Wavelet Based Erasure-Resilient Video Transmission System

Joachim M. Buhmann* Marek Karpinski† Yakov Nekrich‡

Abstract

Robustness of video transmission systems against packet losses is an indispensible requirement for high performance communication networks. This paper describes a system for erasure-resilient transmission of a video data stream over packet-based networks. The video codec combines embedded wavelet compression, enhanced by a block based motion prediction scheme with the priority encoding technique (PET). Such a system which encodes packets with different priority levels by multiple Reed-Solomon codes is protecting the video stream against packet losses. The priority levels are adapted to the bit allocation of the wavelet bands. Video trasmission experiments demonstrate that the proposed video system can be efficiently employed for erasure-resilient transmission of video data.

Keywords: Video Transmission, Video Compression, Unequal Error Protection, Forward Error Correction, Priority Encoding Transmission, Wavelet Compression, Set Partitioning into Hierarchical Trees.

1 Introduction

Video transmission over packet-based networks is an expanding research field with many industrial applications. Video-on-demand services over the Internet as well as applications in tele-medicine require video transmission

---

[*] Institute for Computational Science, ETH Zentrum, Zurich. Work partially supported by a BMBF grant 01 M 3021 A/4 and by Infineon AG. E-mail `jbuhmann@inf.ethz.ch`

[†] Dept. of Computer Science, University of Bonn. Work partially supported by DFG grants , Volkswagen Foundation, PROCOPE project and IST grant 14036 (RAND-APX). E-mail `marek@cs.uni-bonn.de`

[‡] Dept. of Computer Science, University of Bonn. Work partially supported by IST grant 14036 (RAND-APX). E-mail `yasha@cs.uni-bonn.de`

systems with robustness to packet losses. The most common method of protection against packet losses is re-transmission of the lost packets. However this approach is unacceptable for applications, in which real-time transmission is required as well as for multicasting applications with a multitude of receivers. For these kinds of applications Forward Error Correction (FEC) methods are preferable.

It has been shown (see, for instance, [B93]) that packet losses on the Internet can occur at high rates, leading to dramatic decrease in quality of received video. Due to the unpredictable character of packet losses, it is impossible to choose the rate of the FEC code, that will provide acceptable image quality. Therefore in real-time video transmission applications we need a loss protection system, that will provide a *graceful degradation* property. This property guarantees that the quality of transmitted data smoothly decreases with a growing rate of packet losses. Therefore, an efficient multimedia transmission system should contain two components: an efficient rate-scalable codec and an efficient loss protection algorithm, that assigns different degrees of loss protection (and redundancy) to different parts of data.

To fulfil these requirements we will describe a combination of a wavelet-based video compression method with an efficient loss protection scheme. A state-of-the-art codec based on wavelet decomposition of images and block-based motion compensation is employed for video compression. The two main components of our loss protection system are a Priority Encoding Transmission (PET) algorithm ([ABE+94]) and a Cauchy-based implementation of Reed-Solomon codes (see [BKK+95]) . The PET system [ABE+94] assigns different degrees of redundancy $\rho(i)$ to different parts of the data, further called blocks. These degrees of redundancy can be specified by the user. Then, the information in part i can be completely reconstructed by the receiver, if at least a $\rho(i)$ fraction of packets was received.

The rest of this paper is structured as follows. In section 1.1 we will describe prior research on loss protection of video data. In section 1.2 the main idea of our approach will be described and the high-level description of the system will be given. A more detailed description of the codec will be given in the subsequent sections 2.1 and 2.2. In sections 3 and 4 we will discuss implementation details and results of experiments.

1.1 Related Work

The problem how to protect video data against losses in network transmission was considered in a number of papers. In the H.263++ standard, support for Reversible Variable Length Codes (RVLC) is provided. A data stream, encoded with RVLC codes can be decoded from both directions, because codewords of an RVLC code are symmetric. Since there are minimum-redundancy reversible codes, RVLC codes can be used instead of classic Huffman codes without decreasing compression ratios. This method leads to a certain robustness against bit errors without increasing data redundancy and it can be applied to all situations, in which Huffman codes are used.

Several papers discuss varying degrees of protection and how they are applied to different parts of video stream. Various degrees of protection were assigned to the different types of MPEG frames in [S95]. The author has used a PET system ([ABE+94]) to reduce the total expansion of the data stream after the loss protection. A more detailed description of the PET system and its applications to video compression will be given in section 2.2.

Boyce [B99] has introduced a HiPP (High Priority Partitioning) scheme for unequal loss protection of MPEG-2 data. The video data is split into two parts, so that I-frames, motion vectors and low frequency DCT coefficients of P-frames belong to the high priority part. Compressed B-frames and the low frequency DCT coefficients of P-frames belong to the low priority part. In [B99] only high priority part is protected against losses with Reed-Solomon Codes. In [GLM+99] unequal loss protection algorithm from [MRL99] was applied to H.263 videos. All the above methods have applied unequal loss protection to DCT-based video compression.

