PVR System Design of Advanced Video Navigation Reinforced with Audible Sound

Onno Eerenberg, Ronald M. Aarts and Peter H. N. de With, *Fellow*, IEEE

Abstract — *This paper presents an advanced video navigation concept for Personal Video Recording (PVR), based on jointly using the primary image and a Picture-in-Picture (PiP) image, featuring combined rendering of normal-play video fragments with audio and fast-search video. The hindering loss of audio during conventional fast-search trick play is eliminated, by adding the sound associated to the normal-play video fragments. The normal-play fragments provide detailed information, whereas the fast-search signal simultaneously presents a coarse overview, where it will be shown that the audio enhances the overall navigation efficiency. This system setup requires a specific DTV decoding process, combining multiple, independent audiovisual information signals. Experiments revealed that it is possible to decode all signals by efficient signal processing, thereby re-using the standard DTV decoding platform for decoding, both the normal-play audiovisual fragments and associated fast-search information signal. By applying a decoding concept for the fast-search navigation signal that allows exchanging the navigation refresh rate against execution cycles, it is ensured that real-time performance is obtained on an embedded CPU, as typically deployed in DVT platforms for PVR*[1].

Index Terms — Audio, Fast search, H.264/MPEG4-AVC, MPEG-2, Navigation, Picture in Picture, PVR, Scalable decoding, Trick play.

I. INTRODUCTION

The availability of high-capacity hard-disc drives has enabled the development of Personal Video Recording (PVR), which are consumer-based storage products for audiovisual information. These systems are equipped with efficient video navigation features such as key frame extraction or automatic video summarization, which are either not possible in a cost-effective manner with traditional tape-based storage solutions or features, which are far more advanced compared to their traditional counterpart.

A basic PVR implements conventional fast-search and slow-motion trick play and supports time-shift recording [1]. A more advanced PVR has additional features, such as automated video editing [2], video transcoding [3] and advanced methods for video navigation [18]. Video navigation has been subject of research for many years, resulting in methods such as text-based browsing [4], key-frame extraction [5], or program summarization [6], [7], which may be rendered and visualized in an attractive way [8]-[10].

The usage of audio information for trick play has been limited to either playback speeds around unity [11], or for feature extraction, enabling advanced intra-program video navigation [12]. A direction in advanced intra-program video navigation is the simultaneous rendering of multiple information signals. As human perception employs both visual and auditory cues, it is opportune to employ audio information associated with the video navigation information, thereby creating a multi-source information signal for advanced intra-program video navigation. However, cognitive processing of sound occurs at multiple time-scales. In auditory perception, these time-scales range from microseconds for sound-source localization, via milliseconds for pitch analysis and event detection, to tens of milliseconds to analyze speech characteristics. In speech perception, phones are recognized requiring around 100 milliseconds, while syllables and speech require larger time-scales [13]. As a result of this, it is not effective to select audio information associated with a single picture forming the video navigation signal, which generates 40/33 milliseconds of consecutive sound for a 25/30-Hz television system. At present, conventional video navigation methods do not employ audio signals, requiring continuous attention of the viewer during video navigation. The objective of this work is to facilitate the user with a supplementary audio signal for navigation, so that a user can perform another task in parallel with video navigation.

This paper presents a novel audio-enhanced double-window video-navigation technique, based on a conventional primary image with associated audio, combined with an additional Picture-in-Picture (PiP) screen, as depicted in Fig. 1. Hereby, the main window displays fragments of normal-play video

Fig. 1. Video that contains a primary window showing normal-play fragments with associated audio, while the secondary PiP window displays fast-search video.

[1] Onno Eerenberg is with Trident Microsystems, Laan van Diepenvoorde 23, 5582 LA Aalst/Waalre, The Netherlands. (e-mail: onno.eerenberg@tridentmicro.com).

Ronald M. Aarts is with Philips Research Prof. Holstlaan 4 5656 AA Eindhoven, The Netherlands (e-mail: ronald.aarts@ philips.com).

Peter H. N. de With is with Eindhoven University of Technology, Den Dolech 2, 5600 MB, The Netherlands (e-mail: p.h.n.de.with@tue.nl).

Contributed Paper
Original manuscript received 09/03/14
Revised manuscript received 11/25/14
Current version published 01/09/15
Electronic version published 01/09/15.

while conventional trick-mode playback, such as fast forward or fast reverse video, is presented in the PiP window. The normal-play video fragments are equipped with the corresponding audio information, to strengthen the content traceability, thereby making the scene more informative for navigation purposes, up to selecting the video information solely based on the associated audio information. For the visual part, the primary window presents normal-play video fragments providing detailed information to the viewer, whereas the PiP window presents fast-search video offering an outline of the normal-play video sequence.

In order to support the full double-window audio-enhanced video-navigation solution not only for high-end PVR platforms, but also enable mapping on a low-cost PVR platform, this work explores the scalable implementation of a PiP-based video-navigation signal. This scalability is required to fit the implementation to the considered heterogeneous low-cost PVR platform. To this end, a technique is proposed to develop and evaluate a computation-reduced H.264/MPEG4-AVC intraframe decoder for deriving the PiP pictures, while at the same time resolving a quality pixel-drift problem. It will be shown that the implementation can even be executed on a standard DTV computing platform.

The performance scalability is enabled by (1) reduction of the trick-play refresh rate, and (2) by exploiting algorithmic simplifications in the spatial block-based predictor signal computation, as used in modern compression schemes for digital video entertainment. The above description on the outlines of the implementation of the advanced video navigation already clarifies that a number of requirements have to be satisfied, in order to embed this navigation system into an existing PVR. First, the algorithm should enhance the navigation experience. Second, the system complexity should be low enough to seamlessly integrate it into a typical PVR Consumer Electronics (CE) platform. Third, as CE platforms are cost-constrained, the objective is to re-use existing Audio/Video (AV) decoders for the decoding of navigation information signals. Moreover, this also implies that a solution should exploit the predetermined hardware/software architecture in an efficient manner for navigation.

