Wyner-Ziv to H.264 Video Transcoder for Low Cost Video Encoding

J. L. Martínez, G. Fernández-Escribano, H. Kalva, W.A.C Fernando and P. Cuenca

Abstract — *This paper proposes a Wyner-Ziv / H.264 transcoder that enables low cost video applications. The proposed solution supports video encoding on resource constrained devices such as disposable video cameras, network camcorders and low cost video encoders. This approach is based on reducing encoding resource requirements on a device by using Wyner-Ziv video encoding. The system shifts the burden of complexity away from the encoder, for example to a network node, where a transcoder efficiently converts WZ encoded video to H.264 by reusing the information from the WZ decoding stage. The transcoded H.264 video is requires fewer resources than WZ decoding and therefore reduces the complexity of decoding. The complexity of encoding and playback ends of video applications is thus reduced enabling new class of consumer application. The paper is focused on reducing the complexity of the macro-block mode coding decision process carried out in H.264 encoding stage of the transcoder. Based on a data mining process, the approach replaces the high complexity H.264 mode decision algorithm by a faster decision tree. The proposed architecture reduces the battery consumption of the end-user devices and the transcoding time is reduced by 86% with negligible rate-distortion loss*[1].

Index Terms — Wyner-Ziv, Low Cost Video Encoding, H.264, Transcoding, Disposable video camera.

I. INTRODUCTION

Digital video coding today mainly relies on a hybrid of block-based transform and interframe predictive coding approaches. In these architectures, the encoder has the computationally complex task of exploiting both the temporal and spatial redundancies inherent to a video sequence. The decoder is left with a simple decoding procedure which only consists to "execute" the encoder's orders. In order to explore those spatial and temporal correlations, the encoder requires higher computational complexity, than the decoder (typically 5 to 10 times more complex [1]), mainly due to the motion estimation task. On the other hand, *Distributed Video Coding* (DVC) [1] is a technique used to reduce the asymmetry in that traditional video codecs; the processing complexity of the encoders is reduced, leading to a low-cost implementation, while the majority of the computations are taken over by the decoders.

A particular case of DVC, the so-called *Wyner-Ziv* (WZ) coding [1], deals with lossy source coding with side information at the decoder and also enables a flexible allocation of complexity between the encoder and the decoder. In this context, part or the entire motion estimation task is moving to the decoder; and it is the decoder responsibility to obtain the *side information*, an estimate of the encoded WZ frame, and the encoder only sends parity bits to improve its quality. The theoretical framework of DVC is based on the *Distributed Source Coding* (DSC) principles for lossless coding by Slepian and Wolf [2] and lossy coding by Wyner and Ziv [3]. This mathematical background states that, under the same conditions, the *Rate-Distortion* (RD) performance achieved when performing joint encoding and decoding (i.e. as in traditional video coding scheme) of two correlated sources, can also be obtained by doing separate encoding and joint decoding. That means that there is no RD loss in a DVC scenario compared to the traditional video coding approach.

In this context, DVC is a particular realization of DSC when the source is video; DVC has become a very promising approach towards the fulfillment of requirements such as low complexity and low-power encoders which are assuming a growing importance for practical consumer applications.

Nevertheless, the requirements to have low complexity at both encoder and decoder side have not been met using traditional video codecs such as [4]. The latter are more complex at the encoder side (basically due to motion estimation process), but the decoder, however, is less complex. Therefore, low cost video communications employing traditional video codecs leads to an inefficient configuration because the encoders sacrifice RD performance in order to reduce the encoding complexity by using only the lower complexity encoding tools.

This paper is focusing on low cost video encoding applications depicted in Figure 1. Low cost video encoders enable the use of video encoding in low cost devices such as toys, wireless cameras and disposable video cameras; all these devices can use a low cost video camera which can be

[1] This work was supported by the Ministry of Science and Technology of Spain under CONSOLIDER Project CSD2006-46, CICYT Project TIN2006-15516-C04-02, the Council of Science and Technology of Castilla-La Mancha under Project PAI06-0106 and FEDER.

J. L. Martínez is with the I3A, Campus Universitario s/n, 02071, Albacete, Spain; University of Castilla-La Mancha (joseluismm@dsi.uclm.es).

G. Fernández-Escribano is with the I3A, Campus Universitario s/n, 02071, Albacete, Spain; University of Castilla-La Mancha (gerardo@dsi.uclm.es).

H. Kalva is with the Department of Computer Science and Engineering,777 Glades RD,34341 Boca Raton, FL, USA; Florida Atlantic University (hari@cse.fau.edu).

W.A.C. Fernando is with Center for Communications Research, Guildford GU2 7XH, United Kingdom; University of Surrey (W.Fernando@surrey.ac.uk).

P. Cuenca is with the I3A, Campus Universitario s/n, 02071, Albacete, Spain; University of Castilla-La Mancha (pcuenca@dsi.uclm.es).

Contributed Paper
Manuscript received May 28, 2009

supported with reduced complexity encoding algorithms. The proposed transcoder can be used to convert the video to H.264 before burning on to a DVD, storing into a drive or retransmitting again to another low cost / screen receiver device.

The solution adopted in this work is a transcoder framework where the recently propose low complexity / cost Wyner-Ziv video encoding and the inherent low complexity / cost H.264 decoding algorithm can operate together to efficiently support these communications.

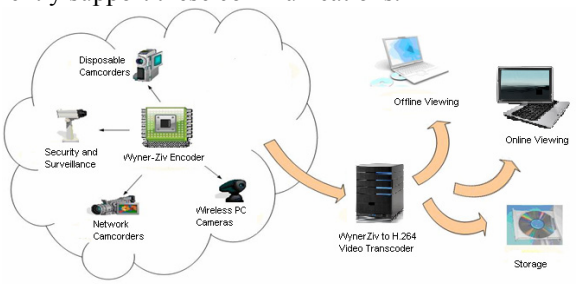


Fig. 1. Low cost communication system using a WZ/H.264 transcoder.