1.2 Main Idea of the Wavelet based PET encoder

The main contribution of this paper is the design of an efficient video system, that combines motion-compensated wavelet video compression with loss protection, based on Priority Encoding Transmission (PET) [ABE+94]. In our compression system we modify the frame prediction order, so that the encoded video data can be decoded at different frame rates without decrease in quality of decoded frames (temporal rate scalability). Our frame prediction schema distinguishes between two types of P-frames, so that frames of the first type (P1-frames) can be decoded independently of the second type P-frames (P2-frames). We assign different priority levels to P1- and P2-frames.

The compressed wavelet coefficients from I-frames are also divided into several priority levels, according to their importance for image reconstruction. By assigning different amounts of redundancy to different priority levels we achieve gradual degradation of video quality as the number of lost packets grows.

Our algorithm has the following key properties:

1. The PET algorithm is used to reduce the overall redundancy for the video data stream.
2. A modified frame prediction schema divides intra-coded frames into several classes. This separation of information allows us to achieve a trade-off between compression efficiency and robustness.
3. Different degrees of redundancy are assigned to different groups of wavelet coefficients in the I-frame as well as to different types of P-frames.

2 Algorithm Description of the Codec

2.1 Video Compression System

Video compression systems usually consist of the following parts: transform coding, motion compensation and statistical coding. Transform coding exploits spatial redundancy in individual frames and motion compensation reduces the dependencies between two or more frames.

In this paper we have chosen wavelet transform as a transform coding method since it decorrelates spatial information in the image frames. Wavelet-based coding is employed for compression of I-frames and for error values of predictive-coded frames. The wavelet transform can be characterized by two sets of filters - g/g' and h/h' where g is a high pass and h is low pass filter. The related filters g' and h' are the respective synthesis filters. By applying filters g and h first in horizontal and then in vertical direction we divide an image into four frequency bands: high-high (HH), low-high (LH), high-low(HL) and low-low(LL). These operations are recursively repeated on the LL band, resulting in a hierarchical structure, which is called the wavelet pyramid and which is shown on Figure 1 and 2. In the inverse wavelet transform we reverse the order in which filters are applied to subbands and replace analysis filters g and h with synthesis filters g' and h'.

Wavelet-based image compression methods turn out to be superior to DCT-based compression for many applications. Wavelets provide an efficient spatial decorrelation scheme which is adapted to the average power decay as a function of spatial frequency in natural images. This transform coding can also be used for efficient rate-scalable compression ([SP96] and

