Distributed Video Coding for Multiview and Video-plus-depth Coding

Salmistraro, Matteo

Publication date:
2014

Document Version
Peer reviewed version

Distributed Video Coding for Multiview and Video-plus-depth Coding

Matteo Salmistraro

April 2014

DTU Fotonik
Department of Photonics Engineering

Coding & Visual Communication
DTU Fotonik
Technical University of Denmark
2800 Kgs. Lyngby
DENMARK

To my family

Abstract

The interest in Distributed Video Coding (DVC) systems has grown considerably in the academic world in recent years. With DVC the correlation between frames is exploited at the decoder (joint decoding). The encoder codes the frame independently, performing relatively simple operations. Therefore, with DVC the complexity is shifted from encoder to decoder, making the coding architecture a viable solution for encoders with limited resources. DVC may empower new applications which can benefit from this reversed coding architecture. Multiview Distributed Video Coding (M-DVC) is the application of the DVC principles to camera networks. Thanks to its reversed coding paradigm M-DVC enables the exploitation of inter-camera redundancy without inter-camera communication, because the frames are encoded independently.

One of the key elements in DVC is the Side Information (SI) which is an estimation of the to-be-decoded frame. Another key element is the Residual estimation, indicating the reliability of the SI, which is used to calculate the parameters of the correlation noise model between SI and original frame. In this thesis new methods for Inter-camera SI generation are analyzed in the Stereo and Multiview scenarios. Furthermore on-line correlation noise models are proposed. On-line models are needed to enable the codec to be used in realistic scenarios. Focus is put on developing and investigating robust fusion techniques, able to correctly fuse various SIs. Learning algorithms for improving the fusion procedure are also explored in this work.

Optical Flow (OF) is a powerful motion estimation technique, enabling precise and flexible calculation of a Motion Vector (MV) for each pixel of the frame. The high density of MVs has discouraged the use of OF in conventional predictive coding, because of the need to code

the MVs. On the other hand DVC can exploit OF because the Motion Estimation (ME) is only performed at the decoder. In this thesis it is proposed to use OF for joint disparity and motion calculation in M-DVC and for joint motion estimation in texture and depth frames in video-plus-depth.

Rate Adaptive (RA) error correcting codes are the core of all the modern DVC codecs, nevertheless they suffer from efficiency problems, most notably for short block lengths and high correlations between SI and original signal. A novel coding architecture based on RA BCH (Bose-Chaudhuri-Hocquenghem) codes has been presented, along with an analytic model for predicting its performance and many different methods to improve the reliability of the decoded results.

Resumé

I den akademiske verden er interessen for Distributed Video Coding (DVC) systemer vokset betragteligt i de senere år. Med DVC bliver korrelationen mellem frames udnyttet i afkoderen (fælles afkodning). Indkoderen koder hver frame uafhængigt og udfører kun relativt simple operationer. Med DVC bliver kompleksiteten derfor flyttet fra indkoderen til afkoderen, hvilket gør kodningsstrukturen til en oplagt løsning for indkodere med begrænsede ressourcer. DVC kan skabe vej for nye applikationer som kan have gavn af denne omvendte kodningsarkitektur. Multiview Distributed Video Coding (M-DVC) er brugen af DVC teknikkerne i kamera netværk. Takket være det omvendte kodningsparadigme hvor frames indkodes uafhængigt, gør M-DVC det muligt at udnytte redundansen kameraer imellem uden direkte indbyrdes kommunikation.

Et af de centrale elementer i DVC er Side Information (SI), som er en estimering af en frame før den afkodes. Et andet centralt element er estimeringen af residualet, som indikerer troværdigheden af SI'en og som bruges til at beregne parametrene i modellen for korrelationsstøj mellem SI og den originale frame. I denne afhandling analyseres nye metoder til at generere SI mellem kameraer i Stereo og Multiview tilfældene. Der fokuseres på udviklingen og undersøgelsen af robuste fusionsteknikker, som kan fusionere forskellige SI'er. Læringsalgoritmer til løbende at forbedre fusionsproceduren bliver også undersøgt.

Optical Flow (OF) er en effektiv teknik til bevægelsesestimering, som understøtter præcis og fleksibel beregning af Motion Vectors (MVs) for hver pixel i en frame. Den høje densitet af MVs gør at OF er upraktisk i konventionel prædiktiv kodning, da hver MV skal kodes. I modsætning hertil kan DVC bedre udnytte OF da Motion Estimation (ME) kun udføres i afkoderen. I denne afhandling bruges OF til fælles disparitet-

og bevægelses-beregning i M-DVC og til fælles bevægelsesestimering i tekstur og dybdebilleder i video med dybde.

Rate Adaptive (RA) fejlkorrigerende koder er kernen i alle moderne DVC kodeks, selvom de er plaget af et effektivitetsproblem som er mest udpræget for korte bloklængder og høj korrelation mellem SI og det originale signal. En ny kodningsarkitektur baseret på RA BCH (Bose-Chaudhuri-Hocquenghem) koder præsenteres sammen med en analytisk model for at forudsige dens ydelse og flere metoder til at forbedre troværdigheden af de afkodede resultater.

Acknowledgements

At the end of this intense and fulfilling journey I would really like to thank all the people who supported and helped me during these three years. I will never forget the valuable lessons, the friendship, the patience, and the support they gave me. I would like to thank my supervisor Prof. Søren Forchhammer for the possibility of studying and working in his group as a Ph.D. student and the great freedom he offered me during my work. I would also like to thank Prof. Fernando Pereira for his support during my external stay in Instituto Superior Técnico: his great attention for details and his approach to research were highly stimulating for me. I would also like to thank the people I worked with during my external stay: Prof. João Ascenso and Catarina Brites for their patience and invaluable support.

I would like to thank the co-authors of my papers, in particular Marco Zamarin for his patience and the interesting discussions on Multiview Coding and Lars Lau Rakêt for his support for the generation of the Optical Flow-related results. They always tried to support me at the best of their capabilities and for this I am grateful. I would also like to thank other friends and co-workers for their friendship and support: Nino Burini, Anna Ukhanova, Federica Genovese, Jacob Søgaard, Ehsan Nadernejad, Giuseppe Toscano, Francesco Da Ros, Claire Mantel, Metodi Yankov and Jari Korhonen. Thanks to the Otto Mønsteds Fond and Oticon Fonden for supporting my participation to conferences and my external stay.

While in Lisbon I had the opportunity to meet some truly awesome people: Berner Panti, Luca Baroffio, Hoang Van Xiem, Pham Tiencuong, Vladimir Ivannikov, Diego Felix de Souza, Dhiraj Shah: thanks for some of the funniest lunches of my life.

Lastly I would like to thank my parents Carla Candeo and Otello Salmistraro, who, despite the distance, always supported and helped me to the best of their capabilities. For this I will be always grateful.

Matteo Salmistraro

Ph.D. Publications

This thesis is based on the following original peer-reviewed publications, which can be found in Appendix A:

PAPER 1 M. Salmistraro, M. Zamarin, S. Forchhammer, "Multi-hypothesis Distributed Stereo Video Coding", *Proc. of 2013 IEEE Int'l Work. on Multimedia Signal Processing (MMSP 2013)*, pp. 093 - 098, Pula, Italy, Sep. 30 - Oct. 2, 2013.

PAPER 2 M. Salmistraro, J. Ascenso, C. Brites, S. Forchhammer, "A Robust Fusion Method for Multiview Distributed Video Coding", *EURASIP Journal on Advances in Signal Processing*, (submitted).

PAPER 3 M. Salmistraro, L. L. Rakêt, Catarina Brites, J. Ascenso, S. Forchhammer, "Joint Disparity and Motion Estimation Using Optical Flow for Multiview Distributed Video Coding", *2014 European Signal Processing Conference (EUSIPCO 2014)* (submitted).

PAPER 4 M. Salmistraro, S. Forchhammer, "Low Delay Wyner-Ziv Coding Using Optical Flow", *2014 IEEE Int'l Conf. on Image Processing (ICIP 2014)*, (submitted).

PAPER 5 M. Zamarin, **M. Salmistraro**, S. Forchhammer, A. Ortega, "Edge-preserving Intra Depth Coding based on Context-coding

and H.264/AVC", *Proc. of 2013 IEEE Int'l Conf. on Multimedia and Expo (ICME 2013)*, pp. 1 - 6, San Jose, CA, USA, July 15-19, 2013.

PAPER 6 M. Salmistraro, L.L. Rakêt, M. Zamarin, A. Ukhanova, S. Forchhammer, "Texture Side Information Generation for Distributed Coding of Video-Plus-Depth", *Proc. of 2013 IEEE Int'l Conf. on Image Processing (ICIP 2013)*, pp. 1699 - 1703, Melbourne, Australia, Sep. 15-18, 2013.

PAPER 7 M. Salmistraro, M. Zamarin, L.L. Rakêt, S. Forchhammer, "Distributed Multi-Hypothesis Coding of Depth Maps using Texture Motion Information and Optical Flow", *Proc. of 2013 IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2013)*, pp. 1685 - 1689, Vancouver, Canada, May 26-31, 2013.

PAPER 8 S. Forchhammer, **M. Salmistraro**, K. J. Larsen, X. Huang, H. V. Luong, "Rate-adaptive BCH Coding for Slepian-Wolf Coding of Highly Correlated Sources", *Proc. of 2012 Data Compression Conference (DCC 2012)*, pp. 237 - 246, Snowbird, UT, USA, Apr. 10-12, 2012.

PAPER 9 M. Salmistraro, K. J. Larsen, S. Forchhammer; "Rate-adaptive BCH Codes for Distributed Source Coding", *EURASIP Journal on Advances in Signal Processing*, Vol. 2013, 166, 2013.

Other publications produced during the Ph.D. not included in this thesis:

[P10] L. L. Rakêt, J. Søgaard, **M. Salmistraro**, H. V. Luong, S. Forchhammer "Exploiting the Error-Correcting Capabilities of Low Density Parity Check Codes in Distributed Video Coding using Optical Flow", *Proc. of 2012 SPIE Optics+Photonics : Applications of Digital Image Processing XXXV*, San Diego, CA, USA, Aug. 12-16, 2012.

[P11] **M. Salmistraro**, S. Forchhammer "Stereo side information generation in low-delay distributed stereo video coding", *Proc. of 2012 SPIE Optics+Photonics : Applications of Digital Image Processing XXXV*, San Diego, CA, USA, Aug. 12-16, 2012.

[P12] **M. Salmistraro**, M. Zamarin, S. Forchhammer, "Wyner-Ziv Coding of Depth Maps Exploiting Color Motion Information", *Proc. of 2013 IS&T/SPIE Visual Information Processing and Communication Conf. (EI 2013)*, Burlingame, CA, USA, Feb. 3-7, 2013.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

In the past years many efforts have been devoted to the investigation of Distributed Source Coding (DSC) [1] and on the application of DSC principles to video coding: Distributed Video Coding (DVC) [2].

Predictive coding solutions, like H.264/AVC [3] or HEVC [4] are well established answers to the one-to-many coding problem, where the encoding of the video signal is performed only once and the decoding is performed many times by different decoders. In this scenario having a complex, expensive, energy-hungry encoder, and a low-complexity and cheap decoder is appealing for an extensive range of applications, for example broadcast, IPTV, etc. In predictive coding the encoder performs motion estimation and compensation, jointly coding the frames to exploit the temporal redundancy. This enables the codec to provide coding performance superior to intra coding (i.e. coding the frames independently) at the expenses of higher complexity at the encoder side.

The progress of technology has enabled, in recent years, the rise of a new kind of application: Wireless Sensor Network (WSN). In a WSN many small, battery powered and inexpensive sensors (or nodes) connected with wireless technologies, are expected to work cooperatively to provide services such as environmental control, health monitoring, detection of dangerous or toxic materials, etc. The nodes are usually connected to a base station, acting as a collection point of the data generated by the sensors.

Video Sensor Network (VSN) is a particular kind of WSN, where the sensors are equipped with a camera. WSNs are usually used in critical or hostile environments, and they are designed to work as long as possible without human intervention, therefore the nodes must be able to work for extensive periods of time without recharging the batteries. In a WSN wireless communication is the most relevant cause of energy consumption, therefore techniques and algorithms are used to reduce the need for communication to the minimum, following the belief that processing steps on the node require less energy than communication. According to this idea in a VSN efficient video coding algorithms should be used, in order to reduce the bits sent on the wireless medium. But predictive coding solutions have energy consumption comparable with transmission [5]. Therefore the energy gains achieved by sending less bits, employing a more efficient encoder, can be nullified if the higher efficiency comes at the cost of higher complexity. Furthermore the nodes usually lack the computational resources a normal computer, a laptop or even a smartphone have because of their reduced size. According to these considerations a low-complexity and efficient video coder is required to support video services over VSNs. On the other hand the decoder used by the base station is not constrained by complexity or power consumption issues. DSC principles enable a video encoder to code independently the frames, leaving to the decoder the task of exploiting the temporal redundancy, therefore DVC can be a viable solution for this problem [2,6].

Predictive coding solutions were extended to the multiview case [7,8], where new coding tools are proposed to leverage not only the temporal redundancy (or intra-view redundancy) but also the inter-view redundancy, due to the overlapping Field Of Views (FOVs) of the cameras. Here, the use of predictive coding solutions leads to a new significant problem: the need for inter-camera communication. Inter-camera communication is needed because joint *coding* needs to be done in order to perform inter-view coding: for example a Multiview Video Coding (MVC) coder [7] requires to have access to the frames coded by the other cameras to perform disparity estimation and compensation. The need for inter-camera communication increases the complexity and power consumption of the whole system. DSC here shows another advantage: the frames are encoded *independently*, and the decoder can exploit intra or inter-camera redundancy and no further actions are required to the cameras, which

can be, for what concerns coding, unaware of the presence of other nodes.

Summarizing DVC may enable the rise of a new kind of architectures, with low-complexity, independent encoders and a decoder able to jointly reconstruct frames exploiting intra and/or inter-camera redundancy [9].

Despite the great efforts of the community in the past years, the gap in performance between DVC-based coding solutions and predictive ones is still significant and new coding tools should be introduced in order to improve DVC performance.

1.2 Goals of the Thesis

The goals of this thesis are multiple in the context of DSC and DVC. In the context of Multiview Distributed Video Coding (M-DVC) novel tools for efficient inter-view Side Information (SI) generation are proposed, for the Stereo and Multiview scenarios. Attention is given to the robustness of the performance of the fusion between different SIs.

The use of advanced motion estimation algorithms, in particular Optical Flow (OF), is also explored, both in the context of M-DVC and for low-delay monoview DVC.

Encoding of depth maps, independently from texture frames, is also investigated: first with a traditional Intra encoder, then following distributed approaches. The Intra encoder leverages the peculiar features of the depth maps, such as wide smooth regions and sharp edges. The distributed approach exploits the correlation of the apparent motion of a depth stream with its corresponding texture stream.

Finally, motivated by the performance of Low Density Parity Check Accumulate (LDPCA) codes in DSC, Rate Adaptive (RA) BCH (Bose-Chaudhuri-Hocquenghem) codes are investigated in the high correlation scenario using short block lengths, as an alternative to the widely used LDPCA and Turbo codes.

1.3 Structure of the Thesis

The rest of the thesis is structured as follows. Chap. 2 introduces the seminal works on the field and the main tools employed in the presented works. Chap. 3 describes the main contributions of this thesis in the context of multiview and monoview DVC, highlighting their relevance

and novelty. Chap. 4 reports the novel tools introduced for efficient video-plus-depth coding. Chap. 5 describes the investigations in the field of RA Codes for efficient, feedback-based DSC. Chap. 6 summarizes the published material included in the thesis. Chapters from 3 to 6 are written in order to be self-contained as much as possible, therefore some overlaps exist between them. The interested reader can refer to the chapters from 3 to 5 for a high level perspective of the contributions of this thesis, or consult directly Chap. 6 for a brief description of the papers. Finally Chap. 7 summarizes the main results and outlines future development directions in the field of DVC and more in general DSC.

Chapter 2

Background

This section outlines the evolution of the research in Distributed Source Coding (DSC) and Distributed Video Coding (DVC). First the theoretical foundations of the field are outlined, then the first practical video codec using DVC principles are briefly described, finally the latest contributions to the field are presented. In the last part of this section a brief description of the current research trends and possible future developments is reported.

2.1 Theoretical Foundations

The seminal works in DSC are the Slepian-Wolf [10] and Wyner-Ziv [11] theorems, both dating back to the 70s of the past century. They prove that it is possible to achieve the same performance of a joint encoding and decoding with a disjoint encoding and a joint decoding.

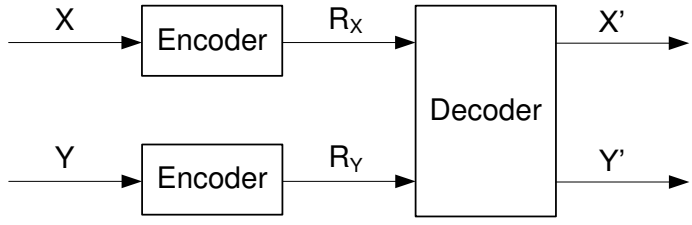


Figure 2.1: Disjoint encoding and joint decoding, following the Slepian-Wolf Theorem.

The Slepian-Wolf theorem [10] addresses lossless DSC. Two correlated sources X and Y, see in Fig. 2.1, are independently encoded but jointly decoded. X and Y are assumed to be correlated, i.i.d., finite alphabet random sequences. To code them the encoders generate two bitstreams, with rates R_X and R_Y respectively. Assuming disjoint coding and decoding the rates are $R_X \geq H(X)$ and $R_Y \geq H(Y)$, where $H(X)$ and $H(Y)$ are the entropies of X and Y. On the other hand, jointly coding X and Y requires a rate $R \geq H(X,Y)$, and $H(X) + H(Y) \geq H(X,Y)$. The Slepian-Wolf theorem establishes that if the rates R_X and R_Y are chosen inside the achievable rate region, the sources X and Y can be jointly reconstructed with a vanishing error probability. The achievable rate region is depicted in Fig. 2.2 and it can be analytically described as:

$$R_X + R_Y \geq H(X,Y) \tag{2.1}$$

$$R_X \geq H(X|Y) \tag{2.2}$$

$$R_Y \geq H(Y|X). \tag{2.3}$$

where $H(X|Y)$ is the entropy of X conditioned to Y. Analyzing the achievable rate region, it can be noted that it is possible to achieve the same performance of a joint encoding and decoding with a disjoint encoding and joint decoding.

The Wyner-Ziv theorem [11] addresses the problem of lossy compression of a source X with the availability of a correlated source Y only at the decoder, see Fig. 2.3. X and Y are samples from i.i.d. random sources of infinite alphabet. Y is referred to as Side Information (SI). The rate $R^*(D)$ is used to encode the source X, and the distortion introduced by the lossy coding is defined as $D = E[d(X, \hat{X})]$. Let $R_{X|Y}(D)$ denote the rate to lossy code the source X having Y available both at the encoder and at the decoder. The result shows that $R^*(D) \geq R_{X|Y}(D)$. More interestingly it is also shown that $R^*(D) = R_{X|Y}(D)$ in the case of Gaussian memoryless sources using as distortion measure the mean square error.

The two aforementioned results [10, 11] demonstrate that disjoint encoding and joint decoding can reach the same performance of conventional joint coding and decoding, but they do not provide a viable way to apply such results to realistic scenarios. Already in [12] it was proposed that channel coding may be used to create practical Slepian-Wolf coding

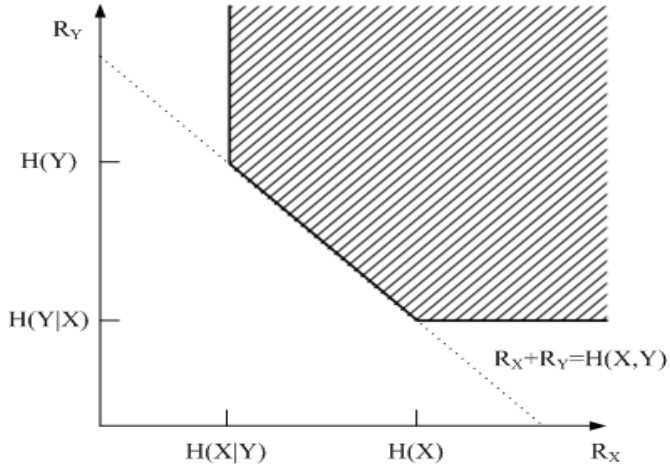


Figure 2.2: Achievable rate region (shaded area) according to the Slepian-Wolf Theorem.

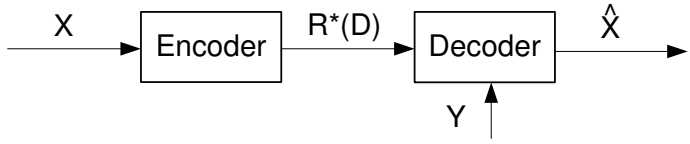


Figure 2.3: Wyner-Ziv Theorem.

solutions: Y can be seen as a corrupted version of X and error correcting codes can be used to correct the differences between the two sources. It has to be noted that the errors in Y are not due to transmission errors but they are introduced by a "virtual correlation channel" capturing the correlation between the two sources [2]. For what concerns a practical Wyner-Ziv coding solution it may be achieved by concatenating a quantizer with a Slepian-Wolf encoder [2], see Fig. 2.4.

This brief discussion does not solve the problem of applying the Wyner-Ziv principles to video coding, many questions remain: e.g. which error correcting codes have to be used for the Slepian-Wolf coder, how to control their rates, how to generate the SI.