Therefore, in order to efficiently exploit the advantages that these two video coding paradigms can offer in terms of low complexity encoding (using WZ coders [1]) and decoding (using traditional video decoders such as [4]) this paper proposes the use of an improved WZ/H.264 video transcoder device located in the network and converts the WZ video from the lower complexity sender to H.264 video to the lower complexity. In the proposed scenario, therefore, the complexity is shifted to the transcoder device and the end points meet the low complexity constraints.

A basic transcoder performs full WZ decoding procedure on the video signal transmitted from the transmitter and then encodes it to H.264. The transcoder thus has to handle two complex processes: WZ decoding and H.264 encoding. Therefore, due to this core of computations it is much recommended to propose new techniques to reduce the complexity in order to achieve real time video communications or, at least, decrease the system delay between end points. An optimized and efficient video transcoder must accelerate the second part of it (in this case the H.264 encoding algorithm) by re-using data and calculations done at the first half (in the WZ decoding). The process more suitable for complexity reduction in the WZ/H.264 transcoder is the inter-prediction as this process takes up most computing resources in the encoding stage. On the other hand the motion estimation and the intra prediction are other tasks that could also be accelerated in the proposed transcoder.

In this paper, the proposed transcoder reuses i) the *Sum of Absolute Differences* (SAD), ii) the length of the *Motion Vectors* (MVs) both from the side information (the WZ decoding motion estimation) and iii) the reconstructed pixels can reduce the number of *Macro-Block* (MB) partitions checked and, therefore, the transcoding time. The approach is based on a *Machine Learning* (ML) process that generates a decision tree that selects only a sub-set of partition modes based on the correlation of these three parameters (SAD, MV and reconstructed pixels) with respect to the structure motion compensation done in H.264.

The paper is organized as follows: Section 2 shows the mathematical background behind the DVC technology and the basics of WZ architectures are shown in Section 3. Section 4 presents the state-of-the-art of WZ transcoder. Then, in Section 5 our proposed WZ/H.264 video transcoder is depicted and evaluated in Section 6. Finally, in Section 7, the conclusion will be shown.

II. MATHEMATICAL BACKGROUND

DSC is a new coding paradigm in which correlated sources are encoded separately and decoded jointly. DSC is based on the *Slepian–Wolf* (SW) theorem presented in [2] (Section II.A) and *Wyner-Ziv* presented in [3] (Section II.B).

Together, the SW and the WZ theorems suggest that it is possible to compress two statistically dependent signals in a distributed way (separate encoding, jointly decoding), approaching the coding efficiency of more conventional predictive coding schemes (joint encoding and decoding). This coding paradigm is known as DVC or Wyner–Ziv coding and opens the doors to new video coding schemes.

A. Slepian-Wolf Theorem

DSC is based on the SW theorem presented in 1973 [2]. The SW theorem addresses the case where two statistically dependent sources are independently encoded, and not jointly encoded as in the largely deployed hybrid coding solution. This theorem states that the minimum rate to encode the two (correlated) sources is the same as the minimum rate for joint encoding, with an arbitrarily small probability of error, if the two sources have certain statistical characteristics. This is a very interesting result in the context of the emerging coding challenges because it opens the doors to a new coding paradigm where, at least in theory, separate encoding does not induce any compression efficiency loss when compared to the joint encoding approach used in the traditional predictive coding paradigm.

B. Wyner-Ziv Theorem

While the SW theorem deals with lossless coding, in 1976, Wyner and Ziv studied the case of lossy coding with side information at the decoder [3]. Under some hypothesis on the joint source statistics and on the distortion measure the Wyner–Ziv theorem states that when the side information (i.e. the correlated source) is made available only at the decoder there is no coding efficiency loss in encoding the other source, with respect to the case when joint encoding of the two sources is performed.

III. WZ VIDEO CODING PARADIGM

The WZ video coding used in this paper follows the insight of the work developed by *Aaron et al* in [1]. This kind of

architecture is well-known in the literature and it is frequently referred as feedback based Stanford architecture. This architecture can work either in the pixel domain or in the transform domain. In [1] the video sequence is divided into key frames and Wyner-Ziv frames. The key frames are traditional intra-frame coded using, intra H.264/AVC [4]. The Wyner-Ziv frame's pixels (pixel domain) or coefficients (transform domain) are quantized and the resulting quantized symbol stream is used to extract bit planes for coding. Each bit plane is then independently turbo encoded, starting with the most significant bit plane. The parity bits produced by the turbo encoder are stored in a buffer and transmitted in small amounts upon decoder request via the feedback channel. At the decoder, the frame interpolation module is used to generate the side information frame, an estimate of the Wyner-Ziv frame, based on previously decoded frames. This technique is based on a *Motion Compensated Temporal Interpolation* (MCTI) [1]. The side information is used by an iterative turbo decoder to obtain the decoded quantized symbol stream. The decoder can request for more parity bits from the encoder via feedback channel; otherwise, the current bit plane turbo decoding task is considered successful and another bit plane starts being turbo decoded. Once, all bitplanes have been processed it calculates a reconstruction for each pixel assuming that the decoded symbols are correct, the Wyner-Ziv codec limits the distortion of each pixel up to a maximum distortion determined by the quantizer coarseness.

IV. RELATED WORK

Transcoding algorithms [5] are not new in video research community but, as far we know, there are only two WZ transcoding approaches available in the literature. One of them is based on WZ/H.263 [6] and the other one on WZ/H264 [7]. But, some years ago, different video standards combination gives way to different video transcoding such as MPEG-2 to H.264 [8], H.263 to H.264 [8] and so on.