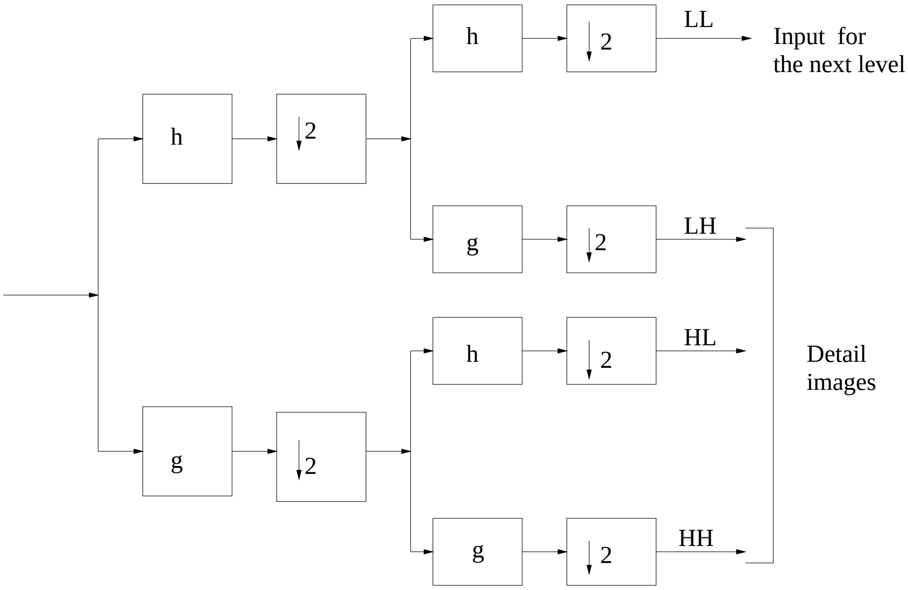

Figure 1: One level of the image wavelet transform

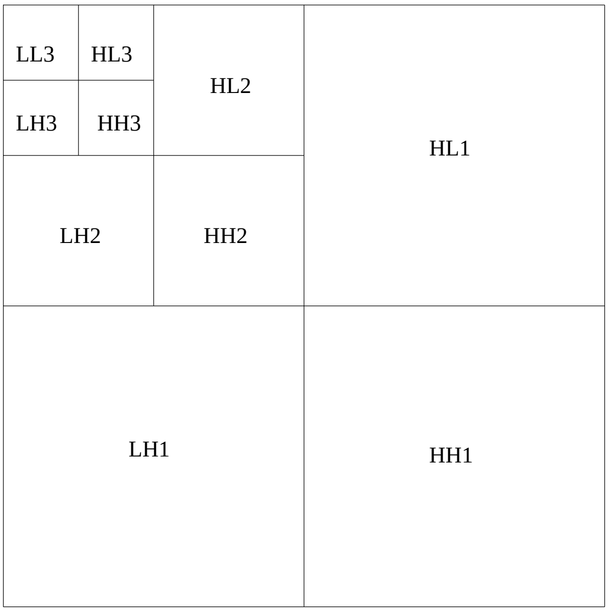

Figure 2: Pyramid structure of a wavelet transformed image

[S93]). In order to achieve rate-scalability we have to code wavelet coefficients in such a way, that important information is transmitted before image details. This goal can be achieved if wavelet coefficients with large magnitude are transmitted first. Besides that, most significant bits of each coefficient should be encoded before less significant bits. Therefore, multiple passes through the data are necessary. Below we will give a short sketch of an approach to rate-scalable wavelet compression (see also [SP96] and [S93]).

Let us first consider a classification scheme to determine when Wavelet coefficients are defined as significant. We start by setting the threshold value T to $v_{max}/2$, where v_{max} is the maximal value of a wavelet coefficient. Coefficients, whose value is larger than the threshold value T are classified as significant, their coordinates are transmitted and their coefficient values are added to the list of significant coefficients. In a second pass the threshold value is lowered by a factor of 2 and the coordinates of those coefficients, that are significant with respect to the new threshold value, are encoded and transmitted. We also encode the first most significant bit, that was not encoded during the previous passes, for every coefficient in the list of significant coefficients. Then we add the coefficients, that are significant with respect to the new threshold value, to the list of significant coefficients. This process is repeated until the required data rate is achieved or the threshold value decays below a pre-defined minimal value.

Shapiro [S93] described an efficient method for encoding the coordinates of significant pixels, based on grouping wavelet coefficients into a tree structure called zerotree or *Spatial Orientation Trees (SOTs)*. The key observation in the zerotree coding of natural images is the following fact: if a coefficient at the node N is insignificant, then its descendants which code high frequency information at that position are also likely to be insignificant. This heuristic is based on the fact that natural images often contain areas with smoothly varying intensities or colors and sharp intensity edges. The edges then cause large wavelet coefficients in all frequency bands. Results of Shapiro were further improved by Said and Pearlman [SP96], who used a slightly different method for traversing the Spatial Orientation Trees. In this paper we use zerotree-based wavelet compression for encoding I-frames on the basis of the Said and Pearlman concept.

Motion compensation is used in video compression for reducing temporal redundancy between two or more frames. In block-based motion compensation the current frame is divided into square-shaped *macroblocks*. For every macroblock in the current frame we search for the most similar macroblock in the previous frame. The sum of the mean square errors is used as a measure of similarity.

During transmission only the difference between the best matched block in the reference frame and the compressed block in the current frame is encoded. We also have to encode the motion vectors that indicate the horizontal and vertical distances between the current macroblock and its match in the reference frame. The prediction errors between the current frame and the reference frame are encoded either with block-based methods, such as DCT compression or with the non-block-based methods, such as the wavelet transform. DCT-based coding of prediction errors usually generates block artifacts which are considered as one of its main drawbacks. Such block artifacts occur especially often at low bit rates. Therefore, we have decided to use wavelet compression for encoding prediction errors in this project.

The wavelet transform was also employed to encode prediction frames in [MSC+97] and [ON93], but these codecs do not produce scalable bitstreams. In [WG97] a rate-scalable wavelet codec is described which uses motion compensation in the wavelet domain. It turns out, however, that motion compensation works much better in the spatial domain than in the wavelet domain. Shen and Delp [SD99] describe a rate-scalable wavelet-based video codec, that is based on the so-called adaptive motion compensation. Suppose, that the target data rates of the rate-scalable codec are in the range $R_L \leq R \leq R_H$. In the adaptive motion compensation we predict the current frame, using the reference frame, decoded at the lowest possible bit rate R_L. However the results of [SD99] appear to be inferior to the H.263 compression scheme in terms of PSNR, especially for low and medium bit rates.

In [TZ94] Taubman and Zakhor describe a video compression algorithm based on 3-D subband decomposition. Their method does not use motion compensation, and therefore can be used to produce an embedded bitstream. Other methods, using 3-D subband coding approach are described in [PJF95], [CP96] and [KPB96]. However these methods are not capable to efficiently exploit the temporal redundancy. As another drawback of Taubman's and Zakhor's approach a large number of frames has to be encoded at the same time. This constraint causes high memory requirements and long delays during encoding and decoding.