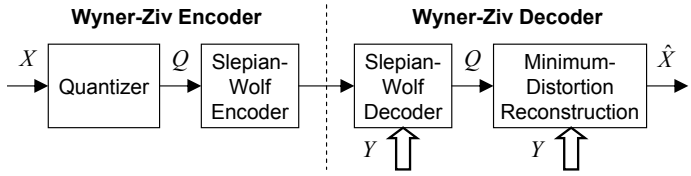


Figure 2.4: An example of a practical Wyner-Ziv Codec, from [2]. Copyright IEEE 2005.

2.2 First Practical Codecs for DVC

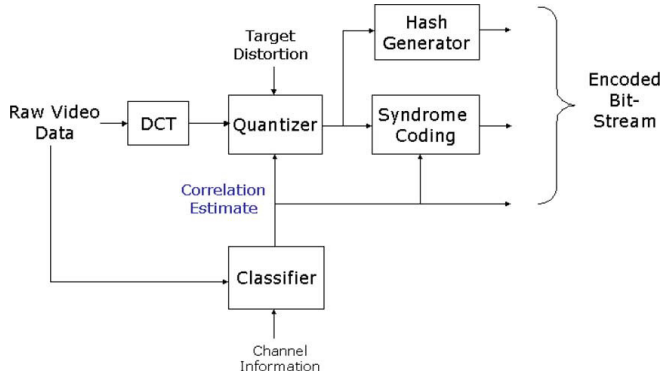
The first practical DVC codec solutions were presented in [13, 14]. The first architecture, called PRISM (Power-efficient, Robust, hIgh-compression, Syndrome-based Multimedia coding), depicted in Fig. 2.5, has been presented, for the first time in [13] and extensively described in [15].

The encoding process, for each 8×8 block of the frame can be summarized as:

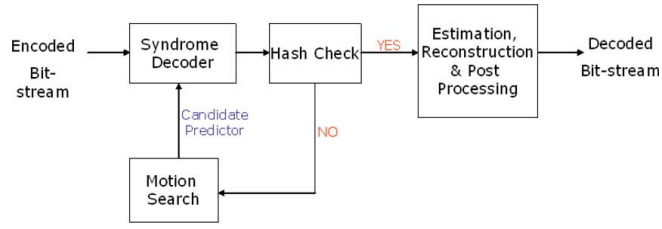
- Each block belonging to the raw video data is first DCT transformed and quantized employing a scalar quantizer;
- The *classifier* estimates the correlation between the SI and the original frame for each DCT block;
- For each DCT block the encoder can choose, according to the classification, to skip, intra code or WZ code the bitplanes. For what concerns the WZ coding part, BCH (Bose-Chaudhuri-Hocquenghem) codes [15] are used and their rate is chosen according to outcome of the previous classification step;
- A hash signature is also calculated and sent for every block, to help the decoder performing motion estimation. In [15] the signature is a Cyclic Redundancy Check (CRC) checksum.

The decoding process, for each 8×8 block can be summarized as:

- The motion search module generates a set of possible candidates (SIs), like the motion search of a normal predictive coding encoder would do, but the search is performed at the decoder;



(a)



(b)

Figure 2.5: PRISM Encoder (a) and Decoder (b), from [15]. Copyright IEEE 2007.

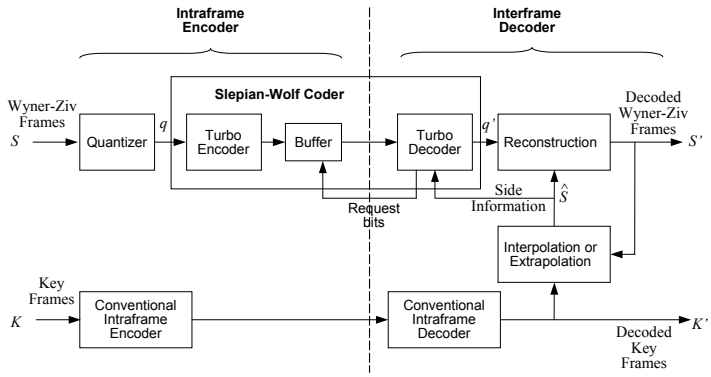


Figure 2.6: The Stanford Codec Architecture, pixel-domain encoding, from [2]. Copyright IEEE 2005.

- Each SI is used together with the encoded bitstream to decode the block, each result is checked with the help of the hash signature. If the decoded sequence satisfies the received hash, the solution (i.e. the sequence of recovered quantized coefficients) is taken as the successfully decoded one;

- The recovered quantized coefficients are used to reconstruct the original DCT coefficients. The reconstructed DCT coefficients are then inverse DCT transformed.

The second approach, usually referred to as Stanford codec [2, 14] is built over a frame-based approach rather than on a block-based one. The architecture is outlined in Fig. 2.6. The encoder presented in [2] codes the pixels: this kind of codecs are referred to as pixel-domain architectures. The transform-domain architectures, on the other hand, code data in the transform domain, a common case being coding DCT coefficients.

Each frame can be a Key Frame (KF) or a Wyner-Ziv (WZ) frame. Without loss of generality the Group Of Pictures (GOP) of size 2 can be considered as an example, where the frames are coded following the structure KF-WZ-KF. KFs are intra coded and decoded, independently with respect to each other and with respect to the WZ frames. For what concerns the encoding process of the WZ frames, the steps can be summarized as follows:

- The pixels are quantized and the obtained quantization indexes are provided to the Turbo encoder;
- The turbo encoder calculates the parity bits, which are stored in a buffer, ready to be delivered upon request.

The decoder performs the most complex tasks and its goal is the reconstruction of the WZ frames:

- Using the previously decoded frames (either KFs or WZ frames) the SI is generated. The SI is an estimation of the WZ frame;
- The Turbo decoder combines the SI and the parity bits from the encoder to recover the quantization indexes;
- If the Turbo decoder is unable to decode successfully, new parity bits are requested via a feedback channel;
- Using the decoded quantized coefficients the frame is reconstructed according to the estimated mean-squared-error.

From this brief description some key differences between the two approaches emerge: first the Stanford codec is frame-based, while PRISM is block-based, therefore PRISM may be able to adapt faster to different characteristics of different zones of the frame. The need for a feedback channel of the Stanford approach enables a very low complexity encoding, but, on the other hand, makes the approach not suitable for applications where a feedback channel is unavailable. PRISM does not require a feedback channel, but the encoder is likely to be more complex than the Stanford encoder. Lastly, the Stanford architecture uses sophisticated error correcting codes, like Turbo codes or Low Density Parity Check Accumulate (LDPCA) [16] codes, while PRISM employs much simpler codes, e.g. BCH codes. For a more detailed review of the two codecs the reader is referred to [17].

2.2.1 Error Correcting codes for DSC

From Fig. 2.6 it is possible to distinguish two different channels, the transmission channel and the feedback channel. The transmission channel is used to transfer the parity bits from encoder to decoder. The

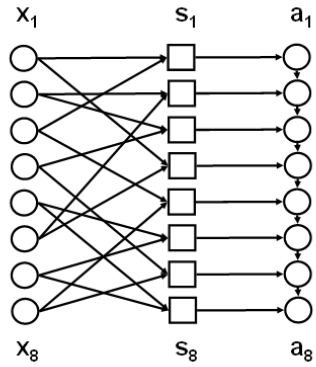


Figure 2.7: The LDPCA Encoder, from [16]. Copyright Elsevier 2006.

feedback channel, if available, is used by the decoder to request parity bits to the encoder. Usually transmission and feedback channels are supposed to be error free when assessing the Rate-Distortion (RD) performance of DVC codecs. The error correcting codes are used to correct the errors in the SI. To model the correlation between the source X and the SI Y the already mentioned concept of "virtual channel" is used. The virtual channel is therefore afflicted by noise.

The first Stanford DVC codec employed Rate Compatible Punctured Turbo (RCPT) codes [14]. RCPT codes were previously studied in the context of communication, for handling varying communication channel statistics [18]. LDPCA codes were proposed later [16]. Punctured Low Density Parity Check (LDPC) codes were first proposed for feedback-based Rate Adaptive (RA) DSC coding, but they perform poorly, due to the high level of degradation of the decoding graphs. The LDPCA encoder is the concatenation of a LDPC encoder with an accumulator, see Fig. 2.7. The accumulator sums, modulo 2, the calculated syndromes, which are stored in a buffer, waiting for request.

In Fig. 2.8 the decoding graphs of three significant cases are provided: transmission of all the syndrome bits, transmission of half of the *accumulated* syndrome bits and transmission of half of the syndrome bits (punctured LDPC code). The motivation of the superiority of the performance of LDPCA codes when compared with punctured LDPC

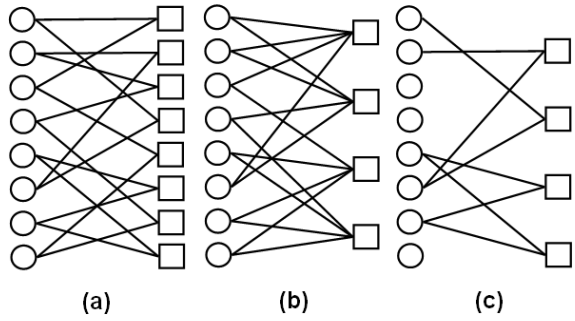


Figure 2.8: Decoding graphs if the encoder transmits (a) the entire accumulated syndrome, (b) half of the *accumulated* syndrome bits, (c) half of the syndrome bits. From [16]. Copyright Elsevier 2006.

codes is apparent when comparing Fig. 2.8(b) with Fig. 2.8(c). When using LDPCA codes the degree of the source nodes is the same between a compression ratio 2:1 (Fig. 2.8(b)) and 1:1 (Fig. 2.8(a)). On the other hand, the decoding graph for the punctured LDPC code at compression ration 2:1 is severely degraded.

In a LDPCA decoder iterative decoding can be applied to the graph, for every given compression ratio. To test the correctness of the decoded result, syndrome bits can be calculated from the decoded sequence and checked against the received bits.

LDPCA codes are widely used in DVC [19] because of their superior RD performance when compared with Turbo codes.

As a final remark, it has to be noted that one of the main arguments for the adoption of DVC is its error resilience due to the use of channel codes for encoding. Therefore some studies have assessed the performance of DVC codecs in case of noisy transmission channel, e.g. [P10], [15,20,21].

2.3 Developments in Monoview DVC

After the seminal works in DVC [13, 14] many investigations have been performed in the field of DVC, most notably the European DIStributed COding for Video sERvices (DISCOVER) Project [22] delivered many

advances in the field. During the 27 months project, six universities combined their efforts to produce new tools and techniques for DVC. The contributions addressed a wide range of issue, for example rate control [23, 24], optimal reconstruction [25], SI generation [26] and modeling of the virtual channel [27]. For a full list of the contributions of the project partners, one can refer to [28]. The DISCOVER codec [19] is one of the most relevant contributions of the project: it is a transform-domain, monoview DVC codec, based on the Stanford architecture, and it is still used as benchmark for DVC coding solutions. The main improvements of DISCOVER over the first Stanford codec are [19]:

- SI generation: Motion Compensated Interpolation (MCI) is used to generate the SI, estimating the motion between a backward and a forward decoded frame. The system is block-based, it uses 16×16 pixels or 8×8 pixels blocks, and assumes linear motion between the frames. Outliers in the motion field are removed with a median filter;

- Noise modeling: the noise on the virtual channel is modeled according to a Laplacian distribution, the parameters of the distribution are estimated on-line, i.e. without requiring any kind of information regarding the original WZ frame. The residual can be also calculated as the difference between the SI and the original frame and it is referred to as off-line residual. It is obvious that the on-line residual is the one enabling the creation of a realistic architecture. The virtual channel models the correlation between corresponding DCT bands in the SI and WZ frame. The distribution is then used (together with previously decoded bitplanes) to calculate the conditional bit probabilities required by the channel decoder;

- Channel decoder: LDPCA codes are used. To check the correctness of the decoded result first it is checked if the syndromes calculated on the decoded bitplane are equal to the received ones. If this first check is successful, a 8-bit CRC check is used to further verify the result.

After the publication of the DISCOVER codec many other improvements have been proposed. Many of the contributions of this thesis use

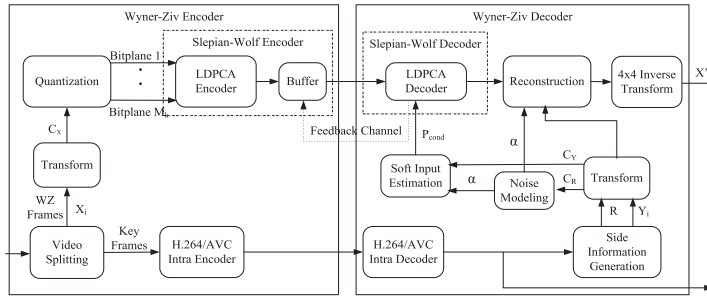


Figure 2.9: The enhanced Transform Domain WZ codec at the basis of some of the contributions of this thesis, from [29]. Copyright Elsevier 2012.

as basis the improved decoder presented in [29]. Therefore it is useful to briefly describe it. Its architecture is presented in Fig. 2.9.

The main contributions of the codec are:

- Advanced SI generation module: Overlapped Block Motion Compensation (OBMC) [30] uses the luminance and chrominance components for enhanced motion estimation between the available decoded frames. The motion estimation is performed using a variable block size, more specifically the default block size is 8×8 pixels, but blocks being identified as unreliable are further divided into four 4×4 blocks for increased precision. Lastly for each SI block the Motion Vectors (MVs) of neighboring blocks are used to generate more candidates for the block. The candidates are averaged using weights depending on the Mean Squared Error (MSE) of the match.

- Cross-band noise model: one of the main problems in DVC is the modeling of the noise on the virtual channel. The noise model parameters are usually calculated from the estimated residual and it is subject to errors, which can be mitigated during the decoding process, following a learning approach. The decoding process follows a zig-zag scan order, starting with the DC band. After decoding the band b_k, new information is available, which can be used to enhance the noise model of the next bands b_l, $l > k$. The difference between the original SI frame and the partially decoded frame is used to segment the coefficients in the next band in two

sets: outliers and inliers. The idea is that if a coefficient in band b_k is wrong, probably the corresponding coefficient in band b_l is wrong too, therefore it is treated as outliers and this is taken into consideration during the calculation of the noise model for the band b_l. For more information the interested reader is referred to [29].

- Residue refinement: the cross-band noise model helps improving the coding performance for each new band. But the DVC codec has a even finer level of refinement: the bitplane. The residue refinement uses the already decoded bitplanes to improve the noise modeling of the band being decoded.

2.3.1 Multiple Side Information Decoding

The codec in [29], and many others proposed in literature, use only one SI generation system. But the various methods proposed in literature so far differ greatly in performance, and it is difficult to find a method able to outperform all the others in all the cases. Given that the SI is generated at the decoder, it is possible to generate more than one SI, without increasing the complexity of the encoder. Different SI generation methods have different performance in different areas of a frame, the decoder can try to estimate the reliability of each candidate and then choose the one having higher (estimated) reliability on a pixel-by-pixel or block-by-block basis. This approach was used, for example, in [31] where a global and a local motion estimations are performed, leading to the generation of two SIs. The candidates are then combined, generating a new fused SI, which is then used as basis for the decoding process. Another approach to this problem has been proposed in [32]: where a Multi-Hypothesis (MH) decoder is used to fuse an OBMC with a SI generated using Optical Flow (OF). In the MH decoder (depicted in Fig. 2.10) the M different SIs, Y_i, $i = 1, \ldots, M$, are generated independently. For each SI the distribution $f_{X|Y_i}$ is calculated, where X is the original WZ frame, and $f_{X|Y_i}$ is the estimated distribution of the to-be-decoded DCT coefficient given Y_i. The j-th fused distribution F_j can be calculated as:

$$F_j = \sum_{i=1}^{M} w_{ji} f_{X|Y_i}, \tag{2.4}$$

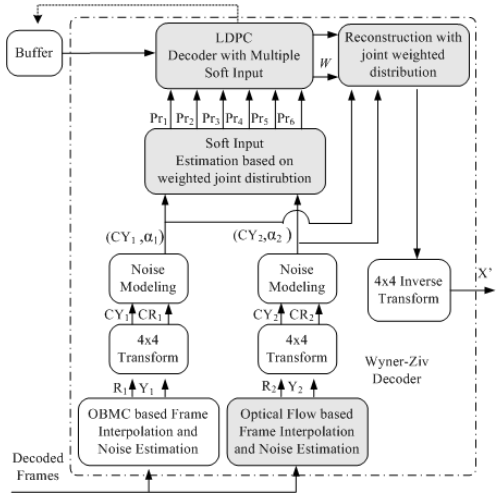


Figure 2.10: The MH decoder using two SIs: OBMC and OF. From [32]. Copyright IEEE 2011.

where $j \in [1, \ldots, N]$. Now the problem seems to be even more complex: before there were M SIs, now N distributions are available: in [32] $M = 2$ or $M = 3$, while $N = 6$. The solution to this problem is the use of parallel LDPCA decoders. In Fig. 2.11 $N = 2$ parallel decoders are depicted for simplicity. p_1 and p_2 are the conditional probabilities for a given bitplane, calculated with the fused distributions F_1 and F_2 respectively. The LDPCA decoders receive the same syndromes from the encoder but use different conditional probabilities. The decoders try to decode, if at least one of the N succeeds, and no decoding errors are detected by the CRC check, the convergence is declared, otherwise a new set of parity bits is requested. Assuming that the decoder employing p_z achieved a successful decoding, F_z is employed in the reconstruction step and the decoded bitplane, b_z, along with the previous ones, is used to determine the interval $[L, U)$ in which the reconstructed coefficient x' lies. For the reconstruction of the coefficient x' the optimal reconstruction proposed in [25] is used:

$$x' = \frac{\int_L^U x F_z(x) dx}{\int_L^U F_z(x) dx}. \tag{2.5}$$

The MH decoder has been a key element of the contributions of this thesis, and it was able to deliver consistent and stable improvements

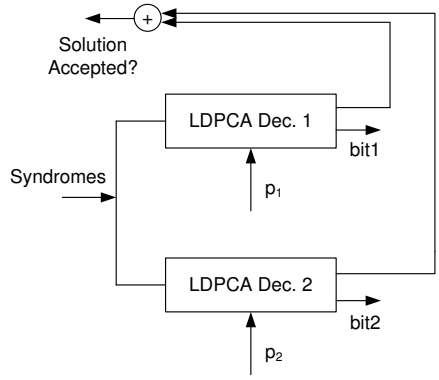


Figure 2.11: Parallel LDPCA decoder for MH decoding, $N = 2$.

over single SI decoders or other kind of fusion techniques. The main drawback of the system is the uses of N LDPCA decoders, which can increase the decoding complexity up to N times when compared with a single SI decoder. Nevertheless, in DVC, the decoding complexity is a minor concern.

2.3.2 Motion Estimation

In [32] OF is used as one of the possible SI candidates. OF-derived techniques are another element at the basis of the contributions of this work, therefore the concept is introduced here. OF is a motion estimation technique used for calculating the displacement field v between two decoded frames $\tilde{I}_0$ and $\tilde{I}_1$. The key difference between OF and block-based techniques, such as OBMC is that OF is *dense*, i.e. a MV is calculated for each position x in the frame being the source of the flow. In general the motion field v is calculated minimizing the constraint $C(x, v)$:

$$C(x, v) = \tilde{I}_0(x) - \tilde{I}_1(x + v(x)). \tag{2.6}$$

In general, in order to make the problem well posed, the irregular behavior of v has to be penalized. In the presented works the TV-L^1 energy $E(v)$ [33] is minimized:

$$E(v) = \int \lambda ||C(x, v)|| dx + \int ||\mathcal{D}v(x)|| dx. \tag{2.7}$$

λ is a weighting factor used to determine the trade-off between an high degree of similarity of the matched pixels (i.e. small $C(x, v)$) and the regularity of the flow. In all the works reported in the thesis the global regularization term $\mathcal{D}v(x)$ is the 1-Jacobian [33]. This description of OF is provided only as reference, for more information on how OF is used in the context of DVC the interested reader is referred to the next section and the corresponding papers, where the contributions related to OF are described in detail. The use of dense motion or disparity estimation techniques were also proposed in the case of predictive video coding, e.g. [34]. The use of OF is challenging, in predictive coding, because the dense motion field must be efficiently encoded. DVC, on the other hand, does not require to code the motion field, because it is generated at the decoder.

OBMC or the OF-based techniques used in e.g. [32] belongs to the area of Interpolation-based SI generation techniques. These techniques interpolates an unknown WZ frame exploiting a preceding frame and a following frame under the linear motion assumption. While these SI generation techniques lead to relatively good RD performance, they increase the delay, because the frames are not coded in the order they are captured. Low-delay DVC tries to solve the problem using, for SI generation purposes, only preceding frames, employing Extrapolation-based techniques, e.g. [35]. Such techniques solve the problem of the delay but at the expenses of lower RD performance when compared to Interpolation-based techniques.

2.3.3 Other Relevant Contributions

The aforementioned works are described in detail to allow the reader to familiarize with the tools and benchmarks used primarily in the papers supporting this thesis. Nevertheless, for completeness, other research trends and relevant works in DVC need to be briefly discussed.

Stanford-based DVC coding architectures require the use of a feedback channel. This may be undesirable or unfeasible for some applications, therefore the community developed tools to avoid the use of feedback channel, e.g. [24,36]. These solutions, while removing the need for a feedback channel, increase the complexity of the encoder and their performance are inferior when compared with codecs using feedback.

Hybrid approaches, putting constraints on the number of requests an encoder can do, were also proposed, e.g. in [37]. In DVC, skip or intra modes were introduced, to improve coding efficiency [38]. Learning refers to the use of previously decoded data (e.g. bitplanes, DCT band, frames) to improve the current decoding, e.g. in [39] after decoding each DCT band the partially decoded frame is used to identify unreliable parts of the SI. For these parts the motion is calculated again with the help of the already decoded information. The new motion is used to refine the SI and update the residual estimation used for the decoding of the next band, leading to an overall improved coding performance.