The paper is divided as follows. Section II introduces the concept of double-window audio-enhanced video navigation and presents the PVR architecture based on the proposed video navigation method. Section III introduces system aspects and requirements for realization of the double-window audio-enhanced trick play. Section IV elaborates on a scalable H.264/MPEG4-AVC computational-reduced intraframe video decoding method. Section V introduces the proposed partial prediction block reconstruction to provide scalable decoding and avoid pixel-drift problems. Section VI discusses experimental results on the above system aspects. Finally, conclusions are presented in Section VII.

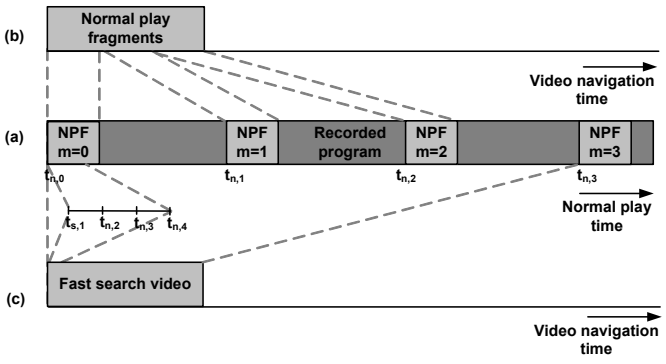

Fig. 2. Double-window based video navigation. (a) Recorded MPEG-compressed program. (b) Selected Normal-Play Fragments (NPF). (c) Fast search video navigation signal.

II. PVR Architecture and Video Navigation

Basic PVR functionality includes simultaneous record and playback, also known as time-shift recording and conventional trick-play search functionality, while an advanced PVR offers e.g. more novel video navigation methods as described in the previous section, which will co-exist next to conventional navigation methods. Low-cost conventional trick play [14] typically involves location knowledge, revealing the location of intraframe-coded (I-type) pictures, which enables re-use of specific normal-play compressed pictures to construct the video navigation signal. This location information and other information, such as scene-change locations, are frequently indicated as Characteristic Point Information (CPI), which is separately stored as metadata. Audio-enhanced double-window based video navigation combines conventional video navigation and normal-play playback, enabling potential re-use of the functional blocks of an existing PVR architecture.

A. Audio-enhanced double-window video navigation

This part of the system involves a novel form of intra-program video navigation, featuring a double-window based video navigation enhanced with audio signals. This concept requires the decoding of multiple information signals, as indicated in Fig 2. The first signal, depicted in Fig. 2 (b), constructs the primary video window on the basis of normal-play fragments, featuring also the corresponding audio information. The second information signal constructs the PiP-based fast-search trick-play video navigation window, of which the time sampling is depicted in Fig. 2 (c). Hereby, $t_{s,k}$ indicates the time location relative to time $t_{n,0}$, the start point of navigation. Time sampling here refers to a temporal picture selection process, in which the normal-play pictures are selected that coincide with the time moments for fast-search video navigation, according to the relation

$$t_{s,k} = \frac{kP_s}{f}. \qquad (1)$$

Parameter P_s is the relative playback speed for the fast-search video navigation signal, $k = \{0, 1, 2, ..., N\}$ a picture index revealing the re-used normal-play pictures constructing the trick-play signal, N the last picture index depending on the selected playback speed P_s and f the television frame rate. The corresponding normal-play fragments are characterized by T_{np}, indicating the normal-play fragment duration time and $t_{n,m}$ the m-th normal-play fragment start position relative to $t_{n,0}$. Hereby, with P_s being the relative playback speed and $m = \{0, 1, 2, ..., M\}$ a normal-play fragment index revealing the re-

used normal play fragments and M the last fragment index depending on the selected playback speed P_s according to

$$t_{n,m} = mT_{np}P_s . \quad (2)$$

When considering a 25-Hz television frame rate and the relative playback speed P_S equal to 12, which is motivated based on a practical value for an intraframe refresh rate of 2 Hz, typical successive values for $t_{s,k}$ are integer multiples of 0.48 seconds, while typical successive values for $t_{n,m}$ are integer multiples of 36 seconds when T_{np} equals 3 seconds.

B. PVR system block-diagram

Figure 3 depicts a basic PVR system block-diagram [14, 21], enabling conventional and advanced PVR features. This diagram includes time-shift recording and two video navigation modes: a conventional search mode and an audio-enhanced mode. When the switch at the bottom of Fig. 3 is set to (c), this resembles conventional fast-search trick play, while when the switch is set to (c), the conventional navigation is integrated into an audio-enhanced double-window video navigation, in combination with signal (d). Note that when the

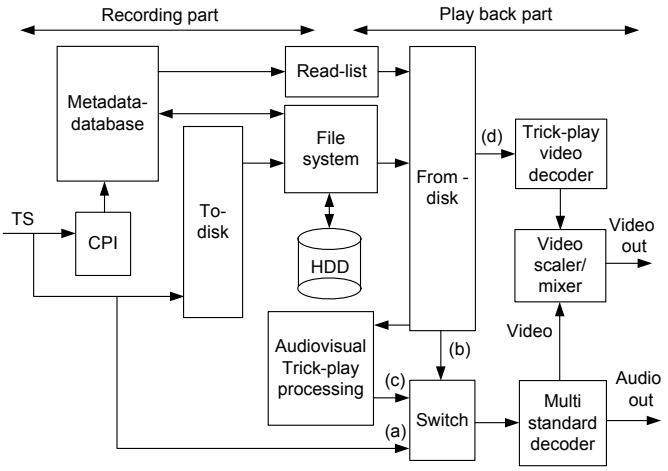

Fig. 3. Main PVR functional blocks enabling double-window video navigation. (a) Switch set to view real-time. (b) Switch set to time-shift recording. (c) Switch set to conventional trick-play or contains video for main window in case of double-window based trick play. (d) Input for PiP window in case of double-window based trick play.

switch is set to (a), this resembles normal viewing condition, whereas switch set to (b) this resembles time-shift recording.