As motivated in the introduction, we need rate-scalable video coding for erasure-resilient transmission. There are three types of rate-scalability for video coding: spatial scalability, SNR scalability and temporal scalability. Temporal scalability and spatial scalability allows us to decode the bit stream with different image resolutions. SNR scalability controls the decoding quality of individual frames. Temporal scalability means that the frame rate of the decoded video depends on the number of decoded parts of the

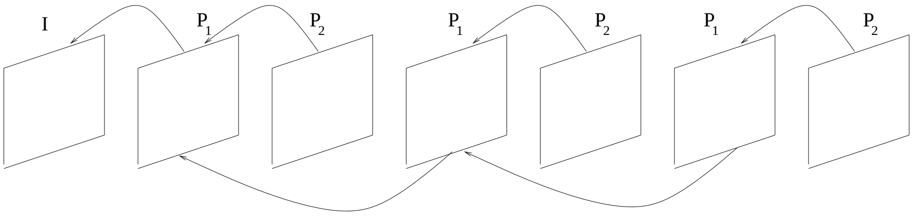

Figure 3: An example of the "prediction pyramid" with two types of P-frames

video stream.

In this paper we introduce a prediction scheme that supports temporal scalability of the produced video bitstream. This prediction scheme is based on a hierarchical rather than a conventional linear prediction structure and it is shown in Figure 3. All P-frames are divided into two classes, P_1 and P_2. Frames P_1 are predicted with help of the previous I-frame or the previous P_1-frame. P_2-frames are predicted using the previous P_1- frame. In the current version of the software only P-frames are used. However the approach, described above can be extended to compression strategies with bidirectional prediction.

2.2 Priority Encoded Transmission

Constructing a transmission system that protects data against packet losses is a well-known problem in the theory of error-correcting codes. A solution based on Reed-Solomon codes was described in [B93].

This system as well as another solution, which was described in [R89], have only one priority level and therefore are not appropriate for the transmission of multimedia data, especially in the case of multi-casting applications, when data have to be transmitted to multiple receivers. In case of multimedia applications we are interested in dividing the transmitted information into several (if possible, into a large number of) parts, adhering to the importance of the data for the application. Since loss protection is achieved at the expense of redundancy, and since we have only limited bandwidth at our disposal, it is desirable to assign different redundancy factors (and hence different loss protection) to different data layers.

The problem of transmitting messages with a prioritized information content was discussed in [S92]. The core idea is to define different channels which transmit information at different priority levels. Each decoder then receives data from as many channels as possible. The loss protection scheme, described in this paper is based on the PET method, first described in [ABE+94]. The PET system accepts as input a message of length m, packet size l and a non-decreasing priority function $\rho : \{1, \dots, m\} \to (0, 1]$. Priorities $\rho(i)$ can be specified by the user and they should be adapted to psychophysically perceived perturbations of the video stream due to packet losses. The system generates n packets of length l, such that the i-th word of the message can be decoded from any set of $n/\rho(i)$ encoding packets.

In [ABE+94] an algorithm is presented, that constructs a PET with total encoding length $\frac{\sum 1/\rho(i)}{1-d/l} + l$. They also show, that $girth_\rho = \sum 1/\rho(i)$ is a lower bound for the length of encoded message in a PET system. Furthermore, Reed-Solomon codes are used in [ABE+94] to encode different parts of the message.

However, we observe that any Maximum Distance Separable code can be implemented without changing the result of the paper in a significant way, because PET system can be used with an arbitrary MDS code (see [ABE+94]). In this paper we have used XOR-based Cauchy coding (see [BKK+95] for a detailed description of this method, that is based on Cauchy matrices).

Applications of PET for transmission of multimedia data are described in [S95] and [MRL99]. In [MRL99] PET is applied to the loss protection of wavelet-compressed images. A wavelet-based image compression method is combined with PET for loss protection. The authors use an SPIHT method of Said and Pearlman to encode the wavelet coefficients. They also present an algorithm for assigning priorities to different parts of the bit stream, generated with the SPIHT method. This algorithm is based on the assumption, that probabilities p_i are known, where p_i is the probability that i packets were lost.

An application of PET to loss protection was also described by F. Riemenschneider [R00]. He applied PET to transmission of images, compressed with the SPIHT method. We observe that the same method can be combined with any zerotree-based wavelet compression algorithm. In [R00] the bit stream is divided into blocks B_i, where B_i corresponds the i-th iteration of the zerotree algorithm (see section 2.1). We assign maximal weight w_{max} to the first four blocks. The weight $w(B_i)$ assigned to block B_i for $i > 4$ is determined by the formula $w(B_i) = 2^{-i+3} w_{max}$. Redundancy assigned to block B_i is proportional to the normalized block weight $w(B_i)/\sum w(B_i)$.

In [S95] PET was applied to transmission of MPEG video sequences. Different priorities were assigned to I- P- and B- frames. The author considers different strategies for assigning redundancies to different frames. He also assumes that probabilities $p_{n_k,n}$ of losing n_k out of n packets are known to the system.