The main objective that a transcoding process should follow is trying to figure out what calculations and process that has been carried out in the first stage could be re-used in the second half. All the information that has to be generated and could be approximated by the data gathered in the first stage is wasted computing time in the transcoding process. In fact, transcoding algorithms between traditional video coding standards are easier to accelerate due to the fact that the input and output video formats are based on more comparable paradigms. In terms of WZ video transcoder, one of the more referenced overviews of DVC [1] mentioned as one of the benefits of this new video coding paradigm the support of low cost video communications using a transcoder device. This work was presented in 2005 but it does not focus on this problem and only offers transcoding based solutions as an application of DVC. The first WZ video transcoder based approach was presented by *Peixoto et al* in [6] in 2008, which is based on a WZ/H.263 video transcoder to support this kind of communications. They [6] proposed a mapping between the WZ *Group of Pictures* (GOP) and the traditional GOP and, moreover, for the P or B slices in H.263 some *Motion Estimation* (ME) refinement was also proposed. However, they [6] failed to exploit well the correlation between the WZ MV and the traditional ME and only uses them to determine the starting center of the H.263 ME process. These drawbacks were improved in our previous work, in [7]. The same authors of this paper, in 2009 proposed an improved WZ/H.264 video transcoder that reuse the incoming MV of the side information generation in order to reduce the ME process done in H.264. This work [7] is based on a dynamic search window and search re-definition per each MB or sub-MB partition done in the H.264 ME process. The length and the orientation of the WZ decoded MV are used to reduce and focus the ME done at H.264 [7]. In the present work, on contrary, the ME itself is kept untouched but, the different MB coding mode partitions are reduced into a sub-set based on the decision tree (as we will show in Section V). The proposed approach is another step in our WZ/H.264 video transcoder and it is focused only on the MB coding mode partition process itself.

V. PROPOSED WZ/H.264 VIDEO TRANSCODER

In the framework of WZ/H.264 transcoders, in the inter-frame coding of H.264 standard, there are inter partition modes and intra modes to be taken into account for determining the best mode. Although H.264 can achieve higher coding efficiency than any other previous coding standard, the computation complexity also increases significantly. In the inter-frame coding of H.264, seven different block division modes (16x16, 16x8, 8x16, 8x8, 8x4, 4x8 and 4x4) can be selected for the motion estimation prediction. Moreover, H.264 adopts the spatial domain intra prediction in the block sizes 16x16 and 4x4 includes four and nine directional predictions, respectively. Therefore, the H.264 encoder part of the WZ/H.264 transcoder takes a large amount of time to search exhaustively all inter modes and intra modes for inter-frame coding. The final MB coding mode decision is carried out by taking into account the amount of the residual between the current block and previous /past ones inside the search range window and the length of the MV.

In the next subsections we will describe the motivations behind of the use of ML to accelerate this mode decision (Section V.A), the training stage done in order to generate the different decision trees (Section V.B), the decision tree themselves (Section V.C) and finally the proposed architecture of the improved WZ/H.264 transcoder (Section V.D).

A. Motivations

In WZ video decoding, the side information generation process is the procedure where an estimation of the current frame (available only at the encoder) is constructed at the

decoder (see Section III). The decoder generates the side information frame using motion-compensated interpolation techniques such as MCTI [1]. The side information process is crucial to any DVC framework, and will be of greater relevance to the transcoder. The side information first uses the previous reconstructed key frame as the reference and the next reconstructed frame as the source to calculate the forward MVs (MV_F). Then, it uses the next reconstructed frame as the reference and the previous reconstructed frame as the source to calculate the backward MVs (MV_B). It then uses $MV_F/2$ on the previous reconstructed frame soured to calculate frame PF and uses $MV_B/2$ on the next reconstructed frame to generate the frame P_B. The final side information is considered as the mean between P_F and P_B. In both calculations the block size used is 16x16 and the search range is fixed to 16.

Therefore, the traditional ME done at H.264 in the encoding stage of the proposed transcoder has a high correlation with the side information developed in the decoding stage (this can be seen as the ME done at WZ decoding). In fact, the MVs generated in P frames in H.264 are correlated with the side information MVB and, also, the MVs of the B frames in H.264 ME are correlated with both MV_B and MV_F.

Moreover, we found in some experimental observation, that the stationary areas or object with slow motion or with slow camera motion are often coded in inter mode for inter-frame coding with higher block partition (such as 16x16,16x8 or 8x16) or even as *Skipped*. On the other hand, the regions with scene change, with light change or with the object which appears suddenly are coded in inter-mode with lower MB mode partition (such as 4x4) or even to *Intra* mode for the inter-frame coding.

In fact, this is one of the motivations for our approach, the SAD calculation derived for the motion compensation PB frame could be utilized to determine the similarity between the current block and the corresponding block in the previous frame (in a P frame). Therefore, the histogram difference can be expressed as in (1):

$$SAD=\sum_{(x,y)\in B}\left|X_{next}(x,y)-X_{previous}(x+dx,y+dy)\right| \quad (1)$$

Where motion vector with components (dx, dy) is applied to block B. X_{next} and $X_{previous}$ are the next and previous reconstructed key frames. In a similar way, we can extend this observation to B frames between the mean of P_F and P_B. But the B frames treatment is out of the scope of this paper since we focus on IPPP H.264 pattern which is the most suitable GOP pattern for real time low cost communications where no buffer are needed in the devices and the complexity is kept to its minimum. If SAD computation is small, it means that the current blocks in two adjacent frames changes slightly. For instance, the SAD difference will be smaller in a still background. In this case, the selected probability of inter coding mode is very high. Conversely, if SAD is large, it means that scene has some change, new objects appear or the objects move fast, the intra coding mode is selected in inter-frame coding. Hence, SAD is one of factors for determining whether the coding mode of block should be intra skip.

We empathise that the SAD procedure is not a new calculation done to be passed through the H.264 encoder part (the second half of the transcoder), this SAD calculation is done in the backward motion estimation in the side information generation process in order to find the best MV per block in the WZ decoder. Therefore, our approach only has to store the SAD computation which determines the optimum MVs for this block.

In some situations, although SAD is large, the blocks are still chosen as inter mode blocks. For example, the objects with uniform and fast motion in the block cause the SAD to be large. However, in this case, inter mode coding performs better than intra mode. To avoid this problem, the proposed algorithm adds another factor to present the temporal correlation of the MB in two adjacent frames: the length of the MVs.