Another approach investigated in literature consists in allowing the encoder to send additional information to the decoder, to improve the coding performance. One example being sending descriptors to allow the decoder to generate a SI based on global motion estimation [31]. These approaches help improving the accuracy of the SI, but they also increase the complexity of the encoder and this may be undesirable for some applications. Lastly, while not related to the works presented in this thesis, it is important to notice that the use of DVC-derived approaches or tools were proposed to improve the performance of predictive coding solutions. One of the first examples being Systematic Lossy Error Protection (SLEP) [2], where the video signal is coded using an ordinary predictive coder. The signal is then transmitted over a lossy channel along with parity bits generated by a Wyner-Ziv encoder. The Wyner-Ziv encoder encodes a coarsely quantized version of the frame. In case of errors the error concealed frame is used as SI of a Wyner-Ziv decoder, which employs parity bits to correct the errors made by the error concealment. This allows a graceful degradation of the video quality in case of errors. More recently, in [40], the SI generation system used in monoview DVC is adapted to enhance the coding performance of the DIRECT mode, which can be selected in B-slices in H.264/AVC. As opposed to DVC the SI is calculated at the encoder and at the decoder, but only the reference frames are needed for the estimation, therefore no additional rate has to be used to send the MVs, because the same SI can be calculated independently by both encoder and decoder.

2.4 Developments in Multiview DVC

DVC, as previously pointed out, may allow a network of cameras to perform joint decoding of the frames without inter-camera communication. Even though this thesis focuses on Stanford-based DVC coding solutions, it is relevant to acknowledge the existence of multiview PRISM-based coding solutions, e.g. [41]. The solution uses the same, basic concept of PRISM: at the decoder, for every 8×8 block a list of predictors (SIs) is generated. For every element of the list, syndrome decoding is performed and the result undergoes CRC check. If the CRC check is successful, the block is declared successfully decoded, and reconstructed. The set of candidates is generated using a view-synthesis-based correlation model or a disparity-based correlation model. When using a view-synthesis-based correlation model three views are needed. The lateral views (left and right views) are available at the decoder, the central view is coded using DVC principles. The SI for the central view is synthesized using the lateral views [42]. The candidates are the blocks contained in a small area near the block corresponding to the to-be-decoded one. Allowing more candidates enables the system to cope with errors in the synthesized view. When using a disparity-based correlation model only one lateral camera is needed (either left or right). The method leverages the epipolar constraint: given a point in one view, the corresponding one may be found along the epipolar line in the second view, if not occluded. Therefore, given a block in the DVC-coded view, the candidates are the blocks along the epipolar line in the lateral view.

For what concerns the Stanford-based approach, it has been applied to stereo and multiview images in [43–45] where unsupervised disparity learning is performed during the decoding procedure. The decoder uses one (stereo) or two (multiview) reference images available at the decoder to generate the SI through disparity estimation. The disparity is refined along the decoding process performed by an LDPCA decoder.

For what concerns distributed coding of multiview video, the project DISCOVER contributed to the development of Multiview Distributed Video Coding (M-DVC) as well, e.g. in [46, 47], even though the focus of the DISCOVER codec is on monoview coding [19]. The majority of the contributions in M-DVC are based on a fusion approach [46, 48–50], where an inter-view SI and an intra-view SI are fused. The intra-view SI

is usually generated by means of interpolation-based techniques, which leverage temporal redundancy. The inter-view SI is generated exploited the redundancy between different views of the same scene. In order to generate the inter-view SI the other views must be available at the decoder, e.g. in [46] left and right cameras are intra-coded and only the central one is DVC encoded. The two SIs (inter and intra-view) perform differently in different areas of the image: while the temporal SI is not accurate when fast motion occurs, the inter-view SI performs poorly in case of occluded objects. The main objective of a fusion technique is to combine the correctly estimated parts of the two SIs, discarding, in the meantime, the erroneous parts. In order to perform this task great efforts have been dedicated to find reliable ways of estimating the quality of the areas of the SI and then robustly fusing them. The encoder can be involved in the SI fusion process, providing information to the decoder to help the estimation of the quality of the SI, e.g. in [46]. This approach may enable better performance at the expense of higher computational burden on the encoder, because the additional information has to be calculated and coded, and both tasks may be expensive. Furthermore the rate needed to transmit the additional coded information may nullify the RD improvement from the enhanced fusion.

Other approaches do not involve the encoder in the fusion process and try to estimate as precisely as possible the quality of each pixel of the SIs using features e.g. the estimated residual [49]. More advanced methods use machine learning techniques, e.g. [50], to robustly estimate the fused SI. All the efforts in the field aim to obtain performance comparable to the ideal fusion or oracle fusion. Ideal fusion provides an upper bound to the performance of fusion techniques. Like off-line residual estimation it requires the knowledge of the WZ frame at the decoder therefore it is not a practical solution for DVC, but it gives insights on the achievable performance. Denoting the two SIs, which need to be fused, as Y_1 and Y_2, denoting the resulting fused SI as Y_{IF} and denoting as the original WZ frame X, for each point (x, y) in Y_{IF}, the fusion is performed as:

$$Y_{IF}(x,y) = \begin{cases} Y_1(x,y) & \text{if } |Y_1(x,y) - X(x,y)| < |Y_2(x,y) - X(x,y)| \\ Y_2(x,y) & \text{otherwise.} \end{cases} \tag{2.8}$$

Despite its wide use in literature, it has to be noted that there is no

guarantee that the RD performance obtained using Y_{IF} as SI is the upper bound of the RD performance. Nevertheless, the gap between ideal fusion and practical fusion schemes is still significant.

A different approach was proposed in [51], where first, either an inter-view or an intra-view SI is used for fully decode the WZ frame. The choice on which SI is used is made according to the estimated motion activity of the central view. Secondly the decoded WZ frame is used to generate a refined SI, which is used in a second reconstruction round improving the reconstruction quality of the decoded WZ frame. The refined SI is generated predicting the decoded WZ frame with left, right, previous and following frames. This step generates four candidate blocks, and the best matching block is chosen according to the Sum of Absolute Differences (SAD). The best matching block is then used as SI block in the refined SI.

The idea of joint decoding without inter-camera communication of multiview video streams has been investigated using other approaches e.g in [52], where two cameras independently (intra) code the video stream. The decoder first reconstructs independently the two frames, then a joint reconstruction is applied as a post-processing step to increase the quality of the decoded frames. This approach does not require a feedback channel.

These works provide insights on how, using DVC-based techniques, it is possible to jointly decode a multiview video stream without inter-camera communication. This is of great interest in the field of Video Sensor Networks (VSNs), and may enable a broad range of future services in the field of e.g. video surveillance. Many problems have still to be addressed: it is still difficult to generate good quality inter-view SI, and despite the improvements in fusion techniques, robustness problems are still present [49].

2.5 Video-plus-depth Coding

Depth maps provide information on the geometry of the scene, an example of depth map with the corresponding texture frame (i.e. the luminance and chrominance information of the scene), is provided in Fig. 2.12. Depth maps are gray-scale images providing an indication of the distance between objects in the scene and the camera, for each pixel.

(a) (b)

Figure 2.12: Texture (a) and Depth (b) frames of the *Ballet* [53] sequence. The convention for the depth map is that the closer the object the brighter the corresponding pixels.

A relevant use of depth maps is for Depth-Image-Based-Rendering (DIBR) [54]. If the decoder has access to two or more views and their depth information is also available, any intermediate position between the available cameras can be estimated with DIBR. This is of interest in Free Viewpoint Video (FVV) where the user can freely navigate the view, watching the scene from a virtual camera, rendered using the other views. Without DIBR applications like FVV would require a much higher number of coded views and it would not be possible to reach the same level of flexibility. Depth maps are also useful tools in monoview scenarios, e.g. in video surveillance, because they can be used for segmentation, scene matting and motion detection [55]. Regardless to the purpose of the use of depth maps, an open issue remains: how to efficiently code the depth information. State-of-the-art codecs can be used for this purpose, but many other tools, exploiting the particular features of depth maps, have been proposed. Depth maps are characterized by smooth regions divided by sharp edges, which set them apart form natural images. The edges play an important role in the view synthesis process and their preservation during the coding procedure is of paramount importance, therefore many edge-aware coding techniques have been proposed during the years, e.g. [56–58].

Texture and corresponding depth maps show similar motion activity, therefore motion vector sharing techniques were also proposed: the motion information of the texture frames is used to motion compensate the

depth stream, as done e.g. in [59], leading to rate gains because the MVs for the depth maps are not coded. This feature was exploited in a DVC scenario, in [60] where decoded frames belonging to the texture stream are used to generate accurate SI for decoding a WZ encoded depth map. The use of DVC tools in video-plus-depth is of interest in case of separate depth and texture cameras or when, due to complexity constraints, joint decoding is preferable to joint encoding.

Chapter 3

Novel Tools for Distributed Video Coding

This section describes various contributions made in the field of Distributed Video Coding (DVC). In the context of Stereo and Multiview DVC two problems are addressed: inter-view Side Information (SI) generation and fusion of inter-view and temporal SIs.

In Stereo DVC, for what concerns inter-view SI generation, previous works addressed the problem with disparity-based frame estimation [48] or disparity-guided temporal interpolation [61]. Both works proposed also realistic fusion techniques, in particular in [48] a delayed or motion compensated mask are used, but they are unable to provide Rate-Distortion (RD) performance gains. Nevertheless [48] shows that an ideal fusion between temporal and inter-view SIs leads to significant gains over a purely temporal SI. [61] proposed a multi-hypothesis based noise modeling system for fusion purposes. A practical stereo DVC codec, presented in **PAPER 1**, is here described. To provide better SI quality the motion of the other view is employed. At the same time, a more robust fusion technique, based on the Multi-Hypothesis (MH) decoder provides stable RD gains.

In Multiview Distributed Video Coding (M-DVC) many works have addressed the fusion problem, e.g. [49, 50]. The fusion problem is particularly hard when the SIs have very different RD performance [62]. Secondly generating inter-view SI for large disparities prove to be challenging. A novel inter-view SI generation method and a novel fusion

technique for M-DVC are investigated in **PAPER 2**. More specifically, the idea of learning outlined in Chap. 2 is here used for improved SI fusion. The problem of the generation of the noise model for the fused SI is also addressed, relying on the concept of fusion of the distributions.

Instead of calculating a temporal SI and an inter-view SI and then perform fusion, in [47] was proposed to predict the motion of the central view from the motion of the left and right views with MultiView Motion Estimation (MVME) SI generation. MVME employs a block-based approach. Motivated by the superior performance of techniques based on Optical Flow (OF) over conventional block-based ones, in **PAPER 3** a conceptually similar method to MVME is proposed. OF techniques for M-DVC SI generation are introduced, and compared with similar block-based approaches.

Extrapolation [35] is a SI generation technique for monoview DVC. It is of interest for low-delay DVC but suffers from lower performance when compared with Interpolation-based methods. To improve Extrapolation learning is used in [35]: the partially decoded Wyner-Ziv (WZ) frame is used to generate a more precise SI. The Lukas-Kanade algorithm was used in [63] to provide a better SI in a low-delay DVC decoder. Motivated by these previous work, in **PAPER 4** it is proposed a low-delay, monoview DVC decoder, employing a novel SI generation method and SI refinement, both based on OF.

In general, the performance of the proposed tools is compared with the single SI Overlapped Block Motion Compensation (OBMC)-based decoder proposed in [29], which outperforms the DISCOVER codec and is the basis of the proposed coding architectures.

3.1 Stereo Distributed Video Coding

A stereo stream is constituted by two different views of the same scene. The disparity between them allows stereoscopic displays to provide a depth illusion to the user. In **PAPER 1** a stereo DVC codec is presented. Two different SI generation systems are investigated as possible candidates for inter-view SI generation: Motion Vector Similarity (MVSim) and Difference Projection (DP). In both cases the SIs are generated leveraging the motion of one view (*support* view) to predict the WZ frame in the other view (*master* view).

MVSim calculates the motion field between two consecutive frames in the support view, $I_{s,t-1}$ and $I_{s,t}$ having time indexes $t-1$ and t respectively. Then, disparity estimation is applied between $I_{s,t-1}$ and its temporally corresponding frame in the master view $I_{m,t-1}$. The disparity field is then used to warp the motion field in the support view towards the master view. Finally the disparity compensated motion field can be used to motion compensate $I_{m,t-1}$ obtaining the SI.

The second method, DP disparity compensate the difference between two consecutive frames in the support view $I_{s,t-1}$ and $I_{s,t}$. The disparity compensation is applied only to the pixels showing enough motion activity to justify the risk of introducing artifacts.

The inter-view SI (either DP or MVSim) is fused with OBMC with a MH decoder [32], achieving higher gains and stability when compared with other fusion methods. Competing fusion methods were, in some cases, unable to outperform the single SI OBMC-based decoder. DP showed superior performance when compared with MVSim, allowing the MH decoder to outperform the single SI, OBMC-based decoder by up to 0.8 dB, measured in Bjøntegaard Peak Signal-to-Noise Ratio (PSNR) distance [64] on the WZ frames, for Group Of Pictures (GOP) 2, or equivalently by up to 0.33 dB for the performance on all the frames.

3.2 Multiview Distributed Video Coding

This section reports the findings of **PAPER 2** and **PAPER 3**. The addressed scenario is the multiview one, depicted in Fig. 3.1, where the lateral views (left and right) are available at the decoder and the central view is WZ encoded. The lateral views are usually H.264/AVC Intra coded.

In **PAPER 2** the problem of unknown and high distance between lateral cameras and the problem of robust fusion are addressed. In multiview SI generation the closest lateral views are usually used [50]. Disparity Compensated View Prediction (DCVP) [46] is widely used for inter-view SI generation. The paper shows that it is difficult, for DCVP, to find a set of parameters (e.g. search range) able to generate good quality SI for different configurations of the cameras (i.e. different inter-camera distances). To overcome the problem, a sliding window approach is proposed, which is able to remove the lateral part of one view out of

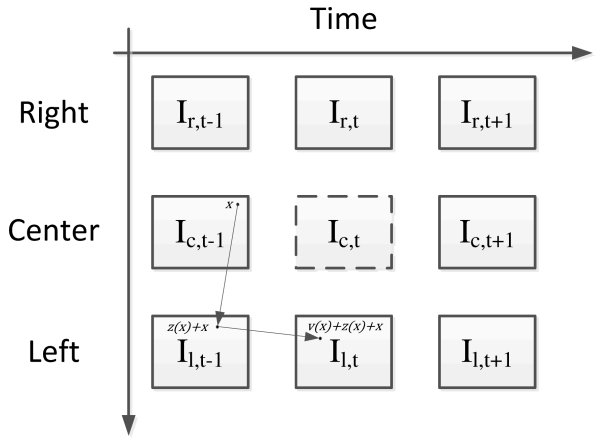


Figure 3.1: The structure of the M-DVC stream. Left and right views, referred to as lateral views are available at the decoder. The central view is WZ encoded, according to a GOP size 2 in this case. From **PAPER 3**.

the Field Of View (FOV) of the other camera. This approach aligns the two views, making the disparity estimation and interpolation task easier. The disparity estimation and interpolation is performed employing the OBMC algorithm, and the method here proposed, divided into an alignment phase and a view interpolation phase through OBMC is called Overlapped Block Disparity Compensation (OBDC).

SI fusion is a difficult task in M-DVC. Furthermore the issue of the calculation of the correlation noise model for the fused SI was not thoroughly investigated. In **PAPER 2** it is proposed to perform fusion not at the SI level, but at the level of the correlation noise model, as it is done in the Multi-Hypothesis decoder used e.g. in **PAPER 1**. The main difference between this approach and the one in **PAPER 1** is that the coefficients, in **PAPER 2**, are not fixed, but they are calculated on a coefficient-by-coefficient basis. The fusion is then improved using a learning approach, where newly decoded DCT coefficients are used to refine the fusion, increasing the decoding performance for the next DCT coefficients. The performance are provided for luminance only, GOP

size 2. Only the rate and PSNR of the central view is analyzed. The proposed decoder is able to provide Bjøntegaard bitrate savings up to 12%, when compared a single SI decoder. The single SI decoder used as benchmark is the one showing the best RD performance between an OBDC-based decoder and an OBMC-based decoder. The system is also extensively tested changing the distance between cameras. A total of 18 configurations are tested. Only for *Book Arrival*, the proposed decoder is outperformed by the single SI OBMC-based decoder by 0.06 dB (Bjøntegaard PSNR difference) in one case: when the PSNR Bjøntegaard difference between the RD performance of the two SIs exceeds 3 dB.

In **PAPER 3** a different approach for SI generation is used, called Time Disparity Optical Flow (TDOF). Here the motion is estimated on the lateral views, and used to estimate the motion in the central view. The concept was first proposed for MVME SI generation [47], the main novelties are the use of OF and scattered set-based filtering and fusion. The SI generation follows the general idea outlined in the MVSim method outlined in **PAPER 1**. The Motion Vectors (MVs) of a lateral view can be matched with the pixels in the central view. The motion field in the lateral view can be used to motion compensates the corresponding pixels in the central view, generating a scattered set of points. The positions in Fig. 3.1: x, $x + z(x)$ and $x + z(x) + v(x)$, show one possible "path", which can be followed to generate an estimation. Four paths can be used, therefore four scattered sets are generated. Instead of interpolating and then fusing the four estimations, following a similar approach to [47], it is proposed to first fuse the sets, filtering out the wrongly matched points, and then perform interpolation. The proposed TDOF method enables rate savings up to 10%, 8.6% and 34% when compared with MVME, OBMC and DCVP respectively. The performance are provided for luminance only, GOP size 2. The rate and PSNR of the lateral views are not taken into account.

3.3 Low-Delay Distributed Video Coding

After showing the improvements that can be achieved when using OF-based techniques in M-DVC, their use is proposed and investigated in monoview low-delay DVC in **PAPER 4**. OF is used for motion esti-

mation and extrapolation. Extrapolation-based SI generation assumes linear motion between frames and uses the motion field between two preceding frames to generate the motion field between the decoded frame at instant $t-1$ and the (unknown) WZ frame, see Fig. 3.2. This estimated motion field is used to motion compensate the frame at instant $t-1$, generating a scattered set of point that is then interpolated.

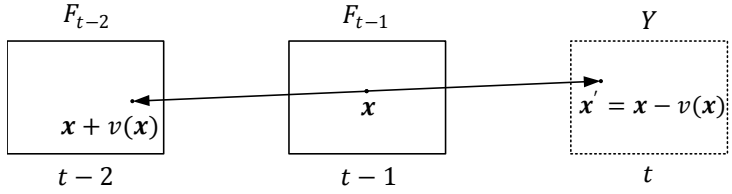


Figure 3.2: Extrapolation uses two already decoded frames to generate the SI for the to-be-decoded frame at instant t.

In **PAPER 4**, OF is also used to refine the SI after each new DCT coefficient is decoded: the motion flow is calculated between the partially decoded WZ frame and the decoded frame at instant $t-1$. The new flow is used to motion compensate the frame at instant $t-1$ obtaining the refined SI. Therefore two OF-based SIs are available: an OF-based extrapolated SI, denoted as EX-OF, and an OF-based refined SI, denoted as REF-OF. REF-OF is not available for the decoding of the DC coefficient. A MH decoder is used to fuse the two OF-based SIs with a block-based Extrapolation. The proposed system outperforms the state-of-the-art block-based Extrapolation SI [65] by up to 1.3 dB in Bjøntegaard PSNR difference, for GOP size 2. Rate and PSNR take into account WZ frames and Key Frames (KFs). Furthermore the system is tested for longer block lengths, finding that using a GOP size 24, for the *Hall* sequence, a sequence characterized by low motion activity, great performance improvements can be achieved: the codec even outperforms a single SI decoder using OBMC, which is an interpolated SI. This suggests that further gains can be achieved if variable GOP structures are used in DVC.

Chapter 4

Novel Approaches for Video-plus-depth Coding

This section deals with video-plus-depth coding. In particular, disjoint encoding of a depth stream and a texture stream is addressed.

In the context of traditional depth map coding, in recent years, many edge-preserving depth map coding algorithms have been presented. Edge adaptive transforms were explored e.g. in [57, 66], nevertheless the proposed methods suffer from high computational complexity, making them impractical. In [66, 67] the concept of *don't care* region is introduced and used to provide enhanced Rate-Distortion (RD) performance. In a *don't care* region depth values can be modified inside a suitable range to obtain a sparse depth representation without affecting the quality of synthesized views. In [68] edge MacroBlocks (MBs) are approximated by a palette and a binary shape map, but the edge structure of previously coded edge MBs is not exploited. Therefore in **PAPER 5** it is proposed an edge-preserving depth map coding architecture, based on H.264/AVC Intra. The proposed architecture allows the exploitation of neighboring edge MBs.

In traditional predictive video coding it is well known that the motion information of a texture stream is highly correlated with the motion of its corresponding depth stream, e.g. in [59] Motion Vectors (MVs) are shared between the depth and the texture stream, leading to improved coding performance. In the context of Distributed Video Coding (DVC), in [60] a depth stream is coded using a Stanford-based distributed codec.

Key Frames (KFs) belonging to the corresponding texture stream, are used to improve the Side Information (SI) generation for the depth stream. Inspired by these works, disjoint encoding and joint decoding of texture and depth information, following a distributed approach, is presented in **PAPER 6** and **PAPER 7**. The main focus of the performance assessment of the proposed coding tools is the comparison with the single SI Overlapped Block Motion Compensation (OBMC)-based decoder proposed in [29], which outperforms the DISCOVER codec and is the basis of the proposed coding architectures. Nevertheless, the coding performance of DISCOVER is provided for completeness.