In this paper we apply PET to loss protection of wavelet-compressed video. We divide the video stream into priority levels in a way, that exploits both temporal and spatial scalability of the video codec, used in our system. Since we allocate more bits to I-frames than to P-frames we divide the wavelet-compressed I-frames into three priority levels. The first level corresponds to the first four passes of the zerotree algorithm. The second priority level corresponds to the next three iterations. The third level corresponds to remaining iterations (i.e to eighth, nineth and so on iterations).

All P-frames in a single group of frames are assigned to the second and third priority levels. In our video compression algorithm we use the prediction scheme depicted in Figure 3. P1-frames and P2-frames are assigned to the second and third class, respectively. In order to achieve robust video transmission we increase the frequency of I-frames in the video stream. Our group of frames consists of nine frames: one I-frame, four P1-frames and four P2-frames.

In case that only P1-frames were decoded (i.e, if we are able to decode only the first and the second priority level) we repeat every P1-frame twice. If only the first priority level was decoded we decode the I-frame, using available information about the first four iterations of the zerotree algorithm. Then we replicate the decoded I-frame eight times. If, because of the high rate of packet losses, we were not able to decode even the first priority level, we replicate the last decoded frame from the previous group of frames nine times.

3 Implementation Details and Experimental Results

In our codec implementation we have used block-based motion compensation with macroblocks of size 16×16 pixels. The sum of absolute differences serves as a measure of mismatch between macroblocks. Only luminance components of macroblocks are used in mismatch measurements. The best matching macroblock is selected in a search range of 12 pixels in both horizontal and vertical directions centered around the coordinates of the encoded block in the reference frame. A bonus value of 100 is subtracted from the mismatch of the zero displacement vector which robustifies our implementation against random fluctuations in the intensities. Half-pel displaced macroblocks are also considered in the same search range in the reference frame.

Four candidate motion vectors for a macroblock can parametrize a motion model, if the mismatch value exceeds a certain maximal value. The following heuristic is implemented to decide between one and four motion vectors per macroblock: If the sum of absolute differences between the best match in the reference frame and the encoded block exceeds the threshold value of 500, we divide the 16x16 macroblock into four 8x8 blocks and perform the motion estimation for every block separately. We add the penalty value of 750 to the total mismatch value for the four 8x8 blocks. If the total sum of absolute differences of the four blocks together with the penalty value exceeds the mismatch value for the 16x16 macroblock, then four separate motion vectors are encoded. Otherwise only one motion vector for the macroblock parametrizes our motion model.

The values of the motion vectors are encoded with prediction coding and Huffman coding. For prediction three closest motion vectors for the blocks, located to the left and above of the current block are selected. For reasons of robust statistics the median of the three motion vectors are used as the prediction value and the difference between the predicted and actual values of the motion vectors are encoded. Golomb-Rice coding is employed to encode prediction errors for the motion vectors. In addition, wavelet coding is used to compress prediction errors for P-frames.

A critical component of the PET system relies on a data driven choice of the redundancy factors. We have assigned redundancy factors of 5, 3 and 2 to the first, second and third priority levels, respectively. Slight changes of these factors will only affect the transmission quality in a moderate way.

For our experiments we have selected several video sequences from the MPEG test set, e.g., we have experimented with the sequences "Coastguard", "Foreman" and "Mother and Daughter". The video sequences are represented using the YUV color space. In the table 1 results of experiments with the selected video sequences are summarized. The compression ratios of the video sequences is fixed at 24 and 48 kilobits per second and the frame rate is 10 frames per second. Due to applied loss protection algorithm the compressed video data will be expanded and we will need approximately three times higher data rates to transmit the video stream. The average PSNR (sum of PSNRs for all frames divided by the number of frames) is monitored as a distortion measure.

In figure 4 the PSNRs for different frames of the mother and daughter sequence compressed at 48 kbps are plotted. The next four pictures show the results of experiments with the same sequence, when random packet losses with probability 0.35 occurred. (i.e. we used a random number generator

File	Compression Rate (in kbps)	Loss Rate	average PSNR
coastguard	48	0	27.12
coastguard	48	50.00%	27.12
coastguard	48	75.00%	24.55
foreman	48	0	27.54
foreman	48	50.00%	27.52
foreman	48	75.00%	24.42
mother and daughter	24	0	30.56
mother and daughter	24	50%	29.59
mother and daughter	24	75%	21.29
mother and daughter	48	0	33.64
mother and daughter	48	50%	33.6
mother and daughter	48	75%	32.08

Table 1: PSNRs for standard video sequences

to mark every packet as lost with probability 0.35).

The upper graphic of Figure 5 shows the PSNRs for individual frame of the sequence. The lower graphic on Figure 5 shows percentage of packet losses for different frames. Since in our model we apply loss-resilient coding to groups of frames and we have chosen the size of a group of frames to be nine, percentage of packet losses remains constant for groups of nine consecutive frames. The upper graphic on Figure 6 shows average motion vectors for different frames of the sequence. Average motion vector value for a frame is computed, acccoriding to the formula $\sum_{i=1}^{N} av_i/N$, where N is the number of macroblocks. If one motion vector is used in a macroblock i av_i equals to the norm of the motion vector. If four motion vectros are used, av_i equals to the averag of the norms of these motion vectors.