In the procedure to determine the optimum MB coding mode partition reusing some information derived from the WZ decoding, we also found a high correlation between the length of the MVs generated in the P_B generation frame with respect to final MB partition decision. In fact, long MVs suggest a more complicated MB partition such as 4x4 whereas simpler MB partition deals with shorter MVs.

Finally, once the bidirectional motion compensated interpolation is carried out according to [1], the interpolation frame itself is more accurate with respect to the original reference for the areas where the movement and the detail is lower. On contrary, poor estimated blocks deal with areas with higher movement and higher information. As we said before in Section III, the turbo decoder task is to try to correct the side information mismatches using the parity bit sent by the encoder. Once, the turbo decoding algorithm is successful, the reconstruction is performed; taking the corresponding side information sample if it falls into the quantization interval. Based on the observations, we found that the number of reconstructed pixel index that differs from the side information quantization bin is higher for the blocks that are poorly estimated by the side information and these blocks are then mapped into more complex MB partitions such as 4x4 or Intra. Based on this observation, the parity bit information sent by the encoder is also taken into account by our algorithm in order to determine the final MB partition decision.

Figure 2 shows the correlation between SAD, MVs length and pixels outside of the reconstruction bin with respect to the MB coded partition done in H.264. It tries to collect the motivation behind of our approach in a visual way; Figure 2a shows the original second frame (the first P frame) of the well-known *flower and garden* sequence; Figure 2b shows the amount of SAD available in the P_B frame (with corresponds to compensated frame in H.264); Figure 2c shows the MVs of the P_B frame according to the side information process and Figure 2d shows the final MB mode codec mapping according to this frame encoded with H.264 as P frame. Figure 2e shows the MCTI done as side information according to [1].Finally, Figure 2f shows the distribution of pixels that fall off outside of the side information interval bin.

In a nutshell, there are three types of information extracted from the WZ decoding algorithm that we found are much correlated with respect to the MB coding mode partition performed in a P frame in H.264. They can be used in a ML process in order to convert this knowledge into faster rules that can replace the more complex MB coding mode decision procedure. These are the following:

1) The SAD residual information per block for the backward motion compensated frame develops in the MCTI side information.

2) The length of the MVs generated in the backward (as equivalent to P frame in H.264) MCTI for this 16x16 block.

3) The amount of pixels that fall off inside to interval bin in the reconstruction process.

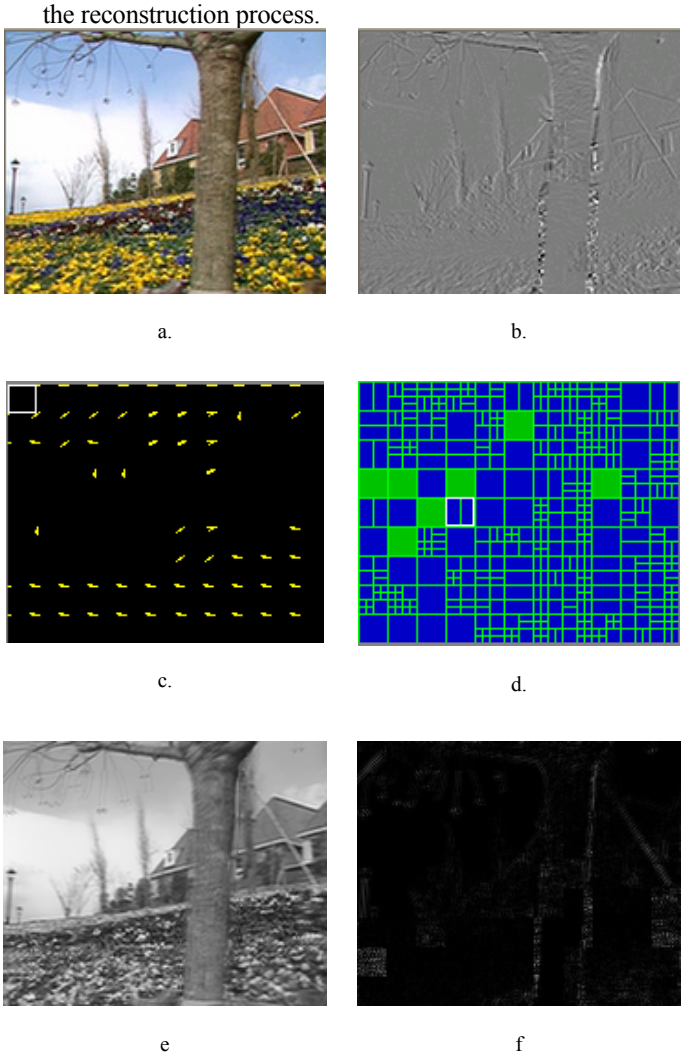


Fig. 2. Exploiting the correlation using Machine Learning.

B. Training Stage

The software used in this data mining process was WEKA [9]. WEKA is a collection of machine learning algorithms for data mining tasks. WEKA contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization and it is an open source tool available at [10].

It is noted that ML is a well-known technology which has the decision making ability with low computation complexity, basically, if-then-else operations. In this framework, we used ML tools in order to convert the relationships that have been enumerated in the Section V.A into rules in order to decide in a faster way (without call the high complexity MB mode partition algorithm) what could be the possible partitions. Therefore, this data mining process is developed as follows: We introduced all this knowledge converted into mathematical values (SAD value, MVs length, number of recon differences) and the final MB partition as variable to understand or to predict. In fact, each instance to understand is formed by an array of variables and one variable to predict. In this case, the latter is the MB mode partition.

We extracted the information (correlated to the class variable) per each MB and we call the ML learning algorithm. In this work the well-known C4.5 [9] algorithm proposed by *Ross Quinlan* has been called as ML algorithm. The training file was generated using 10 frames of QCIF *flower and garden* sequence. In terms of knowledge acquisition, we use one sequence which contains all kind of different context possibilities as well as different details, in this ML process the well-known *flower and garden* sequence was used as the training sequence. The ML algorithm gives us a decision tree formed by MB statistics variables and classifies each MB into a set of different MB mode partitions.

