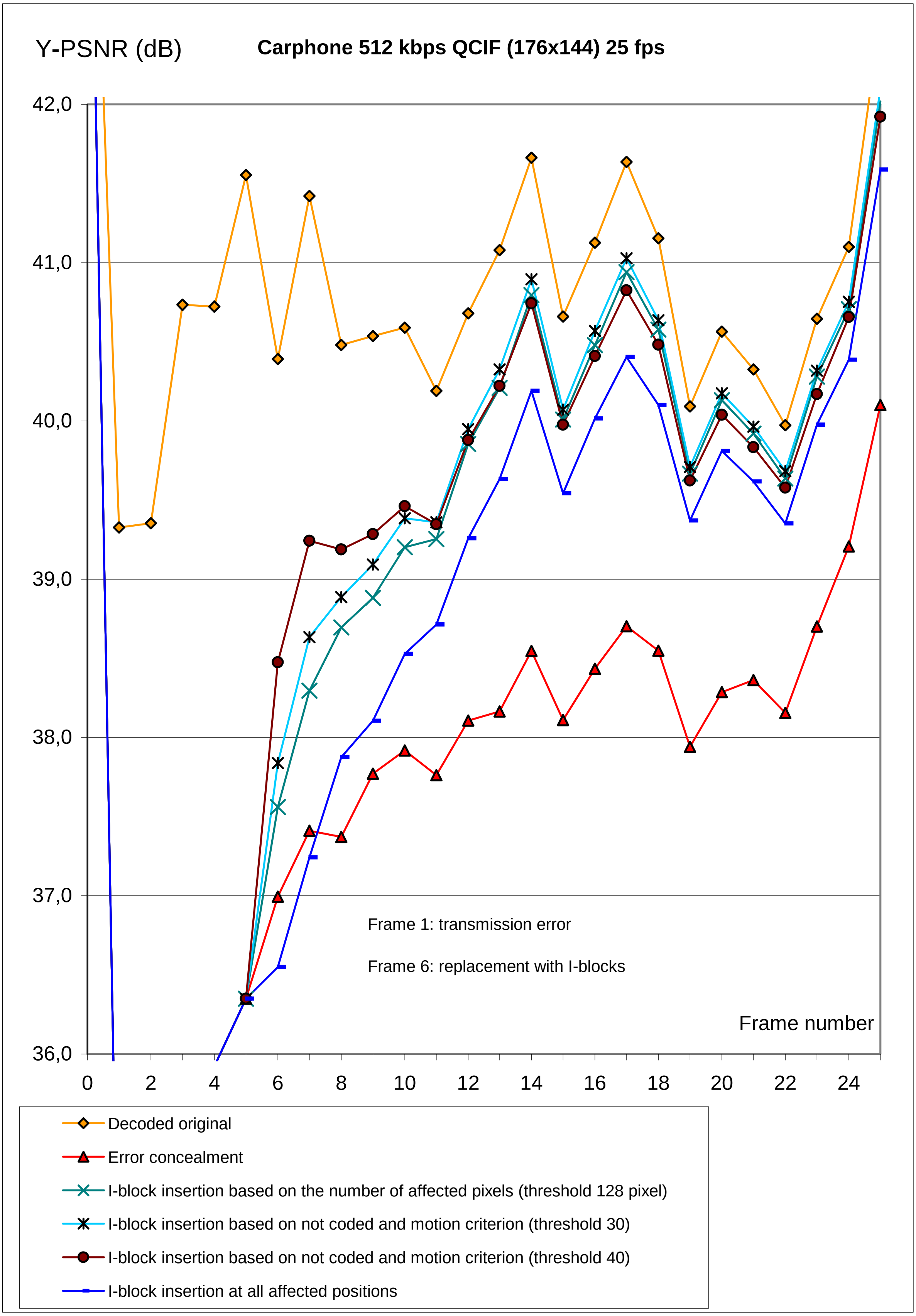

Figure 19 PSNR for different rating criteria

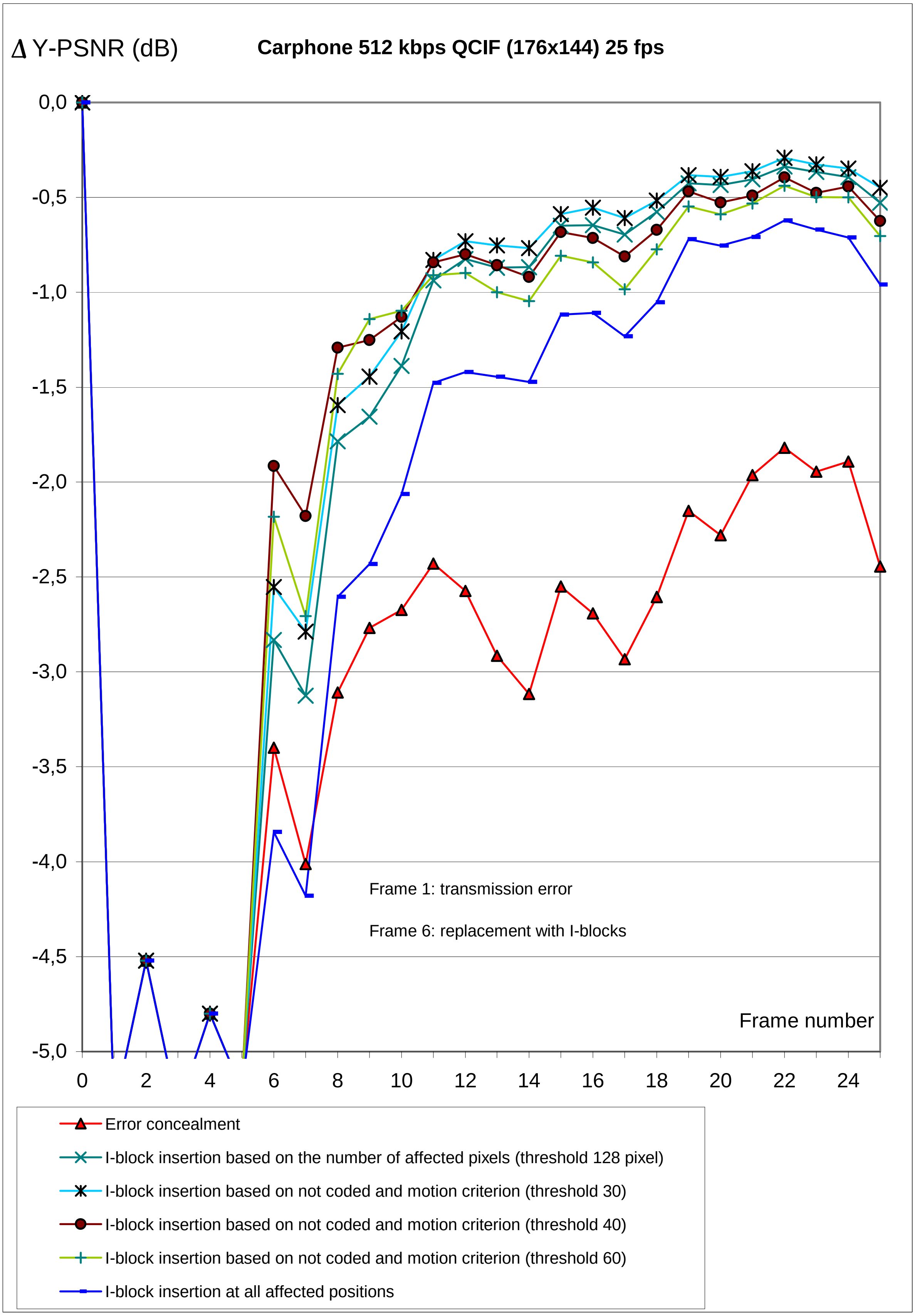

Figure 20 Δ PSNR (different error control techniques vs. decoded original)

In the experiments the macro-block analysis was done using the process illustrated in Figure 16. The initial error rating is calculated using the number of coded blocks (not coded criterion) and the motion vector value (motion criterion):

$$F_r(n_0,e) = F_{nl}(n_0,e) \approx 2^{nc-1} \cdot k_Q + l^2 \cdot k_v \text{ ; with } nc > 0$$

e - run of erroneous macro-blocks

n_0 - erroneous frame

nc - number of coded blocks in a macro-block,

k_Q - correction factor for quantization scale,

$l^2 = x^2 + y^2$ - motion vector value,

k_v - correction factor for picture structure

$F_r(n_0,e)$ - rated error value of erroneous macro-blocks in n_0

Conclusion

This paper shows how video quality can be improved by inserting new references in case of a transmission error, assuming a feedback channel is available on the network. This applies for live-encoding [1], [2] and also for file streaming scenarios. Experiments show that the proposed technique for update erroneous or missing references with I-blocks in case of file streaming can be improved by analyzing the bit stream, so only macro blocks with a great impact on visual quality are replaced by I-blocks. The proposed techniques are likely to apply also for MPEG-4 Advanced Video Coding (AVC).

Acknowledgement:
This work has been co-funded by IST European project PHOENIX under contract FP6-2002-IST-1-001812.

References:

[1] Girod, B.; Färber, N.: Feedback-Based Error Control for Mobile Video Transmission, *Proceedings of the IEEE, Special Issue on Video for mobile Multimedia*, 97(10) Oct. 1999, S.1707-1723

[2] Steinbach, E.; Färber, N.; Girod, B.: Standard Compatible Extension of H.263 for Robust Video Transmission in Mobile Environments, *IEEE Trans. Circuits and Sys. for Video Technology*, 7(6) Dec. 1997, S.872-881

[3] Rathgen, Th.; Rehse, A.; Piesche, F.: Dynamische Makroblock-Ersetzung als Fehlerbehandlung beim File-Streaming, *Posterbeitrag zum 3. Thüringer Medienseminar der FKTG*, Erfurt, July 2005

[4] ISO/IEC 14496-2: Information Technology, Coding of Audio- Visual Objects, Part 2: Visual, Third Edition, 2003

Authors:

Thomas Rathgen
Ilmenau Technical University, Faculty of Electrical Engineering, P.O. Box 100565
98684 Ilmenau, Germany
Phone: +49 3677 69 2603
Fax: +49 3677 69 1163
E-mail: thomas.rathgen@tu-ilmenau.de

Axel Rehse
Ilmenau Technical University, Faculty of Electrical Engineering, P.O. Box 100565
98684 Ilmenau, Germany
E-mail: axel.rehse@stud.tu-ilmenau.de

Peter Amon
Siemens AG, Corporate Technology, CT IC 2, Otto-Hahn-Ring 6
81730 München, Germany
Phone: +49 89 636 54642
Fax: +49 89 636 51115
E-mail: p.amon@siemens.com

Dr. Andreas Hutter
Siemens AG, Corporate Technology, CT IC 2, Otto-Hahn-Ring 6
81730 München, Germany
Phone: +49 89 636 50417
Fax: +49 89 636 51115
E-mail: andreas.hutter@siemens.com