The lower graphic on Figure 6 shows the PSNR between an encoded and decoded frame and the *previous* input frame.

In our codec loss protection is applied to a group of frames. Therefore we may expect a similar character of losses for the groups of frames with the same percentage of the lost packets. However in the experimental results certain differences in quality between different groups of frames and also between different frames from the same group can be observed. These differences are explained by the character of our loss protection method. If the data from P2 frames can not be reconstructed we replace P2-frames by previous P1-frames. The quality of such reconstruction depends on the properties of individual frames in the video sequence. In frames with vivid motion our frame replication strtegy leads to worse results than in frames with insignifacant motion. Besides that in motion intensive frames PSNRs between the orignal frames and the decoded frames, used in replication, are higher.

Significant fluctuations in PSNR values can be observed in figure 5. These fluctuations are caused by the fact that the number of packet losses exceeds a certain threshold value and only two of three priority levels can be used in the decoding. In this case P2-frames in the corresponding groups of frames are not reconstructed and the previous P1-frame replaces the lost P2-frame. While this strategy works very well for groups of frames with moderate motion content it leads to reduced PSNR values for other frames with vivid motion. To support this hypothesis we show PSNRs of frames from the same mother and daughter sequence, compared to previous decoded frames in this sequence in figure 6 Groups of frames with higher average motion vectors and with lower PSNRs are characterised by the higher fluctuation rates. We like to emphasize, however, that our approach yields good results both in terms of average PSNR and in terms of subjective video quality.

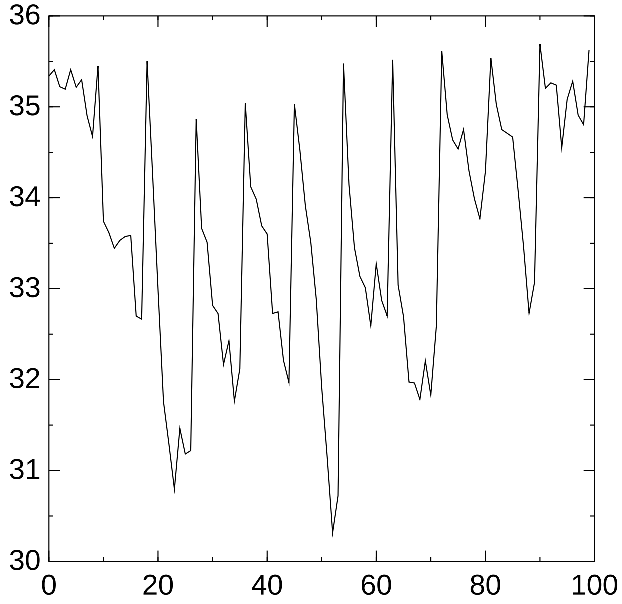

Figure 4: PSNR for mother and daughter sequence without packet losses, compressed at 48 kbps