In this process, we applied supervised learning because we found under some experimentation, that there are MB partitions that are more correlated between them than others. Basically, the final decision follows a binary-decision tree where it has been created per level according to the relations between MB partitions. In this way the training process was develop as following levels:

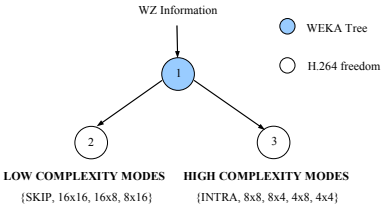
- **1st Level.** For dividing between *LOW COMPLEXITY* and *HIGH COMPLEXITY* modes. The first one is formed by {SKIP, 16x16, 16x8, 8x16} and the second one by {8x8, 8x4, 4x8, 4x4 and INTRA}
- **2nd Level.** Inside the LOW COMPLEXITY tree, we divide it in two leaves. {SKIP, 16x16} and {16x8, 8x16}.
- **3rd Level.** Inside the HIGH COMPLEXITY tree, we divide it in two leaves: {8x8, 8x4 and 4x8} and {4x4 and INTRA}.
- **4th Level**. Continuing with the tree formed by {8x8, 8x4 and 4x8}, we split up in into two leaves: {8x8-4x4 DCT, 8x8-8x8 DCT} and {8x4, 4x8}.

In a nutshell, the final tree was generated step by step, taking into account the similarity between groups of partitions. For each decision level, the ML process carried out was denoted in root node of each tree. The different tree approaches are shown in Figure 3 and Figure 4 in order to efficiently replace the more complex MB coding mode decision done in H.264.

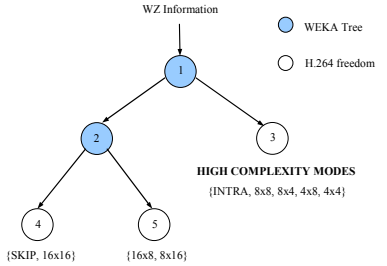
C. Decision Trees

The different decision trees proposed in this paper as a solution to replace the MB coding mode decisions in WZ to H.264 video transcoders are formed by leafs and branches as Figure 3 shows. The tree leafs are the classifications and the branches are the features that lead to a specific classification.

A tree decision is a classifier based on a set of attributes allowing us to determine the category of an input data sample. The blue circles in Figure 3 and Figure 4 represent decision tree and the white circles mean into a set of MB partition where the reference standard can choose. In other words, the proposed technique does not focus the final MB partition for the input block but it focus the different selections into a reduced set based on the correlations between the variables mentioned in Section V.A and the final MB mode selection.



a.



b.

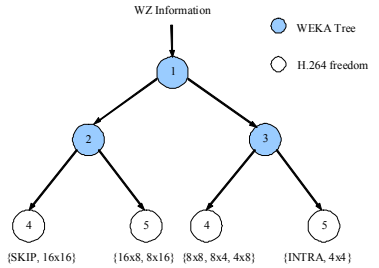
Fig. 3. LOW COMPLEXITY trees

Figure 5 shows the decision tree developed for the first level decision of our approach that corresponds to Figure 3.a. On the other hand, Figure 6 shows the decision tree for the 2nd decision level; the MB that has been selected as *LOW COMPLEXITY MODES* for the 1st decision tree, the 2nd decision level split up this bin into two different leafs based on the correlated information. The different decision trees for the 3rd and 4th decision level follows the same principles as Figure 5 and Figure 6 but, due to space limitations so they will be omitted. In Figure 6 the variable *residual4x4[index]* means the partial amount of the SAD residual available in the *index-th* 4x4 sub-block of the16x16 MB.

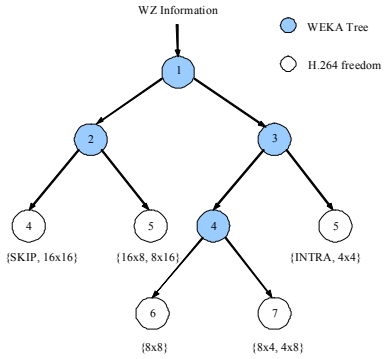
D. Proposed Architecture

The end-user devices of the proposed scenario employ a WZ encoder and a H.264 decoder; the lower complexity parts in both paradigms (as Figure 1 shows). The WZ encoder used in the sender device is our previous WZ architecture based on *Turbo Trellis Coded Modulation* (TTCM) that was proposed in [11]. This architecture is based on *Transform Domain* (TD) and TTCM codes; TTCM is a well known channel coding technique used to optimize the bandwidth requirements while protecting the information bits by increasing the size of the symbol constellation.

The WZ encoding algorithm works as follows: some frames (called key frames) are coded with a regular intra-frame encoder (in this case H.264/AVC Intra). The other frames (called WZ frames) are encoded using the algorithm depicted in Section III. Due to that at the WZ sender K frames are encoded using H.264 intra; these frames are passed through the receiver without any transcoding process as I frame in the transcoder device as shown in the Figure 7.



a.



b.

Fig. 4. HIGH COMPLEXITY trees

```
if(residual <= 74)
{
  if(motion <= 2)
   return LOW_COMPLEXITY_MODES;
  else if(motion > 2)
  {
   if(RCN <= 0)
    {
        if(residual <= 61)
            return LOW_COMPLEXITY_MODES;
        else if(residual > 61)
            return HIGH_COMPLEXITY_MODES;

    }else if(RCN > 0)
        return HIGH_COMPLEXITY_MODES;
  }
  }else if(residual > 74)
        return HIGH_COMPLEXITY_MODES;
}
```

Fig. 5. First level decision tree.

Related to H.264 [4] the standard reference decoder has been introduced. In fact, the traditional video decoder (receiver device) just performs entropy decoding, motion compensation and inverse transform. We emphasize that only these are the algorithms that have been implemented in the end devices: WZ encoding and H.264 video decoding.