Starting at the left-hand side of Fig. 3, an MPEG-2 Transport Stream (TS) enters the system. During TS recording, CPI is derived, which amongst others allows tracking of the intra-encoded pictures of the recorded program and optionally determines the scene-change locations. This information and related parameters are stored in the metadata database. The metadata database is stored and retrieved from the storage medium via its own control. The audiovisual information stream is written or retrieved from the storage medium via the 'to-disk' block and 'from-disk' block. The 'to-disk' block involves write buffering and the control of the writing process. Similarly, the 'from-disk' block contains the TS reading and the involved output buffering and control. For the various playback situations, the 'read-list' block determines, via the 'metadata database', the media access-points of the individual information streams and controls the 'from-disk' block. This block retrieves the normal-play TS or in case of trick play, the normal-play TS fragments for trick play. The TS fragments for trick play contain intraframe-encoded pictures that are employed for constructing the fast-search video trick play signal. The audiovisual trick-play processing block in Fig. 3, performs an MPEG-2 de-multiplexing operation to access the individually compressed audiovisual access units. The video processing, operating in the compressed domain, replaces pictures with repetition pictures to avoid decoding artifacts at fragment boundaries, due to the absence of required anchor pictures. This AV trick-play processing block also performs selection of the compressed normal-play audio frames, which are associated with the video fragments, thereby avoiding potential bit-rate violations. After processing the audiovisual access units, the individual packetized elementary streams are multiplexed into an MPEG-2 TS, which is supplied to the multi-standard audiovisual decoder. For double-window video navigation, a second video decoder processes the fast-search trick-play signal, which is then scaled by the video scaler/mixer (at the right) and combined with the normal-play video fragments.

III. System Aspects of Audio-Enhanced Double-Window Trick Play

This section deals with the system and processing aspects for trick play, deploying double-window audio-enhanced video navigation, which is supported by the architecture block diagram from Fig. 3.

A. Video navigation aspects for audio perception

The double-window audio-enhanced video navigation imposes the following navigation aspects.

- **Normal-play video fragment duration.** The normal-play fragment duration is mainly determined by the perception of the sound signal, which differs for various content (music, speech, etc.).
- **Scene-change detection.** Scene-change information is beneficial in order to adjust the start of a normal-play fragment, thereby optimizing the selection of a sufficiently long meaningful audio fragment, of the same time interval.

B. Audiovisual signal processing aspects

The double-window audio-enhanced navigation method requires independent dual-channel video decoding and additional audiovisual processing, which must be conducted cost-effectively to support low-cost consumer platforms.

- **PiP video decoding.** The PiP-based fast-search video navigation signal can be obtained by means of hardware or software decoding. The former is possible for high-end consumer platforms, which are equipped with dual-channel video decoding, while the latter solution is typically required for low-cost consumer platforms. A PiP-based video navigation signal, which has a reduced spatial resolution compared to the original video, benefits from scalable video decoding, which is more cost-

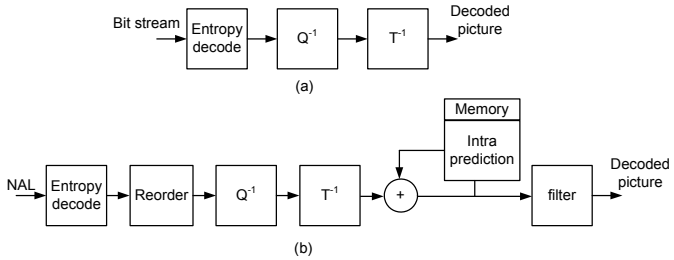

Fig. 4. Basic block diagram for intraframe decoder. (a) MPEG-2. (b) H.264/MPEG4-AVC.

effective regarding system resources, such as cycle consumption, bandwidth and memory footprint.

- **Trick-play refresh rate.** The trick-play refresh rate is typically equal to the rendering rate. For trick play with high speed-up factors, it is advantageous to reduce the refresh rate to allow an improved interpretation by the viewer. A reduced refresh rate also lowers the computational load and decreases bandwidth, which is beneficial when decoding the fast-search video signal on a control processor.
- **Video fragment processing.** The normal-play video fragments are constructed using multiple Group-Of-Pictures (GOP). At the normal-play fragment boundaries, decoding artifacts may be visible depending on the absence of reference pictures. To avoid visible artifacts, the first and last GOP constructing a normal-play fragment (typically open GOPs) requires modification (closing the open GOP structure) for avoiding decoding artifacts.
- **Audio fragment processing.** Depending on the audio compression standard, padding samples may occur in an audio frame. Concatenation of non-sequential audio frames having padding samples causes an audio buffer violation, which can be prevented by applying proper signal processing in the compressed audio domain.

In order to support the full double-window audio-enhanced video-navigation solution not only for high-end PVR platforms, but also enable mapping on a low-cost PVR platform, a scalable decoding method is proposed for deriving a PiP-based video-navigation signal. This method exploits computational scalability, which is further elaborated in Section IV. In the same section, a quality pixel-drift problem is addressed that arises when deploying pure spatial scalability. In order to circumvent this drift, an algorithm is proposed that utilizes a partial spatial block reconstruction for decoding.

IV. Complexity Scalable Video Decoding

Software-based decoding of an intraframe-compressed trick-play signal may exceed the available system resources such as memory footprint, cycle budget and bandwidth. In general, the concept of complexity-scalable video decoding may form an attractive solution to control decoding complexity and thus the required system resources [20]. In order to reduce the spatial resolution, the scalable decoding technique operates in the transform domain, where a sub-set of

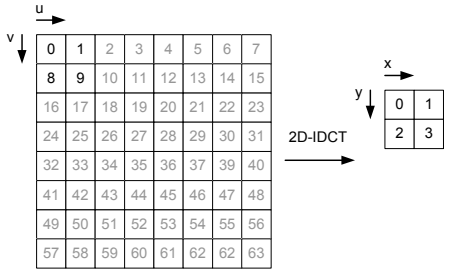

Fig. 5. Scalable 2×2 IDCT for an SD-to-QCIF MPEG-2 decoder. At the left, the received 8×8 coefficients (0..63) are depicted. At the right, a 2×2 pixel block is depicted obtained by applying a 2×2 scalable IDCT.

the transform coefficients are used to reconstruct a set of pixels, forming a reduced block size, compared to the original encoded block size. Typical spatial reduction factors for an 8×8 coefficient block size, which preserve the aspect ratio, are horizontally and vertically a factor of 8, 4 or 2.