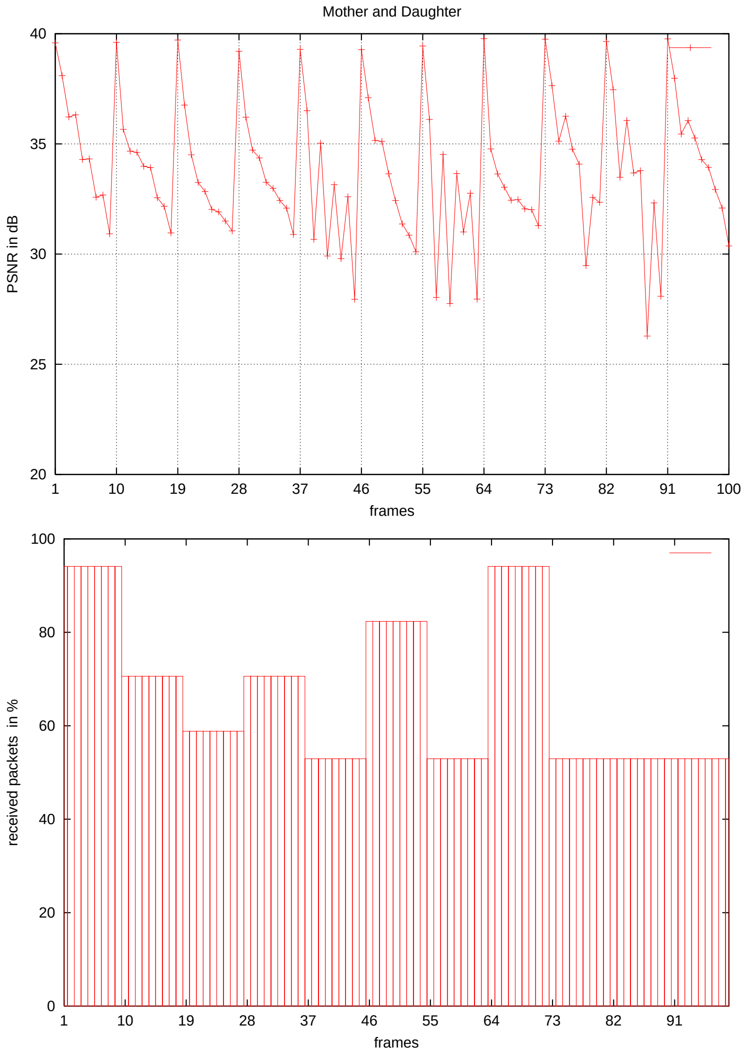

Figure 5: PSNR for mother and daughter sequence with packet loss rate 35%, compressed at 48 kbps. Percentage of packet losses for mother and daughter sequence

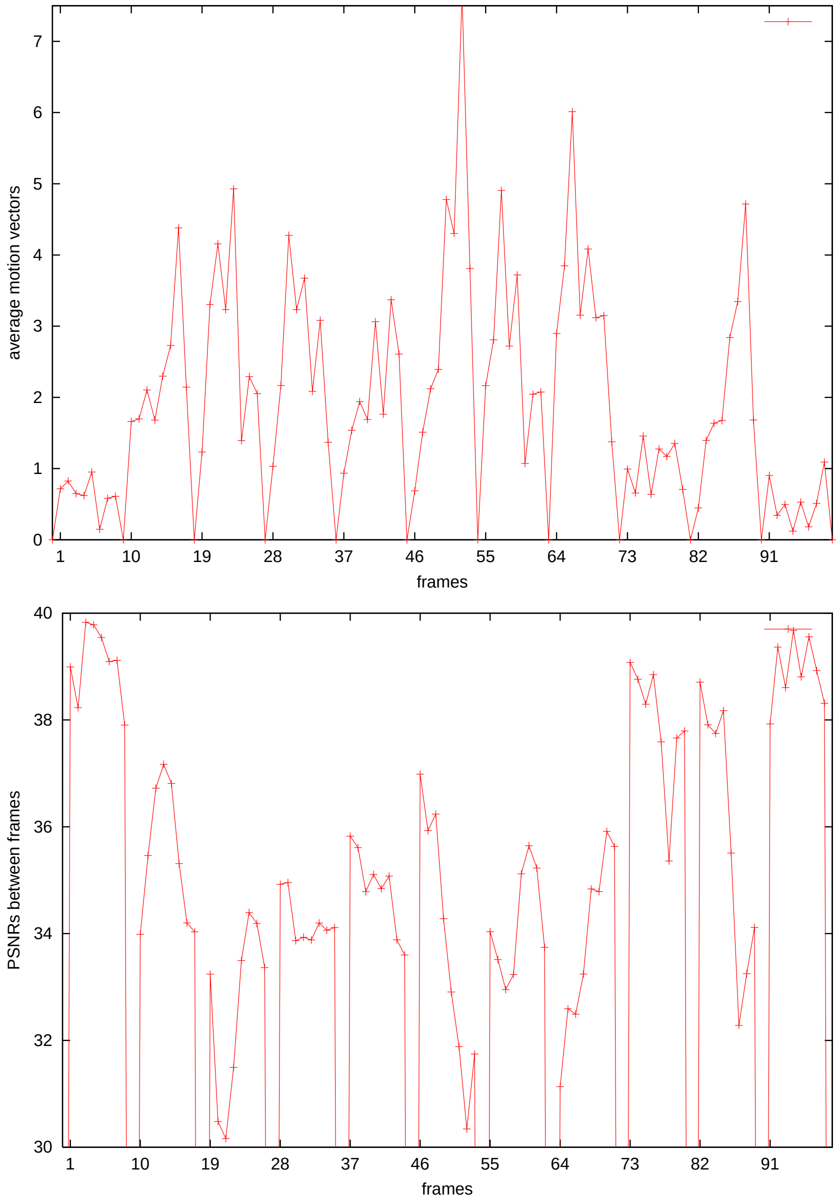

Figure 6: Average motion vectors. PSNRs between the decoded frame and the previous input frame

4 Summary and Further Research

It was shown in this paper that we can construct a rate-scalable codec and achieve graceful degradation in case of image transmission (see, for instance [MRL99]). However, smooth dependency of video data quality on the number of lost packets is still a partially unsolved problem.

This difference between image and video transmission can be explained by differences in compression methods. Several transform-based coding methods, designed for compression of still images such as wavelet compression methods, can also be used to produce a scalable bitstream. Efficient video compression methods, however, must additionally exploit temporal redundancy, i.e. the dependencies between individual frames. To avoid repeated coding of similar pieces of data in different frames prediction-based coding provides a solution. In spite of its efficiency this class of coding methods is not appropriate for scalable compression.