On contrary, all the computations are taken over by the transcoder which is the major contribution of this paper and is depicted in Figure 8. Our previous WZ decoder [11] is part of the transcoder as well as a H.264 encoder [4]. Instead of re-encoding the sequence, the transcoder uses information that

was calculated in the WZ decoding process for speeding up the transcoding (that is depicted in Figure 8 by a dotted line connected both WZ decoder and H.264 encoder). The WZ decoder firstly generates the side information using past and previous key frames based on [11]. Once the side information is generated the TTCM decoder starts to correct the possible mismatches / errors between this side information and the original frame using the parity information sent by the WZ encoder (sender device). Some information used in the side information generated and the reconstruction function is passed to H.264 encoder (second half of the transcoder) in order to reuse it and reduce the complexity of the H.264 encoding algorithm (as has been depicted in Section IV.A).

```
if(residual4x4[1] <= 1)
    return SKIP-16X16;
else if(residual4x4 [1] > 1)
{
   if(residual4x4 [2] <= 5)
   {
     if(residual <= 29)
           return SKIP-16X16;
     else  if(residual > 29)
     {
        if(residual4x4 [10] <= 1)
           return 16X8-8X16;

        else if(residual4x4 [10] > 1)
           return SKIP-16X16;
     }
   }else

   if(residual4x4 [2] > 5)
     return 16X8-8X16;
}
```

Fig. 6. Second level decision tree.

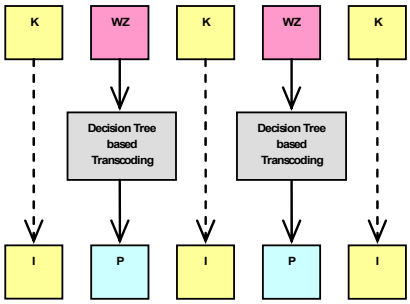


Fig. 7. The proposed transcoder scheme

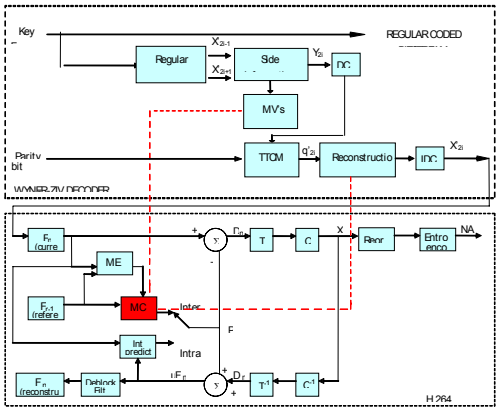


Fig. 8. The proposed WZ/H.264 video transcoder

In this paper, the tree structured motion compensation algorithm developed in the H.264 encoding algorithm has been accelerated using the decision trees generated in the data mining process (that has been depicted in Section IV.B). Therefore, in the improved transcoder the high complexity tree structured motion compensation algorithm has been reduced and only a small set of MB partitions could be selected by the algorithm. The distribution of the partition in each set is fixed by our decision tree (basically, a tree of decision trees) that is the major contribution of this paper.

Therefore, in this paper we propose a new framework where the traditional video coding (represented by H.264) and the new WZ video coding paradigm can live together. This new framework brings us real time low cost video communications as well as other functionalities that is much recommended in the consumer electronics requirements. Moreover, the system delay of the proposed scheme based on WZ/H.264 transcoder is reduced using the proposed decision trees.

VI. PERFORMANCE EVALUATION

To investigate the performance of the WZ/H.264 transcoder proposed in this paper we implemented it in the H.264/AVC reference software 13.2 version [4]. Basically we have concatenated our TTCM based WZ decoder [11] to the H.264 encoder reference software and, some information has been introduced to the H.264 inter-prediction algorithm from the WZ decoder stage (as Figure 8 illustrates). The tests were carried out with the popular sequences whose main characteristics are shown in Table I.

Basically, the incoming video from WZ sender was generated using a fixed QP matrix generation as a trade-off between quality, bit rate and encoder complexity.

TABLE I

Sequence characteristics		
Sequence name	Format	N# frames
Akiyo	QCIF	150
Foreman	QCIF	150
Mobile	QCIF	150
Paris	QCIF	150

In order to evaluate the time saving of the fast MB mode decision algorithm, the calculation defined in equation (2) is used to find the time differences. Let T_{JM} denote the coding time used by the H.264/AVC JM 13.2 [4] reference software encoder for the motion estimation and compensation process, and T_{FI} be the time taken by the algorithm proposed or the mechanism that has been evaluated; ΔTime is defined as (Equation 2):

$$\Delta Time(\%) = \frac{T_{FI} - T_{JM}}{T_{JM}} \times 100 \qquad (2)$$

T_{FI} is also included all the computational cost for the operations needed to prepare the SAD information for our approaches, the reconstruction information and so on.

The detail procedures in calculating these differences can be found from a JVT document authored by *Bjøntegaard* [12], which is recommended by JVT Test Model Ad Hoc Group [13]. Note that bitrate and PSNR differences should be regarded as equivalent, i.e., there is either the decrease in PSNR or the increase in bitrate, but not both at the same time.

One of the outcomes is supposed to be RD-plots where PSNR and bitrate differences between two simulation conditions may be read. This mechanism is proposed for finding numerical averages between RD-curves as part of the presentation of results. This is a more compact and in some senses more accurate way to present the data and comes in addition to the RD-plots. The method for calculating the average difference between two such curves is as follows:

- Fit a curve through 4 data points.
- Based on this, find an expression for the integral of the curve.
- The average difference is the difference between the integrals divided by the integration interval.

For showing transcoding results, the experiments were carried out on the test sequences with the 4 quantization parameters, i.e., QP = 28, 32, 36 and 40 as specified in *Bjøntegaard and Sullivan´s* common test rule [13]. The YUV files that will be compared for getting the PSNR results are the original YUV file at the input of the WZ encoder and that one that will be obtained after decode the H.264 video with an H.264 decoder.