A fast-search navigation signal with a PiP dimensions on the display, represents a video signal with inherently reduced spatial resolution, originating from intraframe-compressed pictures. Therefore, this video PiP signal is suitable to be obtained via computational reduced decoding techniques. A distinction is made between MPEG-2 and H.264/MPEG4-AVC compressed video sequences. Figure 4 indicates the basic block diagrams for an MPEG-2 and H.264/MPEG4-AVC intraframe video decoder. The main difference between the two intraframe coding schemes is that for MPEG-2 after the inverse transformation, block-based pixel data is available, whereas for H.264/MPEG4-AVC the result after inverse transformation is a block-based residue signal, which requires addition to a block-based predictor, in order to reconstruct the

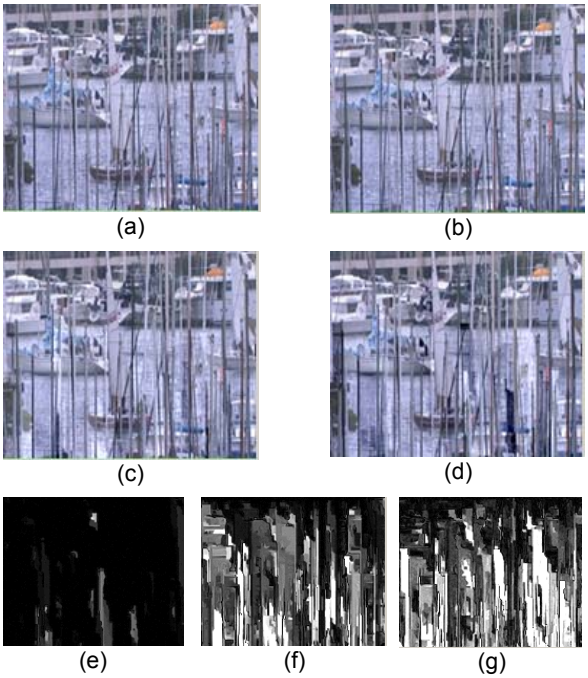

Fig. 6. Comparison of transform-domain reconstructed Harbor pictures of 176x144 pixels and factor 8 amplified corresponding error signal. (a) Original picture. (b) Picture coded at QP=6. (c) Picture coded at QP=18. (d) Picture coded at QP=28. (e) Error for QP=6. (f) Error for QP=18. (g) Error for QP=28.

TABLE I
COMPUTATIONAL REDUCTIONS FOR VARIOUS INTRAFRAME 4X4 AND 8X8 BLOCK-BASED PREDICTOR CALCULATIONS.

Prediction mode	Calculation reduction for 4x4 block size (%)			Calculation reduction for 8x8 block size (%)		
	Add	Mult	Div	Add	Mult	Div
Diagonal down left	45	42	42	46	46	46
Vertical right	27	16	30	30	21	31
Vertical left	36	20	40	42	27	45
Horizontal up	40	50	50	67	42	71
Horizontal down	24	16	30	30	21	31

TABLE II
COMPUTATIONAL REDUCTIONS FOR INTRAFRAME 16X16 PLANE-BASED PREDICTOR CALCULATIONS.

Prediction mode	Calculation reduction (%)			
	Add	Mult	Sub	Div
Plane prediction	75	75	75	75

final pixel data. A complexity-scalable compression technique for MPEG-2 is described in [20], while for scalable decoding a solution is presented in [17] and for H.264/MPEG4-AVC in [15][16].

A. MPEG-2 scalable intraframe decoder

In MPEG-2, video decorrelation is achieved by means of an 8×8 2D-DCT, which enables a scalable decoder to reconstruct a spatial region that has fewer pixels. Figure 5 indicates such a form of spatial scalability, where a selection of 4 coefficients (upper-left corner) from an 8×8 DCT matrix results in spatial region of size 2×2, effectively reducing the original 8×8 region by a factor 4 in horizontal as well as vertical direction.

B. H.264/MPEG4-AVC scalable intraframe decoder

H.264/MPEG4-AVC intraframe compression differs from preceding video coding methods such as MPEG-1/2, due to the presence of spatial prediction prior to transform coding. Intraframe H.264/MPEG4-AVC decoding requires, next to calculating the residual information, a block-based prediction signal, which is calculated using integer arithmetic, to reconstruct the final pixel values. Chen [15] has indicated that the prediction signal can be derived in the frequency domain, enabling scalable decoding in the transform domain. Kim [16] has used this technique to derive thumbnail-sized images from H.264/MPEG4-AVC intraframe-compressed pictures. Although the results in [16] work for H.264/MPEG4-AVC up to main profile, the solution has two weaknesses. The first disadvantage is sample-value drift, see Fig. 6. For thumbnail-sized pictures, this distortion may sometimes be perceptually acceptable, but for a PiP window of size 480×270 pixels this distortion is perceived as annoying and therefore needs to be avoided, see Fig. 7. The second disadvantage of the frequency-domain decoding method of [15] does not provide a solution for the decoding an 8×8 block size, which is deployed

Fig. 7. Drift effect on picture quality for a PiP picture with size 480x270 pixels due to non-exact predictor calculation.

for profiles covered by the Fidelity Range Extension (FRExt) and utilizes low-pass filtering of the reference pixels prior to constructing the block-based prediction. From experiments using scalable decoding techniques as described in [15][16], it becomes clear that a computationally reduced H.264/MPEG4-AVC intraframe decoder must preserve the reference pixels, such that the decoder output is not contaminated by pixel-value drift.

Reference pixels utilized by H.264/MPEG4-AVC spatial prediction are located at the bottom row and at the outer-right column of the 4×4 or 8×8 block. On the basis of the reference pixel locations for spatial prediction and the intraframe H.264/MPEG4-AVC decoder depicted in Fig. 4, it is evident that the transform block and spatial prediction block can benefit from partial calculation. Using this method, pixels not involved by the spatial prediction are omitted from calculation. Furthermore, omitting the final spatial low-pass filtering for de-blocking, which is outside the intraframe decoding loop, further reduces the computational load.
This allows for two computational reduced decoding approaches to calculate the PiP-based fast-search trick-play signal, which are elaborated in the next section.