4.1 Edge-preserving Depth Map Coding

In Multiview Video-plus-Depth (MVD) the accuracy of the depth map plays a major role in Depth-Image-Based-Rendering (DIBR), in particular the edges should be preserved to obtain high quality synthesized views. Edge-preserving joint texture-depth coding schemes have been proposed, e.g. in [69] where decoded texture data is employed to segment the depth image. For each segment the depth data is approximated by a linear surface and only the coefficients are coded. The shape of the segments need not to be coded, because they can be generated both at the encoder and decoder using the reconstructed texture data. In a conventional predictive coding scenario texture and depth must be both available at the encoder. If this is not possible, or not desirable, due to complexity constraints, a different approach can be the use of an encoder for depth maps, equipped with coding tools tailored for the particular properties of depth maps. In **PAPER 5** a new Intra mode for H.264/AVC operating at the MB level is introduced. Each MB is partitioned in two regions , defining a binary mask. Each region is approximated by a constant value. The two constant values are then coded. The binary mask is encoded my means of context-based arithmetic coding. The key novelty of the approach is that already coded constant values or binary masks, belonging to the left or top neighboring MBs, can be used to enhance the coding performance, through *partial* inter-MB coding and *full* inter-MB coding. In partial inter coding only the constant values used in the already coded blocks are inherited by the to-be-decoded one. In the full inter coding mode, constant values, context

Figure 4.1: The neighboring MBs showing edges with similar features are grouped together for enhanced coding performance. From **PAPER 5**, Copyright IEEE 2013.

shape and statistics are inherited. This allows for the creation of edge areas spanning multiple MBs (see Fig. 4.1), enhancing the coding performance. The average Bjøntegaard [64] bitrate saving when compared with H.264/AVC Intra is of about 12% in terms of depth RD performance. In terms of the rate required by the depth maps versus the quality of the synthesized view, the average Bøntegaard bitrate saving is about 25%.

4.2 Coding Tools for Distributed Video-plus-depth

With the solution described in Section 4.1 it is possible to encode and decode a depth stream independently from its corresponding texture stream. Using a DVC-based approach it is possible to perform disjoint encoding of a depth stream and a texture stream, and joint decoding of the two videos. In **PAPER 6** and **PAPER 7** tools for distributed video-plus-depth coding are presented. Initial studies on distributed video-plus-depth coding were presented in [60] and [P12]. The DVC decoder presented in [29] is used as basis for both the presented systems and it is their principal benchmark. In the analyzed scenario a texture stream and its corresponding depth stream are independently encoded and jointly decoded. The setup is similar to the ones seen in Multiview Distributed Video Coding (M-DVC): one of the two streams is H.264/AVC Intra encoded, the other is Wyner-Ziv (WZ) encoded, using the usual Group Of Pictures (GOP) structure of the Stanford-based codecs. In **PAPER 6** the depth stream is Intra coded and it is used to improve the SI generation for the texture stream. The frames belonging to the texture stream are denoted as T_i where i is a temporal index, the

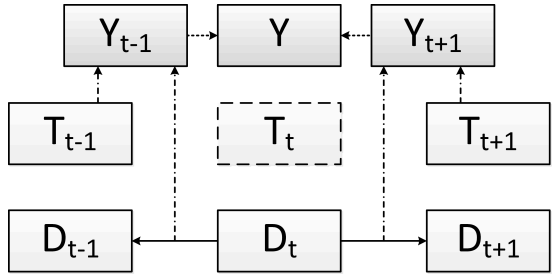


Figure 4.2: The stream structure of the studied scenario, for GOP size 2. Solid lines indicate frames available at the decoder, dashed lines indicate a WZ frame. From **PAPER 6**, Copyright IEEE 2013.

frames belonging to the depth streams are denotes as D_i. The GOP of size 2 structure is used. The to-be-decoded frame is the texture frame at instant t, the depth frames D_i with $i = t-1, t, t+1$ and the texture frames T_{t-1} and T_{t+1} are available at the decoder, as depicted in Fig. 4.2. The simplest method consists in estimating the motion between D_t and D_{t-1}, and use the calculated MVs to motion compensate T_{t-1}, obtaining Y_{t-1}. The same process can be applied to the frames D_t, D_{t+1} and T_{t+1}, leading to the calculation of Y_{t+1}. The final SI can be calculated as average of the two factors Y_{t+1} and Y_{t-1} and the residual can be estimated by the difference of the factors.

This simple approach is implemented using block-based or Optical Flow (OF)-based techniques for the motion estimation on the depth stream. A block-based motion estimation technique called Adaptive Rood Pattern Search (ARPS) [70] is used to calculate the SI denoted as D2T BB, OF is used for D2T OF. In **PAPER 6** another SI generation method is proposed, exploiting the formulation proposed in [71]. A joint depth-texture SI calculating is proposed through the minimization of

$$E(v) = \int \lambda_1 \|C_T(\boldsymbol{x}, v)\| + \lambda_2 \|C_D(\boldsymbol{x}, v)\| + \|\mathscr{D}v(\boldsymbol{x})\| \, \mathrm{d}\boldsymbol{x}, \tag{4.1}$$

where

(a) T2T (b) D2T OF (c) DT2T

Figure 4.3: Texture SI generation for video-plus-depth using optical flow and different sets of constraints. The SI quality, measured in PSNR is (a) 26.1 dB, (b) 27.1 dB and (c) 29.3 dB. The sequence is *Dancer* from Nokia Research [72].

$$C_T(\boldsymbol{x}, v) \triangleq T_{t+1}(\boldsymbol{x} + v(\boldsymbol{x})) - T_{t-1}(\boldsymbol{x} - v(\boldsymbol{x})), \tag{4.2}$$

$$C_D(\boldsymbol{x}, v) \triangleq D_{t'}(\boldsymbol{x} + v(\boldsymbol{x})) - D_t(\boldsymbol{x}), \tag{4.3}$$

and t' is either $t-1$ or $t+1$ depending on which factor is being calculated: Y_{t-1} or Y_{t+1} respectively. This formulation allows to jointly optimize two constraints, a symmetrical one, denoted as $C_T(\boldsymbol{x}, v)$ and a asymmetrical $C_D(\boldsymbol{x}, v)$. With the presented framework it is possible to investigate three SI generation methods: T2T where only the symmetric constraint is considered (i.e. $\lambda_2 = 0$), D2T OF where only the asymmetric constraint is considered (i.e. $\lambda_1 = 0$) and finally DT2T where both constraints are used. The symmetric constraint, commonly used in SI generation, provides good results in slow or medium motion parts, but it is imprecise in fast motion parts, it is possible to see this from the arms of the Dancer in Fig. 4.3(a). On the other hand the asymmetric constraint is unable to accurately predict shadows, as it can seen in Fig. 4.3(b). Generating a motion field that obeys to both constraints, as in DT2T, generates a superior quality SI, which is able to effectively predict both shadows and fast moving objects.

In **PAPER 6** it is shown that a single SI decoder, employing DT2T is able to outperform block-based and OF-based competing methods for what concerns the RD performance of the whole system. In case of uncompressed depth maps DT2T outperforms the benchmark OBMC-based, single SI decoder by up to 1.32 dB in Bjøntegaard PSNR distance on the WZ frames for a GOP size 2 structure. Furthermore, a Multi-

Hypothesis (MH) decoder is used to achieve even higher performance, fusing OBMC, T2T and DT2T, the Bjøntegaard PSNR gains are up to 1.48 dB on the WZ frames for GOP 2. Similar trends are observed when compressed depth maps are used. The best performance are observed for *Ballet* and *Dancer* sequences, both sequences characterized by moderate motion activity. For an high motion activity sequence like *Breakdancers* the gains are lower but still relevant.

PAPER 7 addressed the dual problem of **PAPER 6**: the texture stream is Intra coded, while the depth stream is WZ coded. **PAPER 7** uses block-based and OF-based motion estimation algorithms, but no joint depth-texture motion estimation is employed, the fusion is performed only through multi-hypothesis decoding. The proposed solution is able to outperform the single SI OBMC-based decoder by up to 4.95 dB in Bjøntegaard PSNR difference, for GOP size 8, on the WZ frames for the *Dancer* sequence.

Chapter 5

Rate-Adaptive BCH Codes for Distributed Source Coding

Rate Adaptive (RA) error-correcting codes are at the basis of many practical feedback-based Distributed Video Coding (DVC) coding solutions. Here RA BCH (Bose-Chaudhuri-Hocquenghem) codes for feedback-based Slepian-Wolf coding are investigated as an alternative to traditional Turbo [2] and Low Density Parity Check Accumulate (LDPCA) [16] codes. Broadly speaking a RA code is an error-correcting code having the capability to vary its strength (i.e. increase or decrease the number of parity symbols) in order to adapt to the number of errors in the given block of bits. In order to be of practical interest, the already generated syndromes (i.e. parity bits) must remain the same when the strength (i.e. the rate) of the code is increased. In this section rate refers to the parity bits sent on the channel, therefore the higher the rate, the higher the number of parity bits and therefore the higher the strength of the code. When using RA codes, the encoding and decoding procedure can be structured as follows: at the beginning of the decoding process few syndrome bits are sent by the encoder. In case of *decoding failure* (i.e. the decoder is unable to decode the original bitstream), the decoder requests new syndromes through the feedback channel. The encoder, upon request, generates more syndrome bits using the code to an higher rate level. Among the generated syndromes, some are new, while the others

are equal to the already sent ones. Therefore only the new syndromes are sent. The decoder concatenates the new syndromes with the old ones, obtaining the parity bits generated by the encoder using the more robust code. In many proposed codecs, [2, 16] the encoder encodes only once, using the code to its higher rate, and stores the parity bits in a buffer, sending subsets of the bits upon request.

Codes for feedback-free Distributed Source Coding (DSC) [13, 73] show lower performance when compared with the RA counterparts. This is due to the fact that the code is designed to cope with the large variation of the number of errors in case of short blocks. Therefore, for some blocks, too many parity bits are sent, for other blocks not enough bits are sent. RA codes can easily adapt to the varying number of errors, enabling the DSC codec to achieve better performance. Turbo and Low Density Parity Check (LDPC) codes were studied in e.g. [74] for long block lengths. The use of short block lengths allows the system to have low latency. This may be desirable for some applications. Furthermore, with shorter block lengths, decoded bits can be obtained at a fine granularity. In the context of adaptive codes this may be desirable, in order to allow faster convergence of the model of the source used by the decoder. In the context of DVC in [38] superior Rate-Distortion (RD) is achieved using various coding modes, e.g. skip, intra. The proposed RA BCH codes can be used in a similar way, introducing a new coding mode, in cases where the Side Information (SI) is estimated to have few errors.

The focus of the works presented in **PAPER 8** and **PAPER 9** are on the implementation and analysis of a feedback-based Slepian-Wolf codec (see Fig. 5.1) in the short block-length, high correlation scenario. The encoder has only access to the source X, which is considered a binary i.i.d. source. The decoder has access to the syndrome bits sent by the encoder and the SI Y. The relationship between X and Y is modeled by the error pattern E, also an i.i.d. binary source:

$$Y = X + E. \tag{5.1}$$

The error probability p is the probability $P[E_i = 1]$ where E_i is the i-th element of E.

If s syndromes are used for decoding, up to $t(s)$ errors in Y can be corrected, and the probability of *decoder error* is proportional to $1/t(s)!$. A decoder error happens when a wrong estimation of X is accepted as

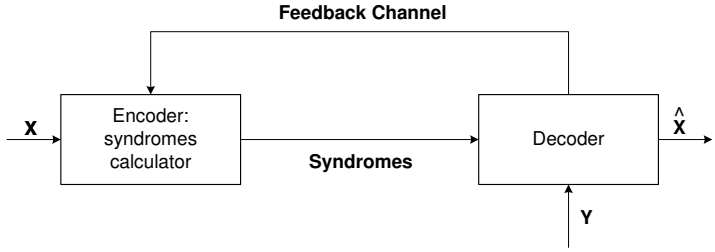


Figure 5.1: Slepian-Wolf Codec using RA BCH codes, with feedback channel for encoder-driven rate-adaptation. From **PAPER 9**, Copyright Springer 2013.

correct. Therefore techniques to improve the reliability of the decoded result have been proposed and tested in **PAPER 8** and **PAPER 9**. The checking procedure relies on requesting $c(s)$ extra syndrome bits to check a result decoded using s syndromes. If the decoded result obtained using the extra check bits matches the previous one, the result is accepted. $c(s)$ is a non-increasing function of s defined by the *checking strategy*.

In **PAPER 8** the use of RA BCH codes is first presented, proving that RA BCH codes are able to outperform conventional LDPCA codes for conditional entropy $H(X|Y) \leq 0.25$. Secondly a mathematical model to estimate the performance of RA BCH codes is presented. The model is able to correctly predict the performance of the code, given the strategy, the block length and p. To further increase the reliability of the decoded result, Hierarchical CRC check is introduced: a c' bit CRC check is use to check a set of f blocks decoded with the already introduced RA BCH codes. The strength of the technique is that, for each block, the rate penalty for the extra check is c'/f. All these results are presented in Fig. 5.2. Lastly the case of unknown p is addressed, providing methods to perform decoding while estimating the error probability.

In **PAPER 9** the model presented in **PAPER 8** is further refined. While in **PAPER 8** the extra checking procedure is modeled like a CRC check, the new model takes into account how the checking procedure is structured, and models it accordingly. While the model in **PAPER 8** is able to correctly predict the behavior of the code in a wide range of cases, it is unable to reach good precision in some cases, e.g. in Fig. 5.3 it can be noted that the model presented in **PAPER 8**, referred to as "DCC Model" is unable to correctly predict the behavior of the BCH code (referred to as "Implemented Decoder"). On the other hand the

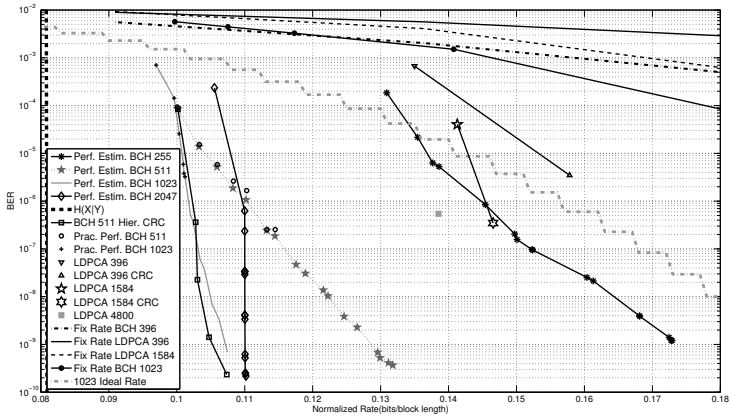


Figure 5.2: Performance of RA BCH Slepian-Wolf codec when compared with state-of-the-art RA and Fix-Rate error correcting codes. From **PAPER 8**, Copyright IEEE 2012.

model presented in **PAPER 9** and denoted as "Proposed Model" is able to correctly predict the behavior of the code, see Fig. 5.3. Similar behavior is observed in other cases in **PAPER 9** and provides a completely mathematical model of Hierarchical CRC check. **PAPER 9** shows also the superiority of using extra syndromes for checking when compared with fixed or variable CRC checking strategies.

The presented models are able to predict quite precisely the rate: the difference between predicted rate and simulated rate is always below 0.04%. The Bit Error Rate (BER) estimation is less precise, nevertheless it is able to provide sufficiently accurate results: one of the worst cases is for block length 1023 and $p = 0.035$ using the syndrome checking strategy. In this case the BER performance of the decoder is 9.36×10^{-6} while the estimation of the model is 8.21×10^{-6}. Comparing the three checking strategies proposed in **PAPER 9** it can be seen that the Fixed CRC check and the Variable CRC check are unable to provide better performance than the syndrome check, because of their inability to adapt to the estimated reliability of the decoded result. Lastly, for $p = 0.005$ the LDPCA code having length 1584 requires twice as many bits as the RA BCH code having length 1023.

Summarizing RA BCH codes are an efficient alternative to LDPCA and Turbo codes for high correlation scenarios, in the case of short block lengths. Furthermore RA BCH codes show lower complexity when com-

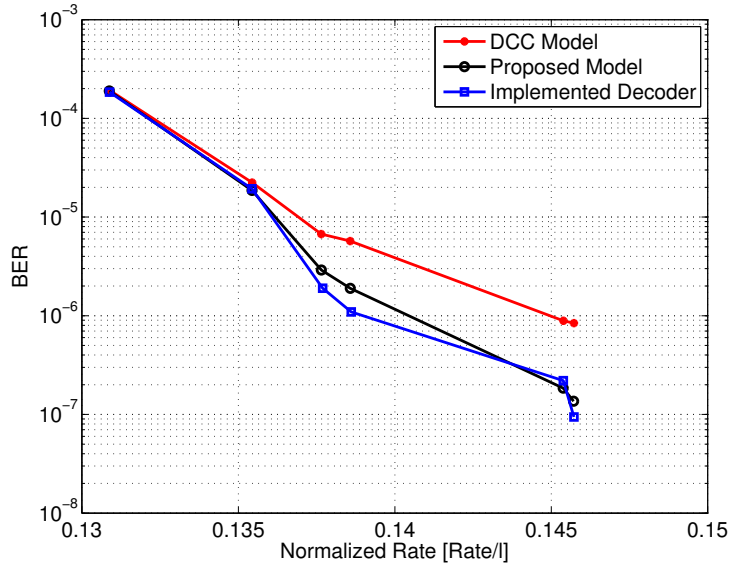


Figure 5.3: Comparison of accuracy of the model presented in **PAPER 8** (DCC Model) against the model presented in **PAPER 9** (Proposed Model) when compared with a RA BCH decoder (Implemented Decoder). for $p = 0.01$ and block length 255.

pared with LDPCA codes, and mathematical models are able to accurately predict the code performance.

Chapter 6

Description of Ph.D. Publications

In this Chapter a description of the publications included in the thesis is provided, highlighting novelties and main results. The papers are divided into three groups, following the order used in the previous chapters. This section is designed to be as much self-contained as possible, giving an overview of the works, therefore overlaps with previous chapters exist.

6.1 Tools for Distributed Video Coding

PAPER 1: Multi-Hypothesis Distributed Stereo Video Coding

This paper introduces a novel codec in the context of distributed stereo video coding. This work is an evolution of an early study presented in [P11].

Two Side Informations (SIs) generation methods are presented, namely Motion Vector Similarity (MVSim) and Difference Projection (DP). Both methods generate the SI estimating the motion on the *master* view, exploiting the *support* view. The master view is coded with a distributed approach. The support view needs only to be available at the decoder, and no further requirements are needed. MVSim uses the disparity estimation between support and master view to disparity compensate the motion field calculated in the support view. DP, on the other hand, dispar-

ity compensates the difference between two consecutive frames in the support view. The disparity compensation in DP can introduce artifacts, therefore it is applied only to the pixels showing enough motion activity (i.e. a big enough difference between the two consecutive frames in the support view). As opposed to the method presented in [61] where three frames in the support view need to be decoded, here, two frames need to be available ($I_{s,t-1}$ and $I_{s,t}$) making the synchronization requirements less stringent. Secondly, a residual estimation method for on-line correlation noise model calculation is proposed, avoiding the use of off-line residual. Using an off-line residual, which requires the knowledge of the original Wyner-Ziv (WZ) frame at the decoder, would compromise the practical interest of the system. The generated intra-view SI, either via MVSim or DP, is not able to deliver comparable Rate-Distortion (RD) performance to Overlapped Block Motion Compensation (OBMC). OBMC outperforms MVSim and DP by up to 5 dB measured with Bjøntegaard Peak Signal-to-Noise Ratio (PSNR) distance [64]. Fusion methods proposed in literature [49,61] are not robust enough to successfully fuse an inter-view SI and OBMC. It is proposed to use the Multi-Hypothesis (MH) decoder, described in [32] for efficient SI fusion. The MH decoder is able to robustly fuse an inter-view SI and OBMC, showing improved performance over other fusion techniques, on the other hand its complexity is up to 6 time higher than the fusion-based, Distributed Video Coding (DVC) decoders. The MH decoder using OBMC and DP is the best performing one, outperforming a single SI decoder based on OBMC by up to 0.8 dB, Bjøntegaard PSNR difference, on the WZ frames only, Group Of Pictures (GOP) 2, while the MH decoder using OBMC and MVSim shows gains up to 0.59 dB. Summarizing DP shows better performance when compared with MVSim. Furthermore, although the decoder complexity is not a major issue in DVC, DP shows lower computational complexity: it does not require the calculation of the motion field in the support view and the disparity estimation is carried out only on a subset of the pixels (up to 11.39% in the reported experiments).

PAPER 2: A Robust Fusion Method for Multiview Distributed Video Coding

Many works in Multiview DVC literature address a three camera setup: left, right and central camera. The central camera is usually WZ coded following the usual GOP structure of Stanford-based DVC codecs. The lateral cameras (left and right) are known at the decoder, and they are used to generate an inter-view SI. The inter-view SI is then fused with a temporal (intra-view) SI generated using the Key Frames (KFs) belonging to the central view. Usually left and right views are close to the central view, e.g. in [49] the distance between central and right (or left) camera is 6.5 cm. In **PAPER 2** the problem of unknown and higher distance between cameras is addressed, and the use of a novel fusion method based on fused distributions and learning is presented.

The SI generation technique, called Overlapped Block Disparity Compensation (OBDC) is constituted by two steps: pre-alignment and view interpolation. The pre-alignment phase employs a sliding-window approach. The Field Of Views (FOVs) of the lateral cameras overlap, but in the left (resp. right) frame the lateral area has no match in the right (resp. left) frame. This is due to the fact that part of the scene falls outside the FOV of one of the two cameras but it is in the FOV of the other. The pre-alignment step removes these areas, generating an aligned version of the frames. The interpolation phase accommodates only for local disparity changes, therefore its task is much easier. For view interpolation OBMC is used. OBDC is first compared with Disparity Compensated View Prediction (DCVP). For fairness, DCVP uses OBMC as interpolation system, without the pre-alignment phase. It is shown that OBDC is able to perform better than DCVP for a wide range of camera setups. In particular, while the parameters (e.g. search range) used for DCVP SI generation are fine tuned for each setup to achieve the best RD performance, OBDC uses always the same parameters. The proposed fusion method relies on the fusion of the estimated distributions instead of fusion of the SIs. This allows to leverage well-known noise models developed in the monoview coding scenario. Lastly the fusion is refined along the decoding process, using already decoded DCT coefficients to refine the fusion.