The H.264 encoder was run with RD-off because this is the lowest complex mode that is more suitable for low cost video communications. The H.264 parameter configuration used in the simulation was the baseline profiles with all parameters in the configuration file are set to default values of H.264 JM encoder. Only three parameters have been modified:

- *NumberReferenceFrames*. By default it is 5 but it is fixed to 1. Our main goal is to do this in real time, so we reduce the complexity by selecting a single reference frame.
- *SearchMode*. It is set to -1. Full Search Range mode for the motion estimation and compensation process.
- *SearchRange*. It is fixed to 16. By default is set to 32. 16 pixels are enough for transcoding the sequences. Moreover, it is closer to the search range used for side information generation process.

The baseline profile is chosen because it is the common profile used in most of the real-time application, such as video and mobile TV and video conference. In order to show the performance evaluation of our approach we split the results in four different scenarios which correspond to different expansions of the tree. As the tree expansion increases, the ΔTime increases on contrary as RD performance decreases

A. Level 1

In this section, we develop the first level of the decision tree which is showed in Figure 3.a. In this case, the proposed tree is only used to discriminate between LOW COMPLEXITY and HIGH COMPLEXITY modes. Then, once this decision has been taken (according to the tree algorithm showed in Figure 5) we leave to the H.264 video standard to choose between one MB mode partitions inside each bin.

TABLE II

Sequence	Format	ΔTime (%)	ΔPSNR (dB)	ΔBitrate (%)
Akiyo	QCIF	56,65%	0,004	-0,12
Foreman	QCIF	57,81%	-0,007	0,12
Mobile	QCIF	61,06%	-0,021	0,47
Paris	QCIF	61,60%	0,001	-0,01
Mean	***QCIF***	***59,28%***	***-0,006***	***0,115***

Table II contains the result for this 1st level decision for the sequences under study (Table I). It shows, the average time reduction is up to 59% with no penalty PSNR and at an increase of the bitrate of 0,12%. It is also noted that, for *akiyo* and *paris* sequences the PSNR achieved by our approach is still better than the reference one; this is due to the *Sum of Absolute Errors* (SAE) mode coding operation of the H.264 video encoder is not the best RD solution.

B. Level 2

Table III shows the performance evaluation of the second level of our decision tree which corresponds to the decision between {SKIP,16x16} or {16x8,8x16} that can appear inside to the LOW COMPLEXITY bin.

As Table III shows, the ΔTime is reduced around 82 % with a RD penalty of 0,028 quality drop and at an increase of 0,72 in bitrate.

TABLE III

Sequence	Format	ΔTime (%)	ΔPSNR (dB)	ΔBitrate (%)
Akiyo	QCIF	83,24%	0,000	-0,03
Foreman	QCIF	80,12%	-0,059	1,56
Mobile	QCIF	78,56%	-0,033	0,80
Paris	QCIF	86,48%	-0,021	0,55
Mean	***QCIF***	***82,10%***	***-0,028***	***0,720***

C. Level 3

Table IV shows the performance evaluation of the third level of our decision tree which corresponds to the decision between {4x4, INTRA} or {8X8, 8X4, 4x8} that can appear inside to the HIGH COMPLEXITY bin.

As Table IV shows, the ΔTime is reduced around 84 % with a RD penalty of 0,030 quality drop and at an increase of 0,76 in bitrate.

TABLE IV

Sequence	Format	ΔTime (%)	ΔPSNR (dB)	ΔBitrate (%)
Akiyo	QCIF	83,84%	0,000	-0,02
Foreman	QCIF	82,45%	-0,064	1,69
Mobile	QCIF	83,18%	-0,034	0,82
Paris	QCIF	87,02%	-0,021	0,55
Mean	***QCIF***	***84,12%***	***-0,030***	***0,760***

D. Level 4

TABLE V

Sequence	Format	ΔTime (%)	ΔPSNR (dB)	ΔBitrate (%)
Akiyo	QCIF	84,57%	-0,030	0,00
Foreman	QCIF	84,31%	-0,069	1,82
Mobile	QCIF	84,96%	-0,035	0,84
Paris	QCIF	88,04%	-0,021	0,55
Mean	***QCIF***	***85,47%***	***-0,039***	***0,803***

Finally, Tables V shows the results in terms of ΔTime, ΔPSNR and ΔBitrate for the final decision tree expansion (which corresponds to Figure 4.b). Also, in this table, the average result of all the sequences for each resolution is shown. In this way, an idea about a normal operation of a transcoder over all kind of video contents can be extrapolated from this result. Compared with the cascade WZ to H.264 reference transcoder, the proposed transcoder has a PSNR drop of at most 0.039 dB for a given bitrate; for the average of all the sequences. This negligible drop in RD performance is more than the offset by the decrease in computational complexity, which has been reduced around an

86% for the average of all the sequences. Time reduction is more than a requirement in real-time WZ to H.264 transcoders, since it determines the incoming stream delay in the end-users devices. This time reduction with the negligible RD penalty loss brings us a practical solution to support low cost video encoding applications that are critical to a large class of consumer electronics.

VII. CONCLUSIONS

In this paper, a framework where traditional video coding and the new WZ video coding paradigms can operate together has been proposed. This configuration can efficiently support low cost video encoding applications with low complexity constraints in the end user devices. Therefore, the battery consumption of the sender and receiver devices can be reduced significantly which is a key requirement for these applications. In the proposed scenario the burden of complexity is shifted away from the encoder and to a transcoder. The proposed transcoder allows lower complexity decoding on the playback devices. The proposed transcoder reduces the complexity of macro block mode decision procedure in the H.264 encoding algorithm based on the correlation of some WZ encoding attributes. The results show a reduction in complexity of up to 85% with negligible RD penalty.

REFERENCES

[1] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed Video Coding," Proceedings of the IEEE, vol. 93, pp. 71-83, Jan. 2005.

[2] D. Slepian, J.K. Wolf, "Noiseless Coding of Correlated Information Sources," IEEE Trans on Inf Theory, Vol. 19, pp. 471–480, July 1973.