V. PARTIAL PREDICTION BLOCK RECONSTRUCTION

Partial calculation of a block-based prediction signal allows for two scenarios. The first scenario utilizes the reduction offered by calculating a partial spatial block-based predictor and bypassing the final low-pass filter block, while the second solution additionally employs the computation reduction offered by a partial decoder inverse transformation of the residual signal.

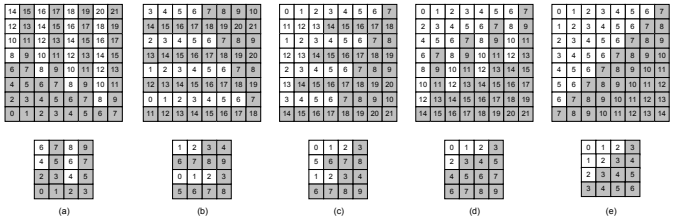

Fig. 8. Block-based predictor for 4×4 and 8×8 pixel prediction modes, utilizing partial calculation. Gray pixels are correctly calculated, whereas white pixels are duplicates of the nearest correct neighbor. (a) Horizontal-down predict. (b) Vertical-right predict. (c) Vertical-left predict. (d) Horizontal-up predict. (e) Diagonal-down-left predict.

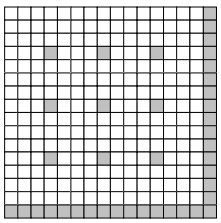

Fig. 9. Reconstructed 16×16 plane-based prediction for reduced calculation in intraframe decoding. Gray pixels are correctly calculated.

A. Scenario 1: Algorithm for partial calculation of a block-based spatial predictor

Up to main profile, H.264/MPEG4-AVC intraframe compression deploys spatial prediction on the basis of a 4×4 or 16×16 pixel block size. For profiles based on Fidelity Range Extensions (FRExt), an additional 8×8 block size is employed. Similar as to the 4×4 block size, with the 8×8 block size a spatial prediction is conducted prior to the 8×8 transformation. Unlike the calculation of a 4×4 predictor, the pixels involved for calculating an 8×8 predictor are low-pass filtered prior to the calculation of the block-based predictor. Both 4×4 and 8×8 block sizes use 9 spatial prediction techniques, while the 16×16 block size uses 4 spatial prediction techniques. Depending on the spatial prediction mode, arithmetic operations are employed to calculate the final block-based prediction signal, using neighboring reference pixels. Furthermore, potential spatial-prediction pixels, located at the bottom row and outer-right column of a 4×4 or 8×8 pixel block may also be used inside such a predictor block, see for examples Fig. 8 and Fig. 9.

From the set of spatial-prediction modes, the vertical and horizontal prediction modes duplicate reference pixels, involving no additional arithmetic operations. The DC-based prediction mode calculates the average of the available reference pixels, while the diagonal-down-right prediction mode yields no computational relief. As a result, only 5 modes benefit from computational reduction as depicted in Fig. 8. Figure 8 indicates the gray prediction pixels that are correctly calculated forming a 4×4 or 8×8 block-based spatial predictor,

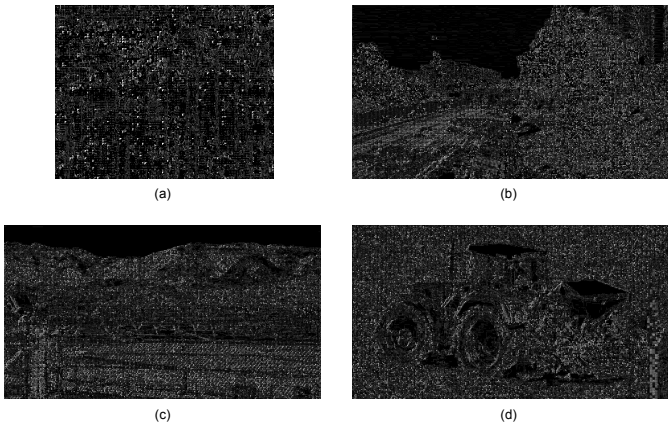

Fig. 10. Factor 32 amplified decoding error between standard H.264/MPEG4-AVC decoding and H.264/MPEG4-AVC intra-frame decoding based on partial calculation of the spatial predictor. (a) Harbor. (b) Freeway. (c) Airplane. (e) Tractor.

TABLE III
NUMBER OF OPERATIONS (# OPS) REQUIRED FOR REGULAR AND PARTIAL CALCULATION OF A 4×4 AND 8×8 BLOCK IN THE H.264/AVC INVERSE DCT TRANSFORMATION.

Transform size	Operation	# Ops per block normal decoding	# Ops per block partial calc. decoding
4x4	+	64	49
	>>	16	13
8x8	+	512	365
	>>	160	118

TABLE IV
OBJECTIVE QUALITY OF COMPUTATIONAL REDUCED DECODED PICTURES AND DECIMATION-FILTERED PIP PICTURES.

Video sequence	Partially decoded trick play picture PSNR Luminance (dB)	PiP PSNR Luminance (dB)
Harbour	34	41
Plane	38	43
Freeway	35	40
Tractor	36	41

while deploying the computational reduction. The positions in white obtain a predictor value based on the nearest gray-pixel location. Note that the index at the various pixel locations in Fig. 8 corresponds to the pixel index, as deployed in the H.264/MPEG4-AVC software model JM18.2 to calculate the predictor. Table I indicates the computational reduction for the 4×4 and 8×8 spatial prediction modes. Although the objective is to calculate only the prediction pixels at the bottom row and outer-right block boundaries, also a substantial amount of non-boundary pixels are still correctly calculated, which limits the deformation of the block-based predictor.