When the last DCT band is decoded, the partially decoded WZ frame is used to guide the final fusion step. The newly fused SI is used for a last refinement step on the decoded WZ frame. The proposed solution is able to achieve gains up to 0.9 dB in Bjøntegaard PSNR distance when compared with best performing single SI decoder. The best performing single SI is chosen between a single SI decoder using OBMC and one using OBDC, according to the RD performance. The performance are calculated on the central view only, on all the frames, both WZ frames and KFs. The GOP structure has size 2.

PAPER 3: Joint Disparity and Motion Estimation Using Optical Flow for Multiview Distributed Video Coding

In this paper a joint disparity-motion estimation method is proposed for SI generation. In many works, e.g. [49] a temporal SI and an inter-view SI are generated and then fused. Another approach, is taken by a minor number of works, e.g. [47] where the motion is estimated in the lateral views and used to estimate the WZ frame. In **PAPER 1** the inter-view SI is generated following a similar approach, but it still relies on block matching for pixel disparity or motion calculation. Furthermore, given the poor performance of the inter-view SI a fusion with the temporal SI is required to achieve RD gains. In this work it is proposed to use only an inter-view SI generation method, referred to as Time Disparity Optical Flow (TDOF) SI generation system, where the motion is estimated on the lateral views and then applied to the KFs of the central view to generate the SI using Optical Flow (OF). More precisely, four different estimations can be done, following four different paths, one path is depicted in Fig. 3.1. The first step of TDOF is a pre-alignment phase, similar to **PAPER 2**, to remove unmatched areas. Secondly disparity is estimated between a KF in the central view and its temporally corresponding frame in the lateral view, using the path depicted in Fig. 3.1 this calculation is done between the aligned versions of the frames $I_{c,t-1}$ and $I_{l,t-1}$. Then motion is calculated in the lateral view, in the case of Fig. 3.1 the motion is calculated between $I_{l,t-1}$ and $I_{l,t}$. The motion flow $v(x)$ is directly coupled with the position x in $I_{c,t-1}$ and it

can be used to motion compensate $I_{c,t-1}$ obtaining a scattered set of points. Since the paths are four, four scattered set of points are calculated. Instead of interpolating the sets and then fusing the obtained estimations, it is proposed to fuse the sets, and then interpolate the fused sets. The path of Fig. 3.1 passes through $I_{l,t-1}$, the corresponding set of points is denoted as $S_{l,t-1}$. The complementary path in the left view passes through $I_{l,t+1}$ and therefore the corresponding set is denoted as $S_{l,t+1}$. A similar naming convention is used for the right view. The procedure for the left view is repeated on the right view, therefore here it is presented only for the left view. The set S_l is defined as:

$$S_l = S_{l,t-1} \cup S_{l,t+1}. \tag{6.1}$$

The new set S_l is composed by twice the number of points of one of the original frames. Therefore given the overabundance of points it is possible to apply filtering, to remove wrongly matched pixels. After the filtering, S_l and S_r are interpolated and the two resulting frames are averaged to generate the final SI.

The use of fusion and filtering of scattered sets enables the generation of higher quality SI when compared to the case when simple interpolation and averaging of the estimations is performed. The TDOF SI generation method leads to Bjøntegaard bitrate savings up to 10%, 8.6% and 34% when compared with MultiView Motion Estimation (MVME), OBMC and DCVP, respectively. The performance is calculated on all the frames (WZ frames and KFs) for the central view only, GOP 2.

PAPER 4: Low Delay Wyner-Ziv Coding Using Optical Flow

In this work OF is used in the context of Low-Delay DVC. OF is used for SI generation and refinement. The SI estimates the WZ frame having time index t. The motion is calculated between two already decoded frames, having time indexes $t-1$ and $t-2$. The newly calculated motion flow $v(x)$ has origin in the frame at instant $t-1$, i.e. x is a pixel position in the frame at instant $t-1$. Assuming linear motion, $-v(x)$ is the flow estimating the motion between the frame at instant $t-1$ and the (unknown) WZ frame. The new field is used to motion compensate the frame at instant $t-1$, generating

a scattered set of points. Differently from **PAPER 3**, filtering is not applied on the scattered set. Interpolation is performed on the set and the SI is generated. This SI is referred to as EX-OF. The similar block-based method proposed in literature is referred to as EX-BB.

The linear motion assumption, used to generate the SI, is a poor approximation of the real motion, therefore the quality of the SI is relatively low, although OF is used. It is therefore proposed to use OF to estimate the motion between the partially decoded WZ frame and the KF at instant $t-1$. The new flow has center in the partially decoded WZ frame, therefore no holes are generated when the frame at instant $t-1$ is motion compensated. This refined SI, unavailable for the decoding of the DC coefficient of the DCT, is referred to as REF-OF.

It is proposed to fuse the three (two for the DC band) SIs using a MH decoder. The proposed system is able to provide Bjøntegaard PSNR gains up to 1.3 dB (on all the frames) when compared with a single SI DVC decoder based on EX-BB in the case of GOP size 2 structure. Furthermore, the proposed decoder is able to outperform an advanced interpolation based system like OBMC when a GOP size of 24 is used, in the low-motion sequence *Hall*.

6.2 Video-plus-depth Coding

PAPER 5: Edge-preserving Intra Depth Coding based on Context-coding and H.264/AVC

The paper proposes a new Intra mode for H.264/AVC, operating on the MacroBlock (MB) level, specifically targeted to areas containing edges of arbitrary shape. Such blocks are usually problematic if handled using the conventional DCT transform, while the proposed method aims to preserve the edges with a higher level of fidelity, given their importance for precise Depth-Image-Based-Rendering (DIBR).

Each MB is partitioned into two areas. Each area is approximated by a constant value, which is coded. A binary mask is used to

provide information on the shape of the regions. The binary mask is coded using context-based arithmetic coding with adaptive template selection. The possible templates are 4 and the chosen one is explicitly indicated in the generated bitstream. The chosen template is the one providing the lower rate. This coding procedure is referred to as *Intra EDGE* coding.

Other two *Inter* modes are available: *Partial Inter EDGE* coding and *Full Inter EDGE* coding. They both require the presence of a left or top *EDGE* coded neighboring block. In Partial Inter coding, the current MB inherits the constant values employed by the neighboring MB, therefore the constant values are not explicitly indicated in the bitstream generated for the MB. The bitstream contains flags to indicate what neighbor is used (if both top and left are available) and if partial or full inter mode is used. In the case of *full inter* coding the template of the context is the same of the selected neighbor. Furthermore, the neighboring MB statistics are used as starting point for the context coding, i.e. the mask is encoded as part of a bigger mask spanning more MBs, improving the coding efficiency. If *full inter* mode is employed, the shape of the context and the constant values are not specified in the bitstream, but flags to allow the decoder to correctly infer the information are given, e.g. flags to determine what neighbor is used or flags to determine what mode is used.

The proposed method allows to code a block in up to 5 different EDGE modes: *Intra EDGE*, the two *Inter* modes using the left neighbor, and the two *Inter* modes using the top neighbor. The EDGE mode(s) are integrated into the RD optimization framework of H.264/AVC and tested together with the usual H.264/AVC Intra modes. The mode providing the lowest RD cost is selected. Improvements in terms of Bjøntegaard rate saving for the depth maps are of about 12%. The Bjøntegaard rate saving for view synthesis are of about 25%.

PAPER 6: Texture Side Information Generation for Distributed Coding of Video-Plus-Depth

In this paper the use of depth maps for improved SI generation in video-plus-depth is investigated. The employed framework is

based on the work presented in [71]. The OF is calculated using the efficient TV-L^1 formulation. Two constraints are jointly optimized: one imposing linear and symmetric motion on the texture stream, the other imposing asymmetric motion on the depth stream. The texture stream is WZ encoded according to a GOP structure of size 2, the depth stream is, in the reported experiments, H.264/AVC Intra encoded. Nevertheless the depth stream can be encoded with any available encoder, as long as it is encoded independently from the texture stream. Two texture frames are available T_{t-1} and T_{t+1}, since GOP 2 is used. The depth frames D_{t-1}, D_t and D_{t+1} are also available. The symmetric constraints is based only on the texture frames, and imposes that the motion between frames is symmetric. Therefore, ideally, the goal when minimizing the symmetric constraint is to find a motion flow $v(x)$, such that: $T_{t+1}(x + v(x)) - T_{t-1}(x - v(x)) = 0$. For each WZ frame only one symmetric constraint exists. The asymmetric constraint is imposed to the corresponding pixels of the depth map: defining $t' = t + 1, t - 1$, $v(x)$ should ideally be such that: $D_{t'}(x + v(x)) - D_t(x) = 0$. The two constraints are jointly minimized, generating a flow where, ideally, smaller details are matched using depth information, while bigger details (including lighting changes and shadows, which are not visible from depth data) should be matched using the texture frames. The joint minimization process uses two weighting factors λ_1 and λ_2 to determine the influence of the symmetrical and asymmetrical constraints respectively. If $\lambda_2 = 0$ only the symmetrical constraint is used, the method is denoted T2T and it is the same method proposed in [P10], therefore the proposed framework is a more general version of the OF framework presented in [P10]. If $\lambda_1 = 0$ only depth information is used and the method is referred to as D2T OF. When $\lambda_1 > 0$ and $\lambda_2 > 0$ the SI is denoted as DT2T and, according to the presented results, leads to superior RD performance when compared with all the other SI generation systems. For completeness a block-based version of the D2T OF approach is presented, and it is denoted as D2T BB, but the performance achieved are relatively low when compared with the OF-based methods. DT2T is able to provide Bjøntegaard PSNR gains on a single SI, OBMC-

based decoder up to 1.32 dB (GOP 2, WZ frames only). To achieve even higher gains a MH decoder using OBMC, DT2T and T2T is employed: the improvements over the single SI, OBMC-based decoder are up to 1.49 dB in Bøntegaard PSNR distance, GOP 2, WZ frames only.

PAPER 7: Distributed Multi-Hypothesis Coding of Depth Maps using Texture Motion Information and Optical Flow

This paper addresses the dual of the problem presented in **PAPER 6**, an early study was presented in [P12]. In the discussion, GOP of size 2 is considered, but results are also reported for longer GOP sizes. The basic assumption is that the frames D_{t-1}, D_{t+1}, T_{t-1}, T_t and T_{t+1} are available at the decoder, the WZ frame is D_t. For the naming convention please refer to the previous discussion on **PAPER 6**. It has to be noted that the texture frames are denoted as C_t in **PAPER 7**, but, in order to make the section consistent the naming convention of **PAPER 6** is used. Three SI generation methods are analyzed in this work: OBMC, OF and BB, their goal is to estimate the frame D_t. OBMC is used as benchmark and uses only the depth KFs D_{t-1} and D_{t+1} to generate the SI. Both the BB (Block-Based) method and the OF (Optical Flow-based) method use the texture stream to generate the SI. They are used in the opposite way the D2T method is used in **PAPER 6**: motion is estimated between $T_{t'}$ and T_t, with $t' = t-1$ or $t' = t+1$ and the motion flow is then used to compensate $D_{t'}$, generating the estimation $Y_{t'}$. The average of the two obtained estimations is the final SI. The BB method employs, for motion estimation, the Adaptive Rood Pattern Search (ARPS) motion estimation algorithm, which does not provide the lowest MSE (Mean-Squared-Error) between the motion compensated texture frame and the original one, however, it is able to capture the motion between the frames in a robust way. The OF methods employs, as the name suggests, OF, using the asymmetric formulation presented in (2.7). In this asymmetric formulation the position x belongs to the frame at time index t. The three SIs are then fused using a MH decoder, leading to rate savings up to 49.06%, on the WZ frames, over a single SI, OBMC based decoder, for GOP size 8.

6.3 Rate-Adaptive Codes for Distributed Source Coding

PAPER 8: Rate-adaptive BCH Coding for Slepian-Wolf Coding of Highly Correlated Sources

In Stanford-based DVC codecs the Rate Adaptive (RA) error correcting code is one of the main elements determining the performance of the whole system. A RA error correcting code is a code able to increase its error correcting strength (i.e. increasing the number of produced parity bits) to adapt to the number of errors in the current block. Fixed-rate codes, on the other hand, have to be designed to cope with the large variation in number of errors in the blocks. As it is demonstrated in this work, the performance of a feedback-free non-rate-adaptive code is clearly inferior compared with the performance of its rate-adaptive counterpart, for short or medium block lengths. In the analyzed scenario, the encoder has access to the source signal X, in this paper an i.i.d. binary source. The decoder has access to the SI, a signal correlated to X. The error pattern E defines the differences between X and Y: $Y = E+X$. In the addressed scenario, the rate-adaptation process is guided by the decoder. The encoder starts the decoding process sending a small amount of parity bits. The decoder tries to reconstruct X using Y and the parity bits. If the decoding is successful the result can be directly accepted or an extra checking can be applied, e.g. a Cyclic Redundancy Check (CRC) in the case of the DISCOVER codec [19]. If the extra checking (if any) is successful the result is accepted. In case of *Decoder Failure*, the decoding or the extra checking fail. When a *Decoder Failure* happens, the decoder detects that the received parity bits are not enough to correct the errors, and new bits are requested via a feedback channel. The encoder, upon request, uses the RA codes to a higher level of correcting strength. A new set of parity bits is generated. If the error correcting code is chosen from a family of codes where more parity bits are produced, without changing the previous ones, only the new ones need to be sent. The decoder concatenates the previously received bits with the new ones and the decoding is attempted again. This

work proposes RA BCH (Bose-Chaudhuri-Hocquenghem) codes as a possible alternative to conventional Turbo or Low Density Parity Check Accumulate (LDPCA) [16] codes for feedback-based Distributed Source Coding (DSC). The high-correlation, short block length scenario is addressed. In such scenario RA BCH codes outperform conventional RA codes. Interestingly, the performance of RA BCH codes can be predicted by a mathematical model, which is detailed in the paper. In the paper the concept of *strategy* is introduced, which is the function defining the strength of the extra check applied on the decoded result.

A *Decoder Error* happens were a wrongly decoded block is accepted as a correct one. The reliability of the decoded result is an increasing function of the used parity bits, in other words, the probability of a Decoder Error decreases when the rate increases. Therefore, if the RA BCH decoder decode a block using few parity bits, the probability that it is accepting an erroneous decoded result is high. On the other hand if an accepted result is decoded using a high number of parity bits, its reliability is high, and a less strong check, using fewer bits, can be used. In this paper it is proposed to vary the strength of the extra check according to the reliability of the decoded result. The function defining the number of bits required for the extra checking is referred to as *strategy*. The use of a model for the performance of the BCH code allows to perform a convex hull optimization on the set of possible strategies, selecting only the best performing ones. In this work the extra check is performed requesting new parity bits to the encoder. Then the new result is checked against the old one, and accepted only if they match. Nevertheless, the model addresses the checking part modeling it like a CRC. Despite this strong assumption the model achieve an overall good precision in predicting the performance of a code.

The paper proposes the use of hierarchical CRC check for further improving the reliability of the decoded result. The idea is to share the cost of a CRC check of c' bits among f blocks. The f blocks are already decoded using the RA BCH decoder and the checking already described. For low block error rates and modest values of f, when a decoding error is detected by the c' CRC, with high probability, only one of the blocks was wrongly decoded.

Therefore it is possible to assume that the probability of decoding error decreases for each block of about $2^{-c'}$, but the rate cost for each block is c'/f. This assumption is tested using a hybrid model.

Lastly the case of unknown error probability is addressed. Only the parameters of the distribution of the error probability are known. In the paper efficient estimators and methods to adapt the decoding strategy to the unknown error probability are explored.

PAPER 9: Rate-adaptive BCH Codes for Distributed Source Coding

This work is an extended version of **PAPER 8**. **PAPER 8** is improved in various aspects: the model of the code, a complete model for the hierarchical CRC check and comparison against other checking procedures.

The model describes accurately the checking procedure, instead of assuming its behavior to be same of a CRC check. This leads to improved performance in the capability of the model to accurately predict the behavior of the code.

A complete mathematical model for the Hierarchical CRC check is presented, as opposed to **PAPER 8** where a hybrid model was used. In **PAPER 8** a RA BCH decoder was employed and only the check part was modeled. This system was therefore impractical, because extensive simulations were needed to provide precise estimations.

Finally other checking procedures other than the one presented in **PAPER 8** are used: *fixed CRC* and *variable CRC* checking procedures are proposed, implemented, modeled and compared. In the case of fixed CRC check a CRC check having a fixed number of bits is requested, without any consideration on the reliability of the decoded result. In the case of variable CRC, its strength is matched with the reliability of the decoded result. If the CRC check detects an error new syndromes are required and the CRC bits are stored to be used for future checking. This is the main difference between the syndrome-based approach and the variable CRC check. If the syndrome check reveals an error, the syndrome bits used for checking are employed for decoding, the next check-

ing procedure will use a number of check bits matched with the reliability of the decoded result. On the other hand, the variable CRC check procedure will continue to use the same CRC check after a failed check. This is a waste of rate, because a strong CRC will be used when there is no need for it, while the syndrome check approach allows the system to go back. The superiority of the syndrome check approach is tested in the paper through extensive simulations, which confirm this argument.

Chapter 7

Conclusion

Wireless Sensor Networks (WSNs) promise to bring to life a new kind of services, where self-organizing networks are able to provide information of the environment without any central control. If the services are based on video delivery and processing the Video Sensor Network (VSN) needs new approaches to distributed compression and signal processing. The application of Distributed Source Coding (DSC) principles to video compression, referred to as Distributed Video Coding (DVC) is a possible solution to this problem. DVC has two decisive advantages over predictive coding solutions: lower encoding complexity and the possibility of leveraging the redundancy between cameras recording the same scene from different points of view. These two strengths can be of interest in the video-plus-depth scenario also, where the communication between a depth camera and a texture camera may be not desirable or where it is convenient to transfer some of the complexity at the decoder. Unfortunately DVC still shows inferior Rate-Distortion (RD) performance when compared with state-of-the-art predictive coding solutions, such as H.264/AVC.

In the context of Multiview Distributed Video Coding (M-DVC) the main contributions of this thesis are the development of new inter-view Side Information (SI) generation techniques and the study of the fusion problem. For the Stereo DVC scenario Difference Projection (DP) is introduced as inter-view SI. On-line residual calculation is proposed in order to make the codec practical. DP is robustly fused with Overlapped Block Motion Compensation (OBMC), improving a single SI OBMC-

based decoder by up to 0.8 dB Bjøntegaard Peak Signal-to-Noise Ratio (PSNR) difference on the Wyner-Ziv (WZ) frames. In M-DVC a novel inter-view SI generation system, called Time Disparity Optical Flow (TDOF) is also introduced, based on Optical Flow (OF). Instead of generating a temporal and an inter-view SI separately, TDOF estimates the motion on the central view from the motion of the lateral views. The concept of scattered set of point filtering and fusion is used, instead of the usual approaches based on hole filling and averaging. The proposed method leads to Bjøntegaard rate savings up to 10%, 8.6% and 34% when compared with MultiView Motion Estimation (MVME), OBMC and Disparity Compensated View Prediction (DCVP) respectively. The problem of high disparity between cameras is also addressed: Overlapped Block Disparity Compensation (OBDC) is introduced as a stable inter-view SI generation method, able to cope with unknown and high disparities. A fusion method, based on joint distribution calculation and learning is also proposed. The proposed system is able to outperform a monoview OBMC-based decoder by up to 0.9 dB Bjøntegaard PSNR difference.

For low-delay monoview DVC a OF-based SI generation and refinement system is introduced. The proposed method is able to outperform a block-based low-delay DVC system by up to 1.3 dB Bjøntegaard PSNR difference. For low motion sequences, using a Group Of Pictures (GOP) size of 24 the proposed method is also able to outperform interpolation-based system.

In the context of video-plus-depth coding, edge-aware intra coding is proposed. The system is tailored for edge block, and it tries to reproduce with a high degree of fidelity the discontinuity in the edge regions. This leads to rate savings and an improved quality of the synthesized view. For the synthesized view, the average Bjøntegaard bitrate saving is about 25%. To address the problem of video-plus-depth coding in a distributed manner, it is proposed to leverage the motion of the depth to improve the SI generation for the texture stream, and vice versa. Using depth frames to improve texture frames leads to Bjøntegaard PSNR gains up to 1.48 dB on the WZ frames, GOP 2. Using texture frames to improve the distributed decoding of depth frames, leads to Bjøntegaard PSNR improvements up to 4.95 dB on the WZ frames, GOP 8.

The last contribution of the thesis is the development of Rate Adap-

tive (RA) BCH (Bose-Chaudhuri-Hocquenghem) codes for DSC is the high correlation scenario, for short block length. Shorter block lengths are of interest in case of low-latency systems or adaptive DSC. The proposed RA BCH showed improved performance when compared with Low Density Parity Check Accumulate (LDPCA) and Turbo codes, which are commonly used in DVC literature, for $H(X|Y) \leq 0.25$. Hierarchical CRC check for improved reliability and mathematical model to predict the behavior of the code are proposed. Lastly various techniques to check the result are tested and compared, and it is demonstrated that in the tested scenario the use of extra syndromes for checking leads to improved performance because the checking strength can be adapted to the estimated reliability of the decoded result. The model is able to provide a reliable estimation of rate and Bit Error Rate (BER) performance. The maximum prediction error for the rate is 0.04%, for what concerns the BER the model is able to provide precise enough estimations. For very low values of p the rate gain is consistent: for $p = 0.005$ the proposed RA BCH codes require half of the rate required by conventional LDPCA codes.