The video system presented in this work addresses this problem by modifying the order in which frame prediction is performed. This system supports efficient and rate-scalable video compression. Combined with the PET algorithm our system results in efficient video transmission at moderate degration in image quality. The novelty of this system is an application of efficient encoding algorithm for the loss protection with rate-scalable video compression algorithm.

In the video codec, presented in this paper, only P-frames have been used for prediction. It would be interesting to use bi-directional prediction for encoding P2-frames, since it promises to yield better compression. Another improvement can be expected from DCT-based compression to encode prediction errors in P-frames. This choice could lead to a more flexible rate-scalability of the video codec and to the higher number of priority levels in our encoding scheme.

5 Acknowledgments

In this project we have extensively used the source code of the video codec, designed by H. J. Klock, A. Polzer and J. M. Buhmann in the BMBF research project Electronic Eye (01 M 3021 A/4).

References

[ABE+94] Andres Albanese, Johannes Blömer, Jeff Edmonds, Michael Luby, Madhu Sudan, Priority Encoding Transmission, IEEE Symposium on Foundations of Computer Science, 1994, pp. 604-612

[BKK+95] Johannes Blömer, Malik Kalfane, Richard Karp, Marek Karpinski, Michael Luby, David Zuckerman, *An XOR-based Erasure-resilient Coding Scheme*, Technical report, International Computer Science Institute, Berkeley, California, 1995

[B99] J. M. Boyce, Packet Loss Resilient Transmission of MPEG Video over the Internet, Signal Processing: Image communications, vol. 15, 1999, pp. 7-24

[CP96] Chen, Y, Pearlman, W., *Three Dimensional Subband Coding of Video Using the Zero-tree Method* Proc. of the SPIE Conference on Visual Communications and Image Processing, 1996, San Jose, California. pp. 1302-1312

[KPB96] H. J. Klock, A. Polzer, J. M. Buhmann Region-based Motion Compensated 3D-Wavelet Transform Coding of Video, Proc. of ICIP, Santa Barbara, 1997 pp.776-779

[GLM+99] J. Goshi, R. E. Ladner, A. E. Mohr, Eve A. Riskin, and A. Lippman Unequal Loss Protection for H.263 Compressed Video. Proc. of the 2003 Data Compression Conference, pp.73-82

[MSC+97] Martucci, S.A., Sodagar, I., Chiang, T., Zhang, Y.-Q., *A Zerotree Wavelet Video Coder* IEEE Transactions on Circuits and Systems for Video Technology, vol. 7, 1997, pp 109-118

[MRL99] Alexander E. Mohr, Eve A. Riskin, Richard E. Ladner, Graceful Degradation over Packet Erasure Channels through Forward Error Correction Proc. of the 1999 Data Compression Conference, 1999, pp. 92-101

[B93] E.W. Biersack Performance Evaluation of Forward Error Correction in an ATM environment Journal of Selected Areas in Communication, vol.11, 1993, pp. 631-640

[ON93] Ohta, M., Nogaki, S., *Hybrid Picture Coding with Wavelet Transform and Overlapped Motion-compensated Interframe Prediction Coding* IEEE Transactions on Signal Processing, vol. 41, 1993, pp.3416-3424

[PJF95] Podilchuck, C.I., Jayant, N.S., Faravardin, N., *Three Dimensional Subband Coding of Video* IEEE Transactions on Image Processing, vol. 4, 1995, pp.125-138.

[R89] Rabin, M.O. *Efficient Dispersal of Information for Security, Load Balancing and Fault Tolerance* Journal of the ACM, vol. 36, No. 2, 1989, pp. 335-348.

[R00] F. Riemneschneider, Ein on-line System für robuste Videodaten Übertragung, Diplomarbeit, Institut für Informatik, Universität Bonn, 2000

[SP96] Amir Said and William A. Pearlman, *A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees* IEEE Transactions on Circuits and Systems for Video Technology, vol. 6, 1996, pp. 243-250

[S92] Schacham, N., *Multicast Routing of Hierarchical Data* Proc. of ICC'92,pp. 1217-1221, vol.3 Chicago, 1992.

[S93] J. Shapiro, *Embedded Image Coding Using Zerotrees of Wavelet Coefficients* IEEE Transactions on Signal Processing, 41(12), 1993, pp. 3445-3462.

[SD99] Shen, K., Delp, E.J. *Wavelet Based Rate-scalable Video Compression* IEEE Transactions on Circuits and Systems for Video Technology, Vol. 9, No. 1, pp. 109-122, 1999.

[S95] Storn, R., *Modeling and Optimization of PET-Redundancy Assignment for MPEG Sequences* , Technical Report TR-95-018, ICSI, May 1995.

[TZ94] Taubman, D., Zakhor, A., *Multirate subband 3-D Coding of Video* IEEE Transactions on Image Processing, vol. 3, 1994, pp. 572-588.

[WG97] Wang Q., Ghanbari, M., *Scalble Coding of Very High Resolution Video Using the Virtual Zerotree* IEEE Transactions on Circuits and Systems for Video Technology, vol. 7, 1997, pp. 719-727