Besides a 4×4 and 8×8 block-based predictor, H.264/MPEG4-AVC also employs 16×16 block-based prediction. From the four 16×16 prediction modes, only the plane prediction mode benefits from the reduced calculation approach. For the 16×16 block-based plane predictor, Fig. 9 depicts the gray-pixel locations that are correctly calculated. The white-pixel positions obtain a predictor value based on the nearest gray-pixel location. Table II indicates the computational reduction gain for the 16×16-based plane predictor. Figure 9 reveals that not only the predictor pixels are calculated that are required to avoid drift, but also predictor samples that are located at a sub-sample grid of 4×4 pixels. In this way, the PiP quality is optimized, when derived on the basis of a factor four horizontal and vertical down-sampled grid. Figure 10 indicates the difference, amplified with a factor 32, between a normal intraframe decoded picture including a de-blocking filter and a picture based on intraframe decoding employing a partial spatial predictor-block reconstruction without de-blocking filter. The strong deviating pixel values are caused by the incorrect predictors, which are situated at the white locations in Fig. 8 and Fig. 9. The amount of errors, i.e. the amount of non-black pixels, reflects a measure for the utilization of the reduced calculation approach. From Fig. 10, it is concluded that spatial prediction modes which allow partial computation are frequently employed in natural video during standard intraframe

decoding operation. Note that due to the absence of a de-blocking filter, also small deviations in the black regions occur.

B. Scenario 2: Algorithm for partial inverse transformation of the residual block

When partially calculating the spatial prediction-block, certain pixels cannot be properly reconstructed. The computational reduction can be further reduced, by employing a partial inverse transformation, restricting the computation to only those pixels that are located at the block bottom-row and outer-right column. Hereby, the computational load for the fast IDCT is reduced, for both the 4×4 and 8×8 inverse DCT transformations, compared to a full transformation [19], see Table III. The computational load reduction is always obtained, as either a 4×4 or 8×8 inverse transformation is employed in H.264/MPEG4-AVC intraframe video compression. The 2D-IDCT deployed in H.264/MPEG4-AVC can be written as $X = A^T Y A$, where matrix Y contains the transform coefficients, A denotes the IDCT transform matrix and X contains the output result of the inverse 2D transform. From the notation $X = A^T Y A$, it is evident that the 2D-IDCT is calculated in two separable stages. Computational reduction is obtained when in the second stage, the transform is only calculated for matrix elements located at potential spatial-prediction candidate locations, i.e. bottom-row and outer-right column. This partial inverse DCT transformation is used for decoding the residual signal that is added to the prediction signal.

VI. Experimental Results

Three aspects of the experimental system are now discussed. The first aspects addresses the final picture quality based on partial block reconstruction for the two proposed scenarios. Second, scalable decoding performance is provided involving the algorithmic simplifications leading to the computation reduction. The third aspect covers the subjective evaluation of the proposed dual-window based video navigation technique involving a test panel concentrating on various system user aspects and the audio enhancement.

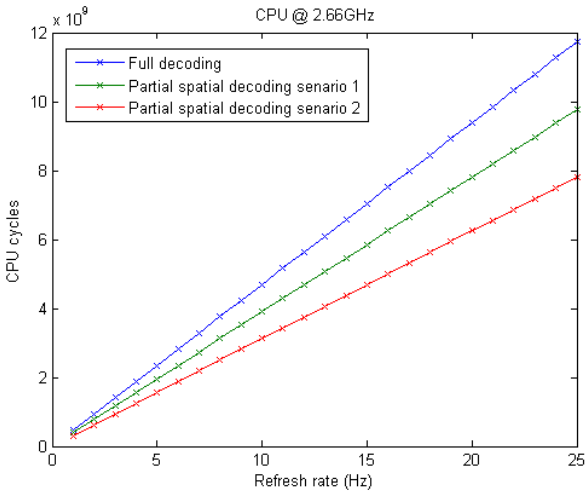

Fig. 11. Picture refresh-rate versus computation load.

TABLE V
REDUCED CALCULATION-BASED PIP WITHOUT DECIMATION-FILERING.

Video sequence (QP=28)	Average error	Maximum error	PiP PSNR Luminance (dB)
Wave	3.1	188	27
Plane	6.3	123	12
Freeway	8.4	148	25
Tractor	6.4	98	28

The experiments are performed in the following experimental setup. The computational reductions, as discussed in Section V, are incorporated in the H.264/MPEG4-AVC reference model JM18.2 to evaluate the final picture quality and executed on a PC platform for performance profiling. Although the platform deploys a multicore CPU@2.66GHz, the program does not utilize the multi-core feature and runs on a single core. The picture quality is evaluated for the two scenarios. In the first scenario, the computation-reduced intraframe decoder employs a partial calculation of the spatial prediction-block and omits the de-blocking filter. In the second scenario, next to the calculation reductions from performing partial prediction and omitting the de-blocking filter, the decoding complexity is further reduced by the partial calculation of the residual signal.

A. Decoded picture quality for decoding scenario 1

Partially calculating a spatial prediction block, results in a number of prediction pixels, which are not calculated, see Fig. 8 and Fig. 9. When applying pixel duplication, the absent prediction pixels obtain a value e.g. based on the nearest calculated neighbor, which is typically close to the original pixel value. Table IV indicates the objective picture quality for full HD decoded pictures, based on the partial calculation of the spatial prediction block and omitting the de-blocking filter. The quality of this navigation signal is first evaluated at full resolution, to mimic the case that the PVR platform is equipped with a video scaler. Although the picture quality is negatively influenced by the pixel duplication process, the objective picture quality does not drop dramatically (30-35dB), see Table IV. However, a good quality PiP can be obtained, when calculating a PiP picture from this partially decoded image, for which a 7-tap horizontal and 5-tap vertical decimation filters are used, see Table IV. The evaluated quality in the range of 40 dB is derived by comparing a down-scaled full-HD decoded picture, with the above-described PiP picture based on partial decoding. Experiments have shown that this method still offers a quite good PiP picture quality. Furthermore, an overall computation reduction of 17% in cycle consumption is obtained at 25-Hz refresh rate, see Table VI and Fig. 11. This cycle reduction is to a large extent caused by the absence of the de-blocking filter.

B. Decoded picture quality for decoding scenario 2

In the second decoding scenario, besides partially calculating the spatial prediction blocks and omitting the de-blocking filter, also the block-based residual information is partially IDCT computed. When introducing a partial inverse transformation to reduce the computation load, the final decoded picture shows strong distortion on the non-reference

TABLE VI
CONTRIBUTION TO PROPOSED COMPUTATION REDUCTION FOR SOFTWARE-BASED H.264/MPEG4-AVC INTRAFRAME DECODING.