Future Work and Discussion

During the development of DVC and DSC many coding tools were proposed to address the performance gap between distributed and predictive coding solutions. These efforts greatly increased the RD performance of the system. Nevertheless a significant performance gap is still present when comparing DVC-based codecs with predictive-based ones. New coding tools need to be developed to address this issue, which is going to be even more complex now, with the release of the new HEVC standard. Improved performance can be achieved for example improving the performance of the employed RA codes, generating codes specifically tailored for particular conditions of the virtual channel. In the context of M-DVC a limited amount of inter-camera communication could be introduced: allowing the cameras to help the decoder in the inter-view SI generation process and/or in the SI fusion process. The presence of a feedback channel is also a problem for many applications. Given that the motion content of different views is similar, one may think to use lateral views to predict the rate at the decoder, communicate such prediction at the encoder and use it to reduce the number of transmissions.

More advanced inter-view SI generation methods, targeting non-rectified views should be proposed. DVC targets low-complexity and inexpensive encoders, therefore problems such as: lack of synchronization between frames, delays, artifacts, difference in color should be taken into account. Heterogeneous networks with cameras having different resolutions should be also investigated.

Finally, it should be noticed that DVC forced the research community to look at the video coding problem in a new light, leading to the development of new approaches, some of them are already used in predictive coding solutions, where the decoder complexity is increased to achieve higher error resiliency, or where the efforts made to create efficient motion estimation systems in DVC are leveraged to improve the performance of H.264/AVC. Maybe some of the techniques developed in DVC can be used to limit or avoid inter-camera communication, still allowing to leverage inter-camera redundancy, while using predictive coding for leveraging temporal redundancy.

Therefore the studies on DVC are still relevant, both as a parallel and competing field to predictive coding, but also, a synergy between these two approaches can lead to new solutions and techniques in video coding.

Appendices

Appendix A

Ph.D. Publications

Multi-hypothesis Distributed Stereo Video Coding

Matteo Salmistraro [1], Marco Zamarin [2], Søren Forchhammer [3]

DTU Fotonik, Technical University of Denmark
Ørsteds Plads, 2800 Kgs. Lyngby, Denmark.
[1] matsl@fotonik.dtu.dk
[2] mzam@fotonik.dtu.dk
[3] sofo@fotonik.dtu.dk

***Abstract*—Distributed Video Coding (DVC) is a video coding paradigm that exploits the source statistics at the decoder based on the availability of the Side Information (SI). Stereo sequences are constituted by two views to give the user an illusion of depth. In this paper, we present a DVC decoder for stereo sequences, exploiting an interpolated intra-view SI and two inter-view SIs. The quality of the SI has a major impact on the DVC Rate-Distortion (RD) performance. As the inter-view SIs individually present lower RD performance compared with the intra-view SI, we propose multi-hypothesis decoding for robust fusion and improved performance. Compared with a state-of-the-art single side information solution, the proposed DVC decoder improves the RD performance for all the chosen test sequences by up to 0.8 dB. The proposed multi-hypothesis decoder showed higher robustness compared with other fusion techniques.**

***Index Terms*—Distributed Video Coding, Stereo Video Coding, Multiview Video Coding, Multi-hypothesis**

I. INTRODUCTION

In recent years DVC has received great attention: the possibility of exploiting the temporal redundancy of a video signal at the decoder rather than at the encoder is appealing for applications like mobile video coding, sensor networks and video surveillance. DVC is based on two information theory results: the Slepian-Wolf [1] and the Wyner-Ziv (WZ) [2] theorems where, in the second case, source data are independently lossy coded but jointly decoded using a correlated source at the decoder, which is commonly referred to as Side Information (SI).

A Stereo sequence is made of two dependent streams, and the disparity between two frames allows stereoscopic displays to provide a depth illusion to the user. The spatio-temporal redundancy between the streams can be used in order to achieve better RD performance. DVC may enable the creation of very low-complexity and low-cost encoders, leading to the development of dense networks of encoders communicating with a central decoder. In this scenario the ability to exploit the redundancy between cameras capturing the same scene (inter-view redundancy) without inter-camera communication is also appealing, leading to many investigations on Multiview DVC (M-DVC) [3], [4]. A particular M-DVC scenario is the Stereo scenario, in which we are interested in exploiting the inter-view redundancy between the two available views. This could be interesting in order to shift the complexity from encoder to decoder while continuing to leverage inter-view redundancy. Some works have addressed the Stereo DVC scenario [5][6], where an inter-view SI has been fused with a temporal interpolated SI. In [5] the disparity between a couple of frames belonging to different views has been estimated and applied to the next temporal frame in order to predict the corresponding one in the other view. The work demonstrated that improvements are possible despite the use of only two views. In [6] Disparity-Guided Temporal Interpolation (DGTI) is used in order to produce the SI, and a Multi-Hypothesis Based Correlation Model (MHBCM) decoder is employed in order to fuse it with a temporal interpolated SI.

Different versions of the proposed inter-view side information generation systems have been presented in [7]. They were used in combination with extrapolation but no on-line residual estimator was proposed, as the work assessed the quality of the SI using off-line residuals. In this work we improve these methods and propose methods for on-line residual estimation, enabling the calculation of the correlation noise distribution. We also address the SI fusion problem.

The paper is organized as follows. In Section II we introduce the basic DVC codec on which our system is built. In Section III we describe the various SI generation methods employed in our system. Finally, we discuss our results in Section IV.

II. DISTRIBUTED VIDEO CODING SYSTEM

The proposed decoder uses, as basis, the monoview single SI codec presented in [8], depicted in Fig. 1. The encoder divides the frames into Key Frames (KFs) and WZ frames. The first ones are encoded independently with respect to each other and with respect to the WZ frames, using H.264/AVC in Intra Mode. The KFs are decoded at the decoder and used to calculate the SI, which is the estimation of the to-be-decoded WZ frame. At the encoder, WZ frames are transformed using a 4×4 integer DCT transform, quantized and organized in bitplanes. Bitplanes are then fed into a Low-Density Parity Check Accumulate (LDPCA) encoder [9], which calculates the parity bits (syndromes). A subset of the syndromes are sent to the decoder and used to correct the errors made in the prediction of the WZ frame. If the amount of bits is not enough to successfully decode, new bits are requested until the solution satisfies the syndrome condition and an 8 bit CRC check. The LDPCA decoder requires: the syndromes from

MMSP'13, Sept. 30 - Oct. 2, 2013, Pula (Sardinia), Italy.
978-1-4799-0125-8/13/$31.00

M. Salmistraro, M. Zamarin, S. Forchhammer, "Multi-hypothesis Distributed Stereo Video Coding", *Proc. of 2013 IEEE Int'l Work. on Multimedia Signal Processing (MMSP 2013)*, pp. 093 - 098, Pula, Italy, Sep. 30 - Oct. 2, 2013.

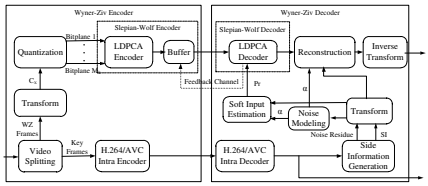


Fig. 1. Monoview single SI DVC codec [8] used as basis.

the encoder, the systematic part of the bitplane (the estimated bitplane obtained from the SI), and the residual estimation. The residual is the difference between the SI frame and the original WZ frame. Since the decoder has no access to the WZ frame, the residual must be estimated. The residual estimation has a high impact on the RD performance of the system.

III. SIDE INFORMATION GENERATION AND FUSION

In a stereo video stream we shall distinguish two views: the *master view*, to which the to-be-decoded WZ frame belongs, and the *support view* (the other view), which is used to improve the decoding performance of the master view. It has to be noted that the two roles can be decided on a frame-by-frame basis. By allowing to switch the roles of each view for every new WZ frame, the first view can leverage the other and vice versa, depending on the location of the WZ frame. Given a frame in the master view at instant t which has to be decoded, denoted as $I_{m,t}$, we suppose to have access to the frames at instants $t-1$ and $t+1$, denoted as $I_{m,t-1}$ and $I_{m,t+1}$, respectively, in order to generate the temporal interpolated SI using Overlapped Block Motion Compensation (OBMC) [8]. In order to generate the inter-view SI, we assume we have access to the frames $t-1$ and t in the support view, denoted as $I_{s,t-1}$ and $I_{s,t}$, respectively. In order to have a more flexible decoder requiring less strict synchronization between views, we avoid using $I_{s,t+1}$, which denotes the frame at instant $t+1$ in the support view stream. In this section, we describe two inter-view SI generation methods suitable for our scenario: Difference Projection (DP) and Motion Vector Similarity (MVSim). Compared with other approaches [6], these approaches may allow each view to exploit the other, since the roles of master and support can be swapped.

A. Difference Projection

The basic idea of DP [7], [10] is to use the estimated disparity between two available frames at $t-1$ to project the difference between the frames at instant $t-1$ and t in the support view towards the master view. The disparity estimation algorithm employed in this work is pixel-based using a 5×5 block surrounding the pixel, and with a search window of $[-26, 26]$ pixels in the horizontal direction. The search aims at finding the lowest Mean Absolute Difference between source and destination blocks. The dimension of the block has been chosen as a compromise between flexibility and robustness. The disparity is calculated with pixel accuracy and then smoothed using a median filter. The difference in the support view is calculated as follows:

$$\delta_s(x,y) = I_{s,t-1}(x,y) - I_{s,t}(x,y), \quad \forall (x,y). \tag{1}$$

By using the iterative procedure described in [10] it is possible to calculate a threshold T and define the change detection map M_s for the support view:

$$M_s(x,y) = \begin{cases} 1 & \text{if } |\delta_s(x,y)| \geq T \ , \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

The purpose of the change detection map is to find the pixels for which the activity (i.e. motion) is high enough to justify the application of the algorithm. The application of the projection to a pixel could lead to problems if the disparity estimation is inaccurate or if we are in the presence of occlusions or disocclusions, because applying the wrong difference to a pixel may lead to errors and hence artefacts in the SI. Hence, it is a risky procedure and it has to be applied only to the pixels having a difference between the two consecutive frames high enough to justify the risk. Thereafter, the change detection map can be warped using the disparity field d calculated from the support to the master view:

$$M_m(x - d(x,y), y) = M_s(x,y) \tag{3}$$

and also the difference can be warped:

$$\delta_m^{(1)}(x - d(x,y), y) = \begin{cases} \delta_s(x,y) & \text{if } M_s(x,y) = 1 \ , \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

When calculating a disparity (or motion) vector, we have a *source* point (x,y) and a *destination* point $(x - d(x,y), y)$; in this case we define as *source* the frame $I_{s,t-1}$. Since the source of the disparity field is the support view, the disparity has to be calculated only for the pixels where $M_s(x,y) = 1$, leading to a complexity reduction. This warping generates cracks and isolated pixels in the warped difference creating, in turn, the same artefacts in the SI. These artefacts have repercussions on the high-frequency bands of the DCT employed by the encoder. The difference is hence post-processed by deleting isolated pixels and filling the empty pixels surrounded by warped pixels using linear interpolation. The same process is applied to M_m: isolated pixels are removed and the cracks filled. Then we can calculate a part of the SI:

$$Y_{DP1} = I_{m,t-1} + \delta_m^{(1)}. \tag{5}$$

Inspired by the work in [3], we also employ another complementary method: which uses the disparity estimation having source in the master view, calculated only for the pixels satisfying $M_m(x,y) = 1$ to reduce the complexity. After this, the new field is calculated and δ_s is warped again obtaining $\delta_m^{(2)}$ which is used to calculate

$$Y_{DP2} = I_{m,t-1} + \delta_m^{(2)}. \tag{6}$$

In the end, we can calculate the difference projected SI:

$$Y_{DP} = \frac{1}{2}\left(Y_{DP1} + Y_{DP2}\right), \tag{7}$$

and also the residual estimation can be calculated:

$$R_{DP}(x,y) = \begin{cases} \frac{1}{2} R_A(x,y) \text{ if } M_m(x,y) = 1 \ , \\ \frac{1}{2}\left|I_{m,t-1}(x,y) - I_{m,t+1}(x,y)\right| \text{ otherwise,} \end{cases} \tag{8}$$

where

$$R_A(x,y) = \left(\left|\delta_m^{(1)}(x,y)\right| + \left|\delta_m^{(2)}(x,y)\right|\right). \tag{9}$$

The idea behind the calculation of the residual is that the higher the difference the higher the motion and thus the higher the probability of introducing an error. We take the absolute value of the two elements in order to avoid underestimation of the movement, due to opposite signs of the two differences for the same pixels. An issue with this approach is that the residual is 0 outside the area in which $M_m(x,y) = 1$. This underestimation of the residual needs to be improved with an approximation in the area in which $M_m(x,y) = 0$. Since the motion in this area is very low, we can employ as estimation half of the absolute difference of the two frames [11], therefore assuming negligible motion. In order to validate the proposed method we compared the performance difference between a single SI DP-based decoder and a single SI Extrapolation-based decoder, both employing off-line residuals. Then, we made the same comparison between the same two decoders using on-line residuals. The performance drop in PSNR is, on average, only 5%.

The method we present requires two partial disparity estimations: the first is from $I_{s,t-1}$ to $I_{m,t-1}$ only for the pixel where $M_s = 1$. This leads to the calculation of M_m and Y_{DP1}. The second is carried out after the first one, from $I_{m,t-1}$ to $I_{s,t-1}$ for the pixel where $M_m = 1$. Y_{DP1} has a similar quality to Y_{DP2} thanks to the linear interpolation, and their average is superior to both the two components.

B. Motion Vector Similarity

Methods similar to MVSim have been proposed in many works, e.g. [7], [3]. With MVSim the motion field in the support view is warped towards the master view. The main idea behind it is similar to the one of DGTI [6], but it does not require $I_{s,t+1}$. We start with the disparity estimation between $I_{s,t-1}$ and $I_{m,t-1}$ in both directions (from support to master view, and vice versa). The disparity fields are denoted as $d^{(1)}$ and $d^{(2)}$ respectively. The estimation is carried out like in DP but for the full frame using a $[-26, 26]$ pixels search window in the horizontal direction and a $[-4, 4]$ pixels search window in the vertical direction. The parameters are different compared with DP as the disparity estimation is carried out for the whole frame. The motion field f between $I_{s,t-1}$ and $I_{s,t}$ is also calculated on a pixel basis: we use a 9×9 block $\mathcal{W}$ centred in the pixel, the search window is $[-10, 10] \times [-10, 10]$:

$$f(x,y) = \arg \min_{(f_1,f_2)} \sum_{(i,j)\in\mathcal{W}} |I_{s,t-1}(i,j) - I_{s,t}(i-f_1, j-f_2)|, \tag{10}$$

therefore the motion vectors have origin in $I_{s,t-1}$. The flow can be warped towards the master view using $d^{(1)}$ or $d^{(2)}$ obtaining two different warped motion fields having origin in $I_{m,t-1}$. These two fields are used to estimate $I_{m,t}$ by warping $I_{m,t-1}$. The two calculated terms of the SI are denoted as Y_{MVSim1} and Y_{MVSim2}. Holes in the two terms are filled using linear interpolation. The final SI can be calculated as:

$$Y_{MVSim} = \frac{1}{2}\left(Y_{MVSim1} + Y_{MVSim2}\right), \tag{11}$$

and the residual can be calculated as the difference between the two SIs, as it is done when dealing with optical flows [12]:

$$R_{MVSim} = |Y_{MVSim1} - Y_{MVSim2}|. \tag{12}$$

We chose to use a motion field having origin in $I_{s,t-1}$ in order to have no holes in the motion field during the warping, leading to a more stable result. Even if the quality of Y_{MVSim1} could be inferior to the quality of Y_{MVSim2} due to the presence of more holes, the linear interpolation allows good performance of Y_{MVSim1}, and Y_{MVSim} shows better performance compared with its two components.

This method requires two full frame disparity estimations and a motion estimation, while DP requires only two partial frame disparity estimations.

C. SI fusion

Once the temporal interpolated SI and one inter-view SI are available, the main problem is to fuse these two SIs in an efficient manner. Usually, the inter-view SI has lower RD-performance compared with the temporal one, hence a careful fusion should be done between the two SIs in order to exploit the best part of both. Since we are using only two frames in the support view the quality of the inter-view SI is much lower compared with the OBMC SI. Hence a robust way to fuse the two SIs is needed. In order to fuse the two SIs, we propose to use the Multi-Hypothesis decoder, using the 2 SIs approach (2SI) presented in [12], where the decoder was used to robustly fuse an Interpolated SI and an Optical Flow-based SI. In that work, it is demonstrated that this decoder is able to successfully fuse two SIs despite the difference in RD performance. Here we use the same idea in order to fuse the temporal SI and the inter-view SI.

The basic idea behind the 2SI decoder is to fuse two different observations of a given signal obtaining various candidates for decoding. The decoder then performs a rate-based optimization choosing the candidate which decodes first, leading to the lowest rate. This system employs 6 parallel LDPCA decoders [12]. The previously introduced residual (R_{DP} or R_{MVSim}) is central in order to estimate the Laplacian distribution of the DCT coefficients $l_{X|Y}$ given a SI (in this case denoted as Y). We employ the noise estimation method outlined in [8] without cross-noise refinement. The

2SI decoder (depicted in Fig. 2) does a weighted average of the two distributions, obtaining six combined distributions F_i, $i = 1, \ldots, 6$. If DP is used as second SI, the combined distribution is:

$$F_i = w_i l_{X|Y_{OBMC}} + (1 - w_i) l_{X|Y_{DP}}. \quad (13)$$

Six different weights $w_i \in \{1, 0.8, 0.6, 0.4, 0.2, 0\}$ are used by the system to produce different linear combinations of the two available observations of the signal. The combined distribution is only used for the unreliable DCT coefficients of the frame (belonging to the set $umap$), while in the reliable part only Y_{OBMC} is used. The decision on the reliability is taken examining the estimated distribution of the coefficient [12]. The soft input for the k-th systematic bit of the i-th LDPCA decoder is calculated as:

$$p_k^{(i)} = \begin{cases} P(b_k = 0 | Y_{OBMC}, Y_{DP}, F_i, b^-) \text{ if } k \in umap, \\ P(b_k = 0 | Y_{OBMC}, b^-) \text{ otherwise,} \end{cases} \quad (14)$$

where b^- is the information from previously decoded bitplanes, b_k is the k-th bit of the current bitplane and $p_k^{(i)}$ is its corresponding conditional probability. Each decoder is also fed with the syndromes coming from the encoder. After receiving a chunk of parity bits, the different decoders try to decode the bitplane. If the z-th decoder, using the weight w_z, succeeds its result is taken as the decoded bitplane, and the corresponding linear combination of the distributions F_z is used in the reconstruction of the coefficient. If none of the decoders succeeds other syndromes are requested.

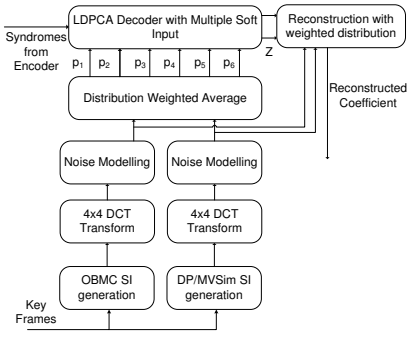


Fig. 2. Multi-hypothesis Decoder.

In order to assess the performance of the fusion method, we compare it with the ideal fusion (IF) [5]. In the ideal fusion we suppose to know the WZ frame at the decoder and fuse the two SIs to get the lower mean-square error between fused SI and original frame. This is obviously an unrealistic assumption since the original frame cannot be known at the decoder but it is useful to assess the performance of the proposed fusion system. It has also to be noted that this method does not create the best possible SI, since it is not a full RD optimization but only a distortion-based optimization. However, it is still a reasonable way of assessing the quality of the produced SIs.

Obviously, the 2SI decoder has higher complexity compared with a single SI decoder (up to 6 times), hence classical fusion techniques are also appealing in order to reduce the computational complexity of the decoder. We have examined the Motion and Disparity Compensated Difference Linear fusion (MDCD-Lin) approach [13] and applied it to DP and OBMC generated SIs. The MDCD-Lin approach estimates the fused SI Y_F taking into account the residuals. MDCD-Lin has been proposed to address a standard multiview scenario, where more than one support view is available and the RD performance of the inter-view SI is usually higher. In our case we need high robustness, because an incorrect fusion would lead to great loss, since the difference in RD performance of the two SIs is relevant. Therefore we use as weight the average of the residual over the 4×4 DCT block to which the point (x, y) belongs. Residuals are denoted as $\overline{R}_{DP}(x, y)$ and $\overline{R}_{OBMC}(x, y)$ for DP and OBMC respectively.

$$Y_F(x,y) = \frac{Y_{OBMC}(x,y)\overline{R}_{DP}(x,y)}{\overline{R}_{OBMC}(x,y) + \overline{R}_{DP}(x,y)} + \\ + \frac{Y_{DP}(x,y)\overline{R}_{OBMC}(x,y)}{\overline{R}_{OBMC}(x,y) + \overline{R}_{DP}(x,y)}. \quad (15)$$

The MHBCM multi-hypothesis decoder has also been tested. MHBCM uses only one LDCPA decoder while the 2SI approach uses six LDPCA decoders in parallel. The key difference between MHBCM and usual fusion techniques is that MHBCM fuses the distributions in a similar way compared with the 2SI approach, but the weights are calculated accordingly to the estimated Laplacian parameter of the distribution and following a clustering approach [6].

The 2SI method does not require a training step, therefore we compared its performance with other methods without training.