[3] A. D. Wyner and J. Ziv, "The Rate-Distortion Function for Source Coding with Side Information at the Decoder". IEEE Transaction on Information Theory, Vol. IT-22, pp. 1–10, January 1976.

[4] ISO/IEC International Standard 14496-10:2003, "Information Technology – Coding of Audio – Visual Objects – Part 10: Advanced Video Coding".

[5] A. Vetro, C. Christopoulos, H. Sun, "Video transcoding architectures and techniques: An overview." IEEE Signal Processing Magazine, vol. 20, pp. 18–29, Mar. 2003.

[6] E. Peixoto, R. L. de Queiroz, and D. Mukherjee, "Mobile video communications using a Wyner-Ziv transcoder", SPIE, San Jose, CA, USA, Jan. 2008.

[7] J. L. Martínez, G. Fernández-Escribano, H. Kalva and P. Cuenca, "Motion Vector Refinement in a Wyner-Ziv to H.264 Transcoder for Mobile Telephony," submitted to IET Image Processing Letter.

[8] G. Fernández-Escribano, J. Bialkowski, J. A. Gámez, H. Kalva, P. Cuenca, L. Orozco-Barbosa, A. Kaup, "Low-Complexity Heterogeneous Video Transcoding Using Data Mining," IEEE Transactions on Multimedia, Vol. 10, No. 2, pp. 286-299, 2008.

[9] I. H. Witten and E. Frank, "Data Mining: Practical Machine Learning Tools and Techniques," 2nd Ed., Morgan Kaufmann, 2005.

[10] WEKA tool. (http://www.cs.waikato.ac.nz/ml/weka/)

[11] J. L. Martinez, W.A.R.J. Weerakkody, W.A.C. Fernando, G. Fernandez-Escribano, H. Kalva and A. Garrido, "Distributed Video Coding using Turbo Trellis Coded Modulation," The Visual Computer, Vol. 25. No 1, pp. 69-82, January 2009.

[12] G. Bjøntegaard, "Calculation of Average PSNR Differences between RD-Curves," presented at the 13th VCEG-M33 Meeting, Austin, TX, April 2001.

[13] JVT Test Model Ad Hoc Group, "Evaluation Sheet for Motion Estimation," Draft version 4, February 2003.

José Luis Martínez (M'07) received his B.Sc. and *M.S. degrees in Compute*r Science from the University of Castilla-La Mancha, Albacete, Spain in 2005 and 2007 respectively. He is completing his PhD in the *Instituto de Investigación en Informática* (I3A) in Albacete, Spain. His research interests include Distributed Video Coding (DVC) and, video transcoding. He has also been a visiting researcher at the Florida Atlantic University, (USA) and CCSR, at the University of Surrey (UK). He has over 20 publications in these areas in international refereed journals and conference proceedings. He is a member of the IEEE.

Gerardo Fernández Escribano (M'05) received his M.Sc. degree in Computer Engineering, in 2003, and the Ph.D. degree from the University of Castilla-La Mancha, Spain, in 2007. In 2004, he joined the Department of Computer Engineering at the UCLM, where. His research interests include multimedia standards, video transcoding, video compression, video transmission, and machine learning mechanisms. He has also been a visiting researcher at the Florida Atlantic University, Boca Raton (USA), and at the *Friedrich Alexander Universität, Erlangen-Nuremberg* (Germany).

Hari Kalva (SM'92-M'00-SM'05) is an Associate Professor in the Department of Computer Science and Engineering at Florida Atlantic University. Dr. Kalva is an expert on digital audio-visual communications systems with over 16 years of experience in multimedia research, development, and standardization. He has made key contributions to the MPEG-4 Systems standard and also contributed to the DAVIC standards development. His research interests include pervasive media delivery, content adaptation, video transcoding, video compression, and communication. He has over 100 published papers and eight patents (12 pending) to his credit. He is the author of two books and co-author of several book-chapters. Dr. Kalva received a Ph.D. and an M.Phil. in Electrical Engineering from Columbia University in 2000 and 1999, respectively. He received an M.S. in Computer Engineering from Florida Atlantic University in 1994, and a B. Tech. in Electronics and Communications Engineering from N.B.K.R. Institute of Science and Technology, S.V. University, Tirupati, India in 1991.

W.A.C. Fernando (M'00-SM'05) W.A.C. Fernando received the B.Sc. Engineering degree (First class) in Electronic and Telecommunications Engineering from the University of Moratuwa, Sri Lanka in 1995 and the MEng degree (Distinction) in Telecommunications from Asian Institute of Technology (AIT), Bangkok, Thailand in 1997. He completed his PhD at the Department of Electrical and Electronic Engineering, University of Bristol, UK in February 2001. Currently, he is a senior lecturer in signal processing at the University of Surrey, UK. Prior to that, he was a senior lecturer in Brunel University, UK and an assistant professor in AIT. His current research interests include Distribute Video Coding (DVC), QoE, 3D video coding, and intelligent video encoding for wireless communications, OFDM and CDMA for wireless channels, channel coding and modulation schemes for wireless channels. He has published more than 175 international papers on these areas. He is a senior member of IEEE and a fellow of the HEA, UK.

Pedro Cuenca (M'95) received his M.Sc. degree in Physics (award extraordinary) from the University of Valencia in 1994. He got his Ph.D. degree in Computer Engineering in 1999 from the Polytechnic University of Valencia. In 1995 he joined the Department de Computer Engineering at the University of Castilla-La Mancha. He is currently a Full Professor of Communications and Computer Networks and Dean of the Escuela Superior de Ingeniería Informática in Albacete. He has also been a visiting researcher at The Nottingham Trent University, Nottingham (England) and at the Multimedia Communications Research Laboratory, University of Ottawa (Canada). His research topics are centered in the area of wireless LAN, video compression, QoS video transmission and error-resilient protocol architectures. He has published over 100 papers in international Journals and Conferences. He has served in the organization of International Conferences as Chair, Technical Program Chair and Technical Program Committee member. He is the Chair of the IFIP 6.8 Working Group and a member of the IEEE.