Scenario	IDCT	Spat. Pred. 8x8 (%)	Spat. Pred. 4x4 (%)	Spat. Pred. 16x16 (%)	Memcpy (%)	Remaining code (%)	Total code (%)
1	0	7	9	25	0	23	17
2	42	7	9	25	94	23	34

pixels locations. Consequently, a PiP picture cannot be derived by spatial decimation filtering and corresponding spatial sub-sampling, since reconstruction pixels are missing. From such a distorted image, a PiP can be derived, without decimation filtering, on the basis of spatial sub-sampling only. This approach is implemented in the H.264/MPEG4-AVC decoder. When selecting pixels from the exact calculated spatial positions, which were used as the reference pixels in Scenario 1, a PiP picture is derived with a good subjective quality, see Fig. 12. However, the objectively measured quality shows an expected significant quality loss, see Table V. Fortunately, the perceived quality is much better than expected due to the unfiltered down-sampling, as aliasing is introduced contributing to the perceived sharpness, thereby justifying the decoding approach. For the situation that the original picture has a resolution of 1,920 pixels times 1,080 lines, such image is effectively reduced horizontally and vertically by a factor 4, resulting in an image of size 480 pixels times 270 lines. The subjective picture quality according to Fig. 12 is of good quality for fast-search video navigation, since potential artifacts are camouflaged due to the fact that the fast-search sequence is constructed using non-neighboring pictures. This causes successive navigation pictures to be less correlated, thereby masking some of the distortion.

C. Obtained computation reduction

The test model used for experiments is formed by a profiled modified reference model JM18.2, to reveal the performance impact of the algorithmic simplifications. From Fig. 11, it becomes clear that when omitting the post-filter and partially calculating the spatial prediction block, this results in an overall cycle load reduction of 17 %. When also partially calculating the residual block and sub-sampling the final image without decimation filtering, the computation reduction becomes 34 %.

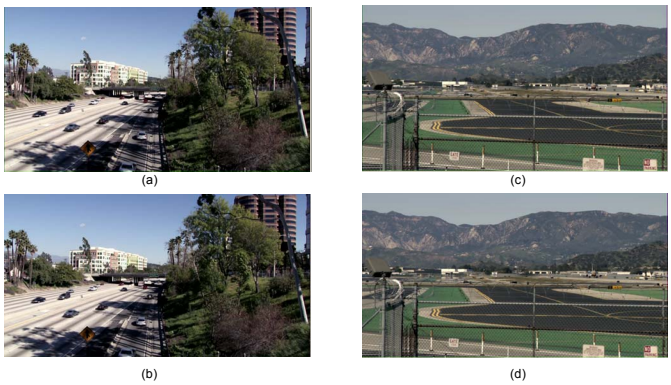

Fig. 12. PiP comparison. (a) Freeway PiP based on full decoding and decimation filter based sub-sampling. (b) Freeway PiP based on reduced calculation decoder and sub-sampling without decimation filter. (c) Airplane PiP based on traditional decoding and sub-sampling with decimation filtering. (d) Airplane PiP based on reduced-calculation decoder and sub-sampling without decimation filter.

Furthermore, the picture clearly shows the computational benefit when lowering the picture refresh rate for the fast-search video navigation signal. Table VI reveals the main functions offering the most percentages of computational reduction. The largest cycle load reduction is obtained by the *memcpy* instructions, the *inverse transformation* and the absence of the de-blocking filter, whereas the cycle load reduction for the spatial prediction block is relatively modest.

D. Normal play audiovisual fragments

The novel audio-enhanced video navigation method has been subjectively tested using a small test panel. In order to interpret audio content, the audio signal must have certain duration. Experiments, using a variety of audiovisual content, indicate that normal-play fragments with 3-second duration result in a good navigation performance. In this way, fatigue of the viewer is avoided, which may occur when exposed to fast changing audiovisual information. The usage of scene-change information is attractive for fine-tuning the normal-play fragment starting point, optimizing the selection of a sufficiently long meaningful audio fragment. Such a fine-tuning operation is particular interesting for audiovisual content with spoken text and to a lesser extent for music-oriented content.

VII. CONCLUSION

This paper has presented a double-window audio-enhanced video navigation method, which combines normal-play fragments and fast-search video trick play, providing an instant global overview, as well as detailed video information. Additional audible information enables the viewer to use both auditory and visual cues, for finding the intended video passage. This system concept has the benefit that a coarse outline of the recording is obtained via the fast-search trick-play PiP sequence, while the normal-play video fragments in full size refine the perceived impression, whereby the audio signal guides the viewer in selecting the proper video window. A further advantage of the supplementary auditory information is that the viewer does not need continuous visual attention for effective navigation, enabling to perform other activities in parallel. Although the system has a double independent video window, the system resources can be kept within acceptable boundaries for the chosen platform at hand. For navigation, this is achieved in two ways. First, lowering the refresh rate of the fast-search video-navigation signal reduces the computation load as well as the involved

bandwidth. Second, the designed system explores complexity-scalable decoding with two implementation scenarios. First, H.264/MPEG4-AVC-encoded video can be reduced in computation complexity, by partial reconstruction of specific pixels in the block-based prediction and omitting the de-blocking filter. The second scenario involves adding partial IDCT, thereby limiting the reconstruction of residual pixels. The finally obtained picture quality may have a low PSNR, but has a subjectively good quality. Potential artifacts in the fast-search video navigation signal are masked, due to the relatively large temporal distance between the normal-play pictures constructing the fast-search navigation signal.

ACKNOWLEDGEMENT.

The authors acknowledge the members of the TU/e-VCA Research group, for aid in the test panel and performance profiling.

REFERENCES.