IV. EXPERIMENTAL RESULTS

In this section we assess the performance of the proposed methods. We use the sequences IU, AC, IUJW, and VK available at [14]. The sequences are stereoscopic, with a resolution of 320×240 pixels, at 15 fps. All the sequences represent a typical stereo video conference, with a person (or two people in the case of IUJW) moving with different patterns, we have slow motion in IU, faster motion and disparity changes in AC and VK. In IUJW we have two people moving on the foreground, leading to a big foreground object presenting independent motion patterns between the two constituting elements. IU and IUJW present also people moving in the background, entering and exiting the scene. We are addressing stereoscopic coding for 3D video visualization, hence we use a dataset devised exclusively for this purpose, and used in error concealment for stereo video coding [10]. The quantization matrices used for the WZ frames are the matrices having index $Qi = 1, 4, 7, 8$ of the DISCOVER

project [15], [16]. KFs are encoded using H.264/AVC Intra mode, the quantization parameters used have been chosen in order to provide similar quality between KFs and WZ frames. We present the results for Group-of-pictures of size 2 (GOP2), for WZ frames only. The Intra results refer to coding the WZ frames using the H.264/AVC Intra mode, using the same settings used for the KFs. As master view we use the right view, and as support view the left one, which is Intra coded using the setting already used for the KFs. Only the Luminance component has been evaluated. In Table I we report the Bjøntegaard differences (BD) [17] between the decoder in [8] and our 2SI solution. We can see that the proposed system improves a state-of-the-art single SI decoder using, as benchmark, one of the best single SI DVC decoder available, which outperforms the reference DISCOVER codec as shown in [8]. The MVSim and DP methods are able to achieve similar results outperforming the single SI system. Their performance is close, but DP is less complex from a computational point of view since the motion field calculation is not needed, and the disparity has to be estimated only for a subset of the pixels, as can we see from Table II. For what concerns the RD performance of the SIs presented in [7] compared with their improved versions, the BD PSNR improvement is up to 0.4 dB when comparing single SI decoders using off-line residuals. In the case of 2SI fusion, the biggest difference between the two proposed techniques, in terms of PSNR RD performance difference, can be found for the IUJW sequence where DP outperforms MVSim by 0.21 dB in BD: in Table I, DP achieves 0.80 dB of improvement (corresponding to 0.33 dB of improvement on all the frames) while MVSim improves of only 0.59 dB (corresponding to 0.25 dB of improvement on all the frames). Secondly, the 2SI decoder is able to fuse the two SIs correctly, outperforming the single SI system in every RD-point. The MHBCM (Table III) does not achieve neither the performance of the 2SI decoder, nor its stability, making it not suitable as the difference between the two SIs is high. In our case the average BD between the single SI OBMC and the single SI DP ranges between 3.5 and 5 dB. It can be also noted that the SI generation method in [6] (DGTI) requires higher frame synchronization, since three frames from the support view ($I_{s,t+1}, I_{s,t}, I_{s,t-1}$) and two from the master view ($I_{m,t+1}$ and $I_{m,t-1}$) must be received and decoded before the WZ decoding of the current frame. Our system requires 4 frames: $I_{s,t-1}, I_{s,t}, I_{m,t-1}, I_{m,t+1}$. This is advantageous in case of real-time, low-delay systems, where good inter-camera synchronization is challenging to obtain. Finally, it can be seen in Table III, that standard fusion techniques like MDCD-Lin are unable to fuse the SIs correctly (sequences AC and VK) or improvements are very low compared with the 2SI approach. The superiority, in a DVC setting, of the 2SI approach over other SI fusion techniques resides in two factors: the first one is the rate-based optimization, since the decoder chooses the fastest converging solution among the presented hypothesis. Secondly, the weights do not depend on the residuals, making the system less flexible but more stable. The RD-curves are reported in Figures 3 to 6, where the benchmark decoder [8] is denoted as OBMC and the MHBCM is used to fuse OBMC and DP. We also present the performance of an IF-based system in Table IV as a term of reference.

TABLE I
BD WHEN COMPARED WITH [8].

	2SI OBMC+DP		2SI OBMC+MVSim	
Seq.	$\Delta Rate$	$\Delta PSNR$	$\Delta Rate$	$\Delta PSNR$
IU	-12.09%	0.67	-10.61%	0.59
AC	-9.16%	0.58	-6.92%	0.44
IUJW	-13.75%	0.80	-10.49%	0.59
VK	-9.21%	0.58	-8.67%	0.55

TABLE II
AVERAGE PERCENTAGE OF PIXELS FOR WHICH THE DISPARITY HAS TO BE CALCULATED WHEN USING DP, FOR $Qi = 8$.

Sequence	Pixels Percentage
IU	10.98%
AC	9.70%
IUJW	13.03%
VK	11.39%

TABLE III
BD FOR THE OTHER TWO FUSION TECHNIQUES, WHEN COMPARED WITH [8].

	MDCD-Lin OBMC+DP		MHBCM OBMC+DP	
Seq.	$\Delta Rate$	$\Delta PSNR$	$\Delta Rate$	$\Delta PSNR$
IU	-3.75%	0.20	-4.31%	0.23
AC	2.37%	-0.14	4.37%	-0.25
IUJW	-1.57%	0.09	-4.52%	0.25
VK	1.92%	-0.11	-0.73%	0.05

TABLE IV
BD IN THE CASE OF IF, WHEN COMPARED WITH [8].

	IF OBMC+DP		IF OBMC+MVSim	
Seq.	$\Delta Rate$	$\Delta PSNR$	$\Delta Rate$	$\Delta PSNR$
IU	-20.17%	1.12	-17.15%	0.94
AC	-12.71%	0.81	-8.90%	0.56
IUJW	-19.11%	1.07	-16.79%	0.92
VK	-13.29%	0.83	-14.74%	0.92

V. CONCLUSION

In this paper we present an improved version of the Difference Projection stereo SI generation method for DVC, along with an improved version of the Motion Vector Similarity method. We also describe a residual estimation method for the proposed SIs. Both SIs have been fused with an OBMC based SI, improving a state-of-the-art single SI DVC codec by up to 0.8 dB in BD. We have assessed the robustness of the Multi-hypothesis decoder when employing two SIs with different RD-performance, demonstrating the validity of this approach in a stereo scenario. We have also compared it against a well-known SI fusion technique, demonstrating its superiority in the chosen scenario. We have shown that DP can compete with the more complex MVSim based systems, which have been used

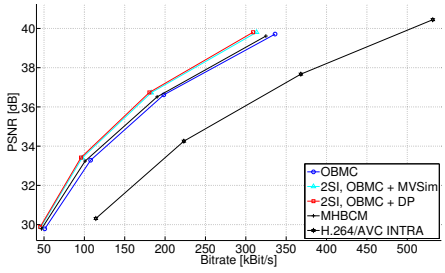


Fig. 3. IU Right sequence, WZ frames only, 15 fps.

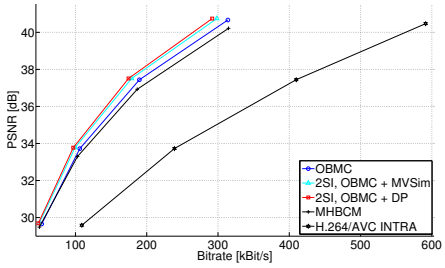


Fig. 4. AC Right sequence, WZ frames only, 15 fps.

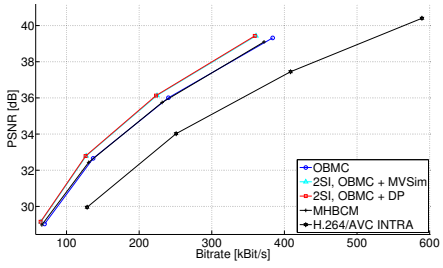


Fig. 5. VK Right sequence, WZ frames only, 15 fps.

in literature so far [3]. The proposed method requires only the estimation of two partial pixel-based disparities. We have also shown the possibility of achieving good results requiring only two frames in the support view, allowing the use of more flexible coding solutions. Also a single LDPCA decoder multi-hypothesis solution proved to be not robust enough in the studied situation. Future work will aim at extending the presented scenario to non-stereo and non-rectified sequences and improve the noise model for both the two inter-view SI

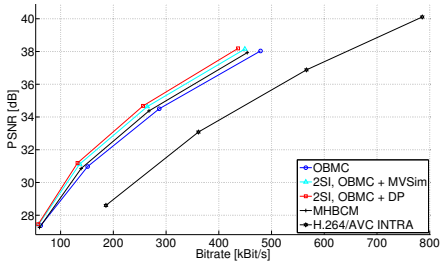


Fig. 6. IUJW Right sequence, WZ frames only, 15 fps.

generation systems.

REFERENCES

[1] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. 19, no. 4, pp. 471–480, July 1973.

[2] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, vol. 22, pp. 1–10, 1976.

[3] M. Ouaret, F. Dufaux, and T. Ebrahimi, "Iterative multiview side information for enhanced reconstruction in distributed video coding," *J. Image Video Process.*, vol. 2009, pp. 3:1–3:17, January 2009.

[4] C. Guillemot, F. Pereira, L. Torres, T. Ebrahimi, R. Leonardi, and J. Ostermann, "Distributed monoview and multiview video coding," *Signal Processing Magazine, IEEE*, vol. 24, no. 5, pp. 67–76, 2007.

[5] J. Areia, J. Ascenso, C. Brites, and F. Pereira, "Wyner-Ziv stereo video coding using a side information fusion approach," in *MMSP 2007*, October 2007, pp. 453–456.

[6] Y. Li, H. Liu, X. Liu, S. Ma, D. Zhao, and W. Gao, "Multi-hypothesis based multi-view distributed video coding," in *PCS 2009*, May 2009, pp. 1–4.

[7] M. Salmistraro and S. Forchhammer, "Stereo side information generation in low-delay distributed stereo video coding," *Progress in Biomedical Optics and Imaging*, vol. 8499, 2012.

[8] X. Huang and S. Forchhammer, "Cross-band noise model refinement for transform domain Wyner-Ziv video coding," *Signal Processing: Image Communication*, vol. 27, no. 1, pp. 16–30, 2012.

[9] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive codes for distributed source coding," *EURASIP Signal Processing Journal*, vol. 86, no. 11, pp. 3123–3130, November 2006.

[10] Y. Chen, C. Cai, and K.-K. Ma, "Stereoscopic video error concealment for missing frame recovery using disparity-based frame difference projection," in *ICIP 2009*, November 2009, pp. 4289–4292.

[11] C. Brites, J. Ascenso, and F. Pereira, "Learning based decoding approach for improved Wyner-Ziv video coding," in *PCS 2012*, May 2012, pp. 165–168.

[12] X. Huang, L. Raket, H. V. Luong, M. Nielsen, F. Lauze, and S. Forchhammer, "Multi-hypothesis transform domain Wyner-Ziv video coding including optical flow," in *MMSP 2011*, October 2011, pp. 1–6.

[13] T. Maugey, W. Miled, M. Cagnazzo, and B. Pesquet-Popescu, "Fusion schemes for multiview distributed video coding," in *European Signal Processing Conference*, vol. 1, Glasgow, Scotland, 2009, pp. 559–563.

[14] Microsoft research, database of stereo video sequences. [Online]. Available: http://research.microsoft.com/en-us/projects/i2i/data.aspx

[15] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: Architecture, techniques and evaluation," *Proc. of PCS*, November 2007.

[16] (2007, December) DISCOVER project test conditions. [Online]. Available: http://www.img.lx.it.pt/~discover/test_conditions.html

[17] G. Bjøntegaard, "Calculation of average PSNR differences between RD curves," in *ITU-T Q6/SG16, Doc. VCEG-M33, in: 13th Meeting, Austin, USA, April*, 2001.

A robust fusion method for multiview distributed video coding

***Matteo Salmistraro*[†], *João Ascenso*[‡], *Catarina Brites*[‡], *Søren Forchhammer**[†]**

[†]*DTU Fotonik, Technical University of Denmark, matsl@fotonik.dtu.dk,*

*sofo@fotonik.dtu.dk (*corresponding author)*

‡*Instituto Superior Técnico, Portugal, {joao.ascenso, catarina.brites}@lx.it.pt*

ABSTRACT

Distributed Video Coding (DVC) is a coding paradigm which exploits the redundancy of the source (video) at the decoder side, as opposed to predictive coding, where the encoder leverages the redundancy. To exploit the correlation between views, multiview predictive video codecs require the encoder to have the various views available simultaneously. However, in Multiview-DVC (M-DVC) the decoder can exploit the redundancy between views, avoiding the need for inter-camera communication. The key element of every DVC decoder is the Side Information (SI), which can be generated leveraging intra-view or inter-view redundancy in the video. In this paper, a novel fusion technique is proposed, which is able to robustly fuse an inter-view SI and an intra-view (temporal) SI. An inter-view SI generation method capable of identifying occluded areas is proposed and is coupled with a robust fusion system, able to improve the quality of the fused SI along the decoding process through a learning process performed on already decoded data. It is here proposed to fuse the estimated distributions of the SIs as opposed to conventional fusion algorithm relying on the fusion of pixel values. The proposed solution is able to achieve gains up to 0.9 dB in Bjøntegaard difference when compared with the best performing (in a RD sense) single SI DVC decoder, chosen between an inter-view SI-based decoder and a temporal one.

Keywords: Distributed video coding; Multiview video coding; Side information fusion; Learning

M. Salmistraro, J. Ascenso, C. Brites, S. Forchhammer, "A Robust Fusion Method for Multiview Distributed Video Coding", *EURASIP Journal on Advances in Signal Processing*, (submitted).

1 INTRODUCTION

Distributed Video Coding (DVC) [1]–[3] is a coding paradigm based on the theoretical results of Distributed Source Coding (DSC): the Slepian-Wolf [4], and the Wyner-Ziv (WZ) theorems [5]. These foundations establish a different way to compress information, namely by independently coding the source data but jointly decoding it. Thus, in DVC, the source correlation is exploited at the decoder, as opposed to the widely adopted predictive coding solutions where the encoder is responsible for exploiting all the correlation. One of the key blocks of every DVC decoder is the Side Information (SI) generation module which estimates the WZ frame. Typically, in monoview systems, the SI creation exploits temporal redundancy, by making assumptions of the apparent motion in a video stream, e.g. linear motion between reference frames is assumed [6]. Thus, the SI errors must be corrected, which requires the transmission of parity bits (or syndromes) from the encoder, and the use of error correcting codes. The channel decoder requires soft inputs for the SI bitplanes, which can be calculated from the residual. In DVC, an on-line residual is used: it is obtained by estimation of the residual, without using the original WZ frame.

An efficient DVC system must be able to minimize the amount of data sent from the encoder. Therefore, the quality of the SI has high importance for the Rate-Distortion (RD) performance of the DVC decoder; in fact, having a high quality SI, characterized by few errors, allows the transmission of less error correcting data (requiring a lower bitrate) and enables improving the decoded WZ frame quality.

In monoview DVC codecs, every frame is independently processed without any reference to other decoded frames. This allows lowering the encoding complexity since the complex task of exploiting the temporal correlation is left to the decoder. When different views of the same visual scene are coded in different camera nodes, e.g. in visual sensors networks, inter-view coding can further improve the coding performance, exploiting inter-camera redundancy. If a predictive multiview video codec is used, e.g. Multiview Video Coding (MVC) [7] inter-camera communication is needed. MVC relies on the same coding tools used in H.264/AVC: frames belonging to other views are inserted in the reference picture lists and used for disparity compensation. This approach requires inter-camera communication to enable one camera to use frames of another camera for disparity compensation.

On the other hand, in DVC solutions for the multiview scenario, each camera can independently code the frames, relying on the decoder to exploit the correlation between views [8], [9]. Typically, the Multiview-DVC (M-DVC) decoder tries to exploit, at the same time, temporal intra-view and inter-view correlation, generating two SI frames: 1) temporal SI, by means of motion estimation and interpolation, e.g. employing Overlapped Block Motion Compensation (OBMC) [6] and 2) inter-view SI, generated leveraging the inter-view redundancy [3]. To exploit the best part of each estimated SI frame, it is necessary to fuse the frames, choosing the best regions of each estimated SI frame to create a final SI frame that is used for decoding [8], [9]. The regions are chosen according to an estimation of their quality. SI frame fusion is a hard problem, and there are many fusion techniques available in the literature [8] with various degree of efficiency. The goal of an efficient frame fusion technique is to deliver an RD performance better than the best performing single SI decoder out of the one using the inter-view SI and the one using the temporal SI. In general, the bigger the difference in RD performance between the SIs, the harder the fusion task is, because, fusing incorrectly a region of the frame, may generate consistent loss in RD performance.

The main contributions of this work are: 1) OBDC (Overlapped Block Disparity Compensation) a novel inter-view SI generation system is presented. It is able to cope with high camera distance and detect occlusions due to part of the scene outside the field of view of one camera. It is also able to adapt to unknown camera distances; 2) the fusion of the estimated distributions of the DCT coefficients of the SI; 3) a novel learning technique based on the refinement of the quality of the fused SI along the decoding process exploiting already decoded data. The fusion of distributions is here proposed as an alternative to the fusion of the SI frames. The use of distributions to estimate the reliability of the regions of the SI allows to exploit high-performance noise modelling algorithms developed in literature. The learning allows to correct wrong initial estimations of the quality of the SIs, leading to superior RD performance for the next steps of the decoding process.

This paper is structured as follows: Section 2 deals with related works and with pixel and block based fusion techniques. An overview of the decoder is given in Section 3. The novel fusion algorithm as well as the SI generation method are described in Section 4. In Section 5, the performance of the

proposed tools are assessed and compared with state-of-the-art distributed coding solutions, as well as monoview predictive codecs.

2 RELATED WORK

2.1 Interview SI creation

Disparity Compensation View Prediction (DCVP) [10] is one of the simplest inter-view SI generation techniques, where the same algorithm used for temporal interpolation is applied between adjacent views to perform disparity estimation and compensation. However, the DCVP SI quality deteriorates when the distance between views is increased. The majority of the studies proposed in literature focus on really close cameras, for example the distance between the cameras in [8] is 6.5 cm, and the problem of cameras moving with respect to each other is not addressed.

A different way of addressing the SI generation problem was proposed in [11], where MVME (MultiView Motion Estimation) was presented. The key idea of MVME is to estimate a single SI frame by jointly exploiting the motion of neighbouring views and projecting the motion field in the current view. MVME generates the SI in two separate steps: 1) motion estimation is performed on the available lateral (left and right) views, 2) motion compensation using the reference (decoded) frames in the view to decode (the central view). A fusion step is also performed in MVME, but it is needed to fuse various joint motion and disparity estimations, while in the previous cases the fusion was performed between a purely inter-view SI and a purely temporal one. MVME demonstrates high performance in fast motion sequences, but it is outperformed by motion compensation and interpolation techniques in slow motion cases [11]. More recently, in [12], a modified version of the temporal motion estimation algorithm employed in DISCOVER [13] is proposed for inter-view SI generation. The key novelty is the penalization of small disparities, which characterize background blocks.

2.2 SI Fusion Techniques

In recent years, SI fusion methods which use estimated distributions of the DCT coefficients were proposed for monoview DVC [14], [15] and applied to M-DVC [16], [17]. In [14] optimal reconstruction for a multi-hypothesis decoder was proposed. In [16] the authors enhanced [14], proposing a cluster-based noise modelling system and fusion. In [15], the concept of parallel decoding was introduced: the distributions of the available SIs were fused using different weights, generating, in the aforementioned case, six different fused distributions. From each fused distribution it is possible to calculate a set of conditional probabilities which are fed into six parallel LDPCA decoders. Thereafter, the decoders try to reconstruct the original bitplane considered in parallel, for each new chunk of received parity bits. The process stops when the bitplane is successfully decoded by at least one LDPCA decoder. The method proposed in [15] can be seen as a brute-force rate-based optimization approach but it suffers from high computational complexity; to perform an efficient SI fusion, several channel decoders need to be used. In [17], the method proposed in [15] was applied to stereo M-DVC to fuse an inter-view and temporal SI frames. Nevertheless, the issue related to the complexity of [15], was not addressed, since [17] still relies on parallel LDPCA decoding.

In M-DVC pixel and block-based fusion techniques are widely adopted [8], [9]. The results of [8], show that finding a fusion method, able to perform robustly for a wide range of different video sequences is difficult, in particular when the quality of the two SIs is very different and therefore the probability of making errors in the fusion process is high. A different approach for fusion in M-DVC is proposed in [9], where a past decoded WZ frame and its corresponding SI are used to train a Support Vector Machine classifier, which is then used to perform the fusion task, classifying the reliability of each pixel in the SIs. In [12] the fusion is performed according to an occlusion map: temporal SI is used if pixels belonging to the left or right views are estimated to be occluded. In [12] adaptive validation is also introduced: for a small subset of the WZ frames, parity bits are requested for correct inter-view and temporal SIs, introducing an overhead. If the two SIs require similar rates the fused SI is chosen, otherwise the single SI providing the lower rate is chosen.

The partially decoded information obtained during the decoding process can be used to enhance the RD performance of a DVC codec, by improving the correlation noise [6], [18], the SI [19] or, as it is proposed in this work, the fusion process in a multiview decoder.

In [20] first the WZ frame is decoded using either inter or intra-view SI, according to the video signal motion activity. Then the completely reconstructed WZ frame is used as basis for the generation of a refined SI, either disparity or motion compensation are used on a block basis. Lastly the refined SI is used in a new reconstruction step obtaining a higher quality reconstruction.

In [10] the encoder sends information to improve the fusion process: since the encoder has access to the original WZ frame and the KFs, a fusion mask can be generated, based on the difference between the KFs and WZ frame (both known at the encoder). The mask is then compressed and sent to the decoder to drive the fusion process. However, when the encoder participates in the fusion process its computational complexity is increased which may be impractical for some applications. In addition, the overhead can lead to a significant increase of the bitrate, which may severely limit the improvements obtained from having a higher quality SI fused frame.