[1] E.S. Kim S.W. Jung and D.H. Lee, “Design and implementation of an enhanced personal video recorder for DTV,” *IEEE Trans. Consumer Electronics*, vol. 47, no. 4, pp. 864-869, Nov. 2001.
[2] M.F. Demeyer, “Apparatus and method for automated video editing,” US8290334B2, Oct. 2012.
[3] V. Iverson, et al., “Transcoding media content from a personal video recorder for a portable device”, WO2003107672A1, Mrt. 2005.
[4] J. Park, “Method of searching scenes recorded in PVR and television receiver using the same,” US20070040936A1, Feb. 2007.
[5] H. Kim, J. Kim, M. Rostoker, Y.S. Seong and S. Sull, “Techniques for navigating multiple video streams,” US20060064716A1, Mar. 2006.
[6] G.G. Lee J.H. Lee and W.Y. Kim, “Automatic video summarizing tool using MPEG-7 descriptors for personal video recorder,” *IEEE Trans. Consumer Electronics*, vol. 49, no. 3, pp. 742-749, Aug. 2003.
[7] Y. Ruiy, Z. Xiongy, R. Radhakrishnanz, A. Divakaranz and T.S. Huangy, “A Unified Framework for Video Summarization Browsing and Retrieval,” Academic Press, 2005.
[8] S.H. Lee, “Personal Video Recorder and method for operating the same,” US20030128969A1, Jan. 2003.
[9] R.R. Dunton et al., “Image-Keyed Index for Video Program Stored in Personal Video Recorder,” US20060110128A1, May 2006.
[10] W.S. Herz, “Video Navigation System and Method,” US20090074377A1, Mar. 2009.
[11] O. Eerenberg, and P.H.N. de With, “Digital Video,” Chapter 15, Intech, Vukovar Croatia, Feb. 2010.
[12] E. Belinsky, E. Shahar, “Method and System for Navigation of Audio and Video Files,” US20100141655A1, June 2010.
[13] J.R. Leonardi, “Multiple time-scales of language dynamics: An example from psycholinguistics,” Taylor & Francis, Ecological Psychology, pp. 269-285, 2010.
[14] O. Eerenberg and P.H.N. de With, “MPEG-2 compliant trick play over a digital interface," *IEEE Trans. Consumer Electronics*,” vol. 51, no. 3, pp. 958-966, Aug. 2005.
[15] Chen Chen, Ping-Hao Wu and Chen H, "Transform-Domain Intra Prediction for H.264," *IEEE ISCAS*, pp.1497-1500, May 2005.
[16] Eun-Seok Kim, Tae-Woong Um, and Seoung-Jun Oh, “A Fast Thumbnail Extraction Method in H.264/AVC Video Streams,” *IEEE Trans. Cons. Electronics*, Vol. 55, No. 3, pp. 1424-1430, Aug. 2009.
[17] C. Yingwei, Z. Zhun, L. Tse-Hua, S. Peng, K. van Zon, "Regulated complexity scalable MPEG-2 video decoding for media processors," *IEEE Trans. Circuits and Systems for Video Tech.*, Vol. 12, No. 8, Aug. 2002, pp. 678-687.
[18] H. Yuen, “PVR with High-Capacity Archive,” US20020186957A1, Apr. 2002.
[19] T. Wiegand and S. Gordon, “H.264/MPEG4-AVC fidelity range extensions: tools, profiles, performance, and application areas”, in Proc. IEEE International Conf. Image Processing, Vol. 1, Sept. 2005.
[20] S.O. Mietens, P.H.N. de With and C. Hentschel, "New DCT Computation Technique based on Scalable Resources", in Proc. IEEE Workshop on Signal Processing Systems, pp. 285-296, Sept 2001.
[21] J. Bae, et al.. “An Efficient Personal Video Recorder System,". In Proc. Int. Conf. on Intell. Comput. Technol. and Autom., May 2010, pp. 501-504.

BIOGRAPHIES

Onno Eerenberg was born in Zwolle, the Netherlands, in 1966. He graduated from the Polytechnical College in Amsterdam in 1992. He joined Philips Research Laboratories Eindhoven, The Netherlands where he worked in the Magnetic Recording Systems department on digital video and data recording systems. He was involved in several European research projects in this area and was involved in the implementation of e.g. video compression systems. He received a MSc degree in engineering product design in 1998 from the University of Wolverhampton, UK. He is currently working for Trident Microsystems as a member of the video innovation team. His work resulted in various papers, book-chapters and 26 patent applications.

Ronald Aarts was born in 1956, in Amsterdam, the Netherlands. He received a BSc degree in electrical engineering in 1977, and a PhD in physics from Delft University of Technology in 1995. He joined the Optics group at Philips Research Laboratories, Eindhoven, the Netherlands in 1977 where he was involved in signal processing for optical storage systems. In 1984 he continued his work on Acoustics where he worked on the development of CAD tools and signal processing for loudspeaker systems. In 1994 he became a member of the DSP group at where he has led various research projects on the improvement of sound reproduction. His work resulted in two hundred papers and reports and over 160 patent applications. Ronald is part-time full-professor at the Eindhoven University of Technology (TU/e).

Peter. H.N de With (F’07) graduated in electrical engineering from the University of Technology in Eindhoven and received his Ph.D. degree from the University of Technology Delft, The Netherlands. He joined Philips Research Labs Eindhoven and became a member of the Magnetic Recording Systems Department. From 1985 to 1993, he was involved in several European projects on SDTV and HDTV recording. In this period, he contributed as a principal coding expert to the DV standardization for digital camcording. In 1994, he became a member of the TV Systems group at Philips Research Eindhoven, where he was leading the design of advanced programmable video architectures. In 1996, he became senior TV systems architect and in 1997, he was appointed as full professor at the University of Mannheim, Germany, at the faculty Computer Engineering, where he was heading the chair on Digital Circuitry and Simulation. Between 2000 and 2007, he was with LogicaCMG Eindhoven as a principal consultant, and also professor at the University of Technology Eindhoven, at the faculty of Electrical Engineering. Between 2008-2011, he was Vice President Video Technology at Cyclomedia Technology, The Netherlands. He is now full professor at Eindhoven University of Technology and program director of the healthcare research. He has written and co-authored over 400 papers on video coding, architectures and their realization. He regularly received the IEEE Transactions on Consumer Electronics Paper Award for co-authored papers, and the VCIP Best Paper Award. In 1996, he obtained a company Invention Award. Mr. de With is Fellow of the IEEE (2007), program committee member of the IEEE CES and ICIP, SPIE EI, honorary member of the Benelux WIC, former scientific board member of CMG, scientific advisor to CycloMedia Technology, ViNotion and board member of various other working groups.