2.3 Benchmarks for SI Fusion

In [8], [9], many fusion solutions are presented. The solutions employed as benchmark, in the result section, are described here. The original WZ frame is denoted as X. The SIs employed for fusion, in all the analysed benchmarks, are generated through OBMC and OBDC and denoted as Y_{OBMC} and Y_{OBDC} respectively. The corresponding estimated residuals are denoted as R_{OBMC} and R_{OBDC}.

Motion and Disparity Compensated Difference Linear fusion (MDCD-Lin) is a Multiview fusion technique [8] used as benchmark in [9], [12]. The techniques presented in [9] showed to perform either as well as MDCD-Lin or as well as the best single SI decoder, therefore MDCD-Lin and two single SI decoders are employed as benchmarks. MDCD-Lin fuses pixel values, using the estimated residuals as weights for generating the fused SI, for the pixel having position $\boldsymbol{x}$. The weight is calculated as:

$$w(\boldsymbol{x}) = \frac{|R_{OBMC}(\boldsymbol{x})|}{|R_{OBMC}(\boldsymbol{x})| + |R_{OBDC}(\boldsymbol{x})|}. \quad (1)$$

The final SI, is calculated as:

$$Y(\boldsymbol{x}) = w(\boldsymbol{x})Y_{OBDC}(\boldsymbol{x}) + \big(1 - w(\boldsymbol{x})\big)Y_{OBMC}(\boldsymbol{x}). \quad (2)$$

The residual for the final SI is calculated using the same weighted average of the residuals.

Ideal Fusion (IF) is also considered [8], [9], which is sometimes referred to as Oracle Fusion. This is a quite common bound in M-DVC literature, it is often used as an upper bound to the performance a fusion technique can achieve. The fused SI is calculated as:

$$Y(\boldsymbol{x}) = \begin{cases} Y_{OBDC}(\boldsymbol{x}) & if\ |X(\boldsymbol{x}) - Y_{OBMC}(\boldsymbol{x})| > |X(\boldsymbol{x}) - Y_{OBDC}(\boldsymbol{x})|, \\ Y_{OBMC}(\boldsymbol{x}) & otherwise, \end{cases} \quad (3)$$

and the same rule is applied to the residuals, in order to fuse them, obtaining the final residual. The technique requires that the original WZ frame, X, is known at the decoder, and therefore, the technique is not applicable in a practical scenario, but it may be used as a bound for the performance of the system. Even though IF is often used as upper bound (e.g. [9]), it is not an upper bound in a strict sense, since it performs a distortion-based optimization on the quality of the SI, and an improved PSNR of the SI need not always lead to superior RD performance.

IF BB is also introduced here as Block-Based (BB) Ideal Fusion (IF). Given a block $\mathcal{B}$, of 4×4 pixels, corresponding to a DCT block, the SAD (Sum of Absolute Differences) of the block between the SI and the corresponding block in the original WZ frame is calculated, and used as reliability measure, to calculate the weight:

$$w_{\mathcal{B}} = \frac{\sum_{\boldsymbol{r}\in\mathcal{B}} |X(\boldsymbol{r}) - Y_{OBMC}(\boldsymbol{r})|}{\sum_{\boldsymbol{r}\in\mathcal{B}} |X(\boldsymbol{r}) - Y_{OBMC}(\boldsymbol{r})| + \sum_{\boldsymbol{r}\in\mathcal{B}} |X(\boldsymbol{r}) - Y_{OBDC}(\boldsymbol{r})|}. \quad (4)$$

The weight $w_{\mathcal{B}}$ is then used to fuse each pixel $\boldsymbol{r}$ belonging to $\mathcal{B}$ as in (2) as well as it is used to generate the residual of the fused SI. Since IF BB requires the knowledge of the original WZ frame, X, this technique cannot be employed in a realistic scenario (as for IF), but it is a useful bound for what concerns the performance which can be reached using the learning approach presented in the next section.

3 PROPOSED M-DVC CODEC ARCHITECTURE

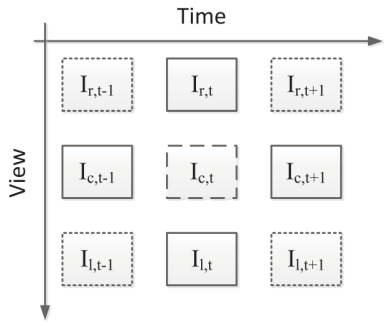


Fig. 1. Stream structure, frames in solid are KFs which the decoder has access to.

At the decoder, the multiview DVC solution has access to frames from other views, as shown in Fig. 1: the left and right views are intra-coded, OBMC only needs to access the decoded frames $I_{c,t-1}$ and $I_{c,t+1}$, OBDC requires also the decoded frames $I_{r,t}$ and $I_{l,t}$, X is the WZ frame, unknown at the decoder. The *central view* is WZ encoded, the *lateral views* (left and right views) are H.264/AVC Intra coded. The architecture of the proposed DVC codec is depicted in Fig. 2 for the encoder and Fig. 3 for the central view decoder (in Fig. 3 the proposed tools are shaded).

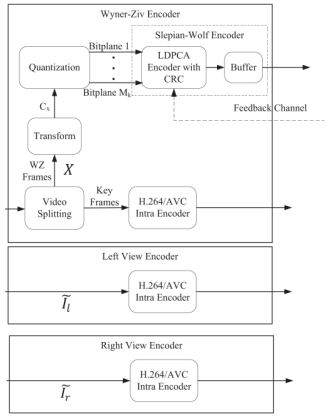


Fig. 2. Independent encoders of the three views, no inter-camera communication is needed.

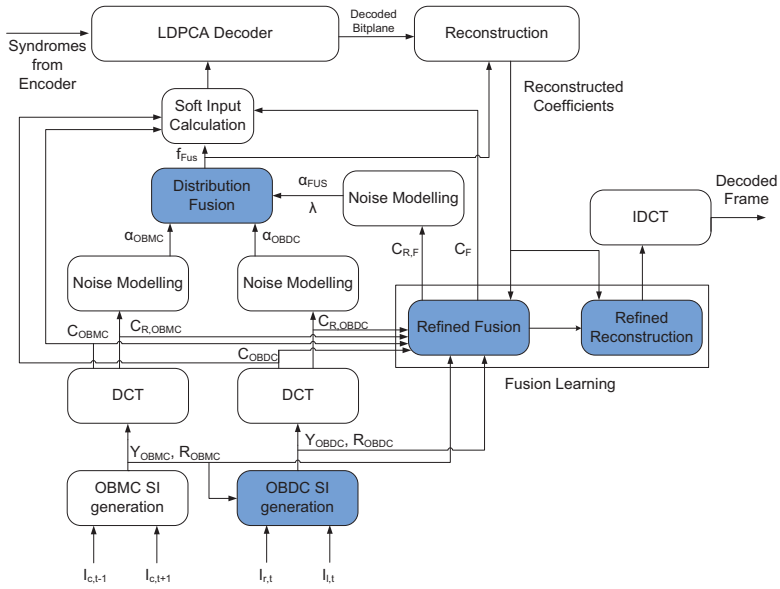


Fig. 3. Architecture of the proposed central view decoder.

The flow and modules (in italics) for the multiview DVC codec can be described as follows:

Central View Encoder (Fig. 2):

- First, the *Video Splitting* module classifies the video frames into WZ frames and key frames according to the Group-of-Pictures (GOP) structure. In a GOP, the first frame is a KF, the others are WZ frames. The frames selected as KFs are encoded by the *H.264/AVC Intra encoder* and sent to the decoder.
- For the WZ frames X, a *Transform* is applied, in this case an integer, 4×4 DCT. The DCT coefficients are uniformly quantized (according to the selected RD point) and divided into bitplanes by the *Quantization* module;
- Each bitplane is fed as input to an *LDPCA Encoder* [21], which generates syndromes which are stored in a *buffer* and sent upon request by the decoder.

Lateral View Encoders (Fig. 2)

- In general, the only multiview codec requirement is that the lateral views (Fig. 1) are encoded independently, i.e. without exploiting any past decoded frames of the same view or from the central view. In this setup, the lateral view frames ($\tilde{I}_l$, $\tilde{I}_r$) are coded with a *H.264/AVC Intra Encoder* but other solutions could be used, e.g. monoview DVC codec.

Lateral View Decoders:

- In this case, the lateral view frames are H.264/AVC Intra decoded but, as previously stated, other solutions could be used, e.g. monoview DVC codec. The left and right reconstructed frames are denoted as I_l and I_r, respectively.

Central View Decoder (Fig. 3):

- The KFs are decoded first, using an H.264/AVC decoder, obtaining $I_{c,t-1}$ and $I_{c,t+1}$. Typically, the keyframe quality matches the quality of the reconstructed WZ frame;
- Then, $I_{c,t-1}$ and $I_{c,t+1}$ are used by the *OBMC SI generation* module to calculate the SI Y_{OBMC} and the (on-line) residual R_{OBMC}. Thereafter, Y_{OBMC} and R_{OBMC} are DCT transformed, and two sets of DCT coefficients C_{OBMC} and $C_{R,OBMC}$ are obtained. In this work on-line residual estimation as detailed in [6] is employed to estimate the difference between the original WZ frame and the SI (the technique employed [6] does not require to have access to the original WZ frame);
- $C_{R,OBMC}$ is used, by the *Noise Modelling* module, to calculate the parameter α_{OBMC} of the laplacian distribution of the correlation noise model [6];
- The *OBDC SI generation* module calculates Y_{OBDC} and the corresponding residual R_{OBDC}. In OBDC, pre-aligned frames $I_{l,t}^{(a)}$ and $I_{r,t}^{(a)}$, are generated from the left-view $I_{l,t}$ and right-view $I_{r,t}$ respectively, removing lateral regions where no correspondence exists between frames. These regions cannot be interpolated using disparity compensation and thus, the co-located pixels in Y_{OBMC} are used. SI and residual are both DCT transformed, generating C_{OBDC} and $C_{R,OBDC}$, respectively.

- $C_{R,OBDC}$ is used, by the *Noise Modelling* module, to calculate the parameter α_{OBDC} of the laplacian distribution of the correlation noise model [6];
- The *Refined Fusion* module generates the fused SI coefficients $C_F^{b_k}$ for DCT band b_k. The calculation of the corresponding residual coefficient $C_{R,F}^{b_k}$ is also carried out. Both coefficients are calculated as weighted averages of the corresponding coefficients (SI and residual) of OBMC and OBDC. The weights are calculated using the MADs (Mean Absolute Differences) between the partially decoded WZ frames and the SIs, see Section 4.3 for more details. The residual coefficients are used by the *Noise Modelling* module to calculate the distribution of the correlation noise model.
- The *Distribution Fusion* module calculates the joint distribution $f_{Fus}^{b_k}$ from the three correlation noise models: OBMC, OBDC and the fused SI. Then, the joint distribution is used by the *Soft Input Calculation* module to calculate the conditional probabilities for the LDPCA decoder. The joint distribution allows the systems to effectively fuse the three different SIs, taking into account the previously decoded information;
- The *LDPCA decoder* requests syndromes from the encoder using a feedback channel: initially, a subset of syndromes is received by the decoder, which attempts to decode the source (bitplane). If the LDPCA decoding succeeds and an 8-bit CRC does not detect any error, the bitplane is assumed to be decoded, otherwise new syndromes are requested via the feedback channel, until successful decoding;
- Once all the bitplanes of the band b_k are decoded, the band is reconstructed by the *Reconstruction* module, using $f_{Fus}^{b_k}$, employing the optimal reconstruction technique outlined in [14];
- At last, when all the bands are successfully decoded, OBMC and OBDC are fused again. The new fused SI is used in a last reconstruction step, in the *Refined Reconstruction* module, to further improve the quality of the decoded WZ frame.

4 MULTIVIEW DECODING TOOLS

In this section the novel tools introduced are analyzed. They are: inter-view *OBDC SI generation*, *Distribution Fusion* and the *Fusion Learning,* which can be divided into two distinct elements: the *Refined Fusion* used during the decoding process, and the *Refined Reconstruction* used at the end of the decoding process (Fig. 3).

4.1 Inter-view Side-Information Generation

When using DCVP for inter-view SI generation, the same algorithm applied for motion interpolation is applied between lateral views. This generates errors, for example, entering and exiting objects from the scene can create areas of wrong matches, because one element in one view has no matches in the other view. This generates wrong disparity vectors, which, in turn, generate erroneous predictions. Typically, when content is acquired in a multiview system there are regions which are present in one view but are occluded in another view, since objects of the scene could be partially or totally occluded from the field-of-view of one camera when compared to another camera. These areas are referred to as lateral areas. On the other hand, there are regions where there are clear correspondences between two views. In addition, when disparity between views is high, a higher search range is needed to have correct correspondences between views. This could lead to wrong matches in lowly textured areas. A way to mitigate these two aforementioned problems is to remove lateral areas from the two frames by aligning them. Naturally, disparity estimation and compensation still needs to be performed, as each object has its own disparity due to distance of the object to the cameras of the multiview system.

4.1.1 Overlapped Block Disparity Compensation

As stated in the previous section, OBDC is conceptually similar to the idea of DCVP, but in order to allow for bigger disparities $I_{r,t}$ and $I_{l,t}$ shall be pre-aligned. Consider that each frame of the multiview system has $n \times m$ spatial resolution. The average disparity d_{avg} between two views is calculated by:

$$d_{avg} = \underset{q\in[-r,r]}{\operatorname{argmin}} \sum_{i=0+q\chi(q)}^{m+q(\chi(-q))-1} \sum_{j=0}^{n-1} \frac{\left|I_{l,t}(i,j) - I_{r,t}(i-q,j)\right|}{(m-|q|)n}, \qquad (5)$$

where $\chi(q)$ is an indicator function, with $\chi(q) = 1$ if $q \geq 0$, and $\chi(q) = 0$ otherwise. r is the positive bound of the search range. If $d_{avg} > 0$ the pixels belonging to the area having i coordinates in the interval $[0, |d_{avg}| - 1]$ are removed from $I_{l,t}(i,j)$ frame, generating $I_{l,t}^{(a)}$, and for $I_{r,t}$ the pixels in the area $[m - 1 - |d_{avg}|, m - 1]$ are removed. In case $d_{avg} < 0$ the roles of the two frames are inverted as can be seen from the interval covered by the i variable in the first sum for a negative q.

The pixels contained in the lateral areas cannot be used for the disparity estimation and interpolation, since they have no match in the other area, therefore these two areas are removed, generating the aligned frames $I_{l,t}^{(a)}$ and $I_{r,t}^{(a)}$, to which OBMC is applied, generating $Y_{OBDC}^{(a)}$. In $Y_{OBDC}^{(a)}$ there are now two areas, $|d_{avg}|/2$ pixels wide, which cannot be interpolated since their corresponding pixels are visible only in one KF. The assumption on the structure of the areas in $Y_{OBDC}^{(a)}$ come from the symmetrical structure of the placement of the cameras. Therefore the unmatched pixels are substituted with the co-located pixels in Y_{OBMC}. A schematic of the algorithm is depicted in Fig. 4. The same substitution is applied to the residual of OBDC, since it suffers from the same problem.

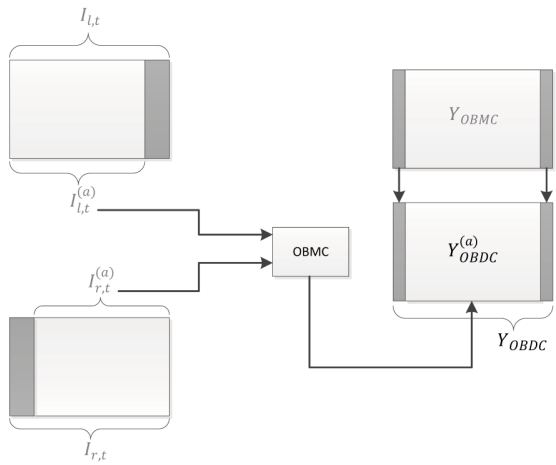


Fig. 4. Illustration of the OBDC SI generation module.

Using the pre-alignment phase, the length of the disparity vectors is reduced. This allows to use a smaller search range, more reliable estimation (less wrong matches) and also lower computational

complexity. In addition, the calculation of the disparity field in the unmatched areas is not performed, allowing a more robust motion estimation for the other blocks: in OBMC (which is the core of OBDC, Fig. 4), and in many similar motion estimation algorithms, smoothing is applied on the motion field after its initial calculation. Erroneous disparity vectors may influence correct ones, therefore with the alignment the propagation of the error is avoided.

4.2 Fusion based on Weighted Distribution

The techniques previously proposed in literature make use of the residual or similar features to estimate the reliability of a given pixel (or block) for the two SI estimations. Once the SI reliability is estimated locally, it is possible to fuse each SI estimates, combining the SI estimates to achieve a higher reliability. Traditionally, many fusion methods for DVC use a binary mask which indicates how the two SI estimations should be fused to maximize the final SI frame quality. However, using this approach a hard decision is made which could be far from optimal and the generation of a new correlation noise model for the fused SI frame would be necessary. Here, a different approach is investigated; by fusing the correlation noise model distributions obtained for the two SI estimations independently, thus avoiding the need to calculate a residual for the fused SI. The better the residual and correlation noise model estimation is, the better the fusion process works. In addition, fusing the distributions according to the correlation model can be improved as better correlation noise models have been proposed in the literature. First, the correlation noise modelling presented in [6] is summarized here for completeness. Defining $C_R^{b_k}$ as the DCT transform of the estimated residual for band b_k, $D(u,v)$ measures the distance between individual coefficients and the average value of coefficients within band b_k:

$$D(u,v) = \left|C_R^{b_k}(u,v)\right| - E\left[\left|C_R^{b_k}\right|\right]. \tag{6}$$

The parameter $\alpha^{b_k}(u,v)$ is calculated by [6]:

$$\alpha^{b_k}(u,v) = \frac{\alpha_c^{b_k}\beta E\left[\left|C_R^{b_k}\right|\right]}{\beta E\left[\left|C_R^{b_k}\right|\right] + (1-\beta)D(u,v)}, \tag{7}$$

The possible values of β are described in [6]. $\alpha_c^{b_k}$ is calculated based on the cluster c (inliers or outliers), the position (u, v) belongs to:

$$\alpha_c^{b_k} = \frac{N_c}{\sum_{(u,v)\in c}\left|\left|C_R^{b_k}(u,v)\right| - E\left[\left|C_R^{b_k}\right||(u,v)\in c\right]\right|}, \tag{8}$$

where N_c is the number of positions belonging to cluster c.

To determine which cluster the coefficient $C_R^{b_k}(u, v)$ belongs to, an estimation function is used, based on the classification (inliers of outliers) on the already decoded coefficients [6]. The algorithm employed is more complex [6], but here the main elements necessary to understand the rest of the work are provided.

Using the procedure previously outlined for the generic laplacian parameter $\alpha^{b_k}(u, v)$, two sets of laplacian parameters can be defined: those for the OBMC SI and those for the OBDC SI, $\alpha_{OBMC}^{b_k}(u, v)$ and $\alpha_{OBDC}^{b_k}(u, v)$, respectively. The weight for fusing the distribution is calculated as proposed in [16]:

$$w^{b_k}(u,v) = \frac{\left(\alpha_{OBMC}^{b_k}(u,v)\right)^2}{\left(\alpha_{OBMC}^{b_k}(u,v)\right)^2 + \left(\alpha_{OBDC}^{b_k}(u,v)\right)^2}. \tag{9}$$

Once the weights are calculated the joint distribution for each position is defined as:

$$f^{b_k,(u,v)} = w^{b_k}(u,v) f_{X|Y_{OBMC}}^{b_k,\ (u,v)} + (1 - w^{b_k}(u,v)) f_{X|Y_{OBDC}}^{b_k,\ (u,v)}, \tag{10}$$

where $f_{X|Y}^{b_k,(u,v)}$ is the estimated distribution for the coefficient (u, v) in band b_k given Y. The idea is the same employed in both pixel and block-based approaches: the weights give an indication of the reliability of the SIs and therefore they are used to fuse the distributions.

This system is compatible with and exploits the efficient block-based correlation noise estimations available in literature.

4.3 Fusion Learning

The SI fusion process described in the previous Section can be improved by using a learning based approach, to leverage the knowledge of the already decoded bands. The idea is to use the already

decoded bands to perform a more reliable SI fusion. This new fusion is then used as part of the distribution fusion. Assuming that band b_k, with $k > 0$, is being decoded (b_0 indicates the DC coefficient) and that the decoding follows a zig-zag scan order, the previously decoded bands b_l, $l < k$ can be used to guide the fusion for each SI DCT coefficient.

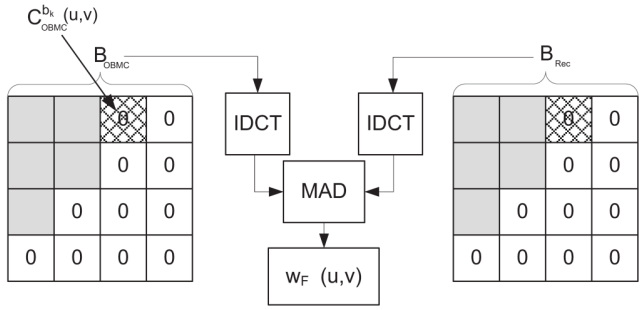


Fig. 5. Calculation of the weights used for the refined fusion for OBMC.

Consider a 4 × 4 DCT block in Y_{OBMC}, denoted as B_{OBMC} and its corresponding block in the partially reconstructed frame B_{Rec}. $C_{OBMC}^{b_k}(u,v)$ denotes the coefficient in band b_k having position (u,v).

First, the non-decoded coefficients are forced to be zero in B_{OBMC} and in the partially reconstructed block B_{Rec}. Then, both DCT blocks are inverse DCT transformed and the MAD between the two blocks is calculated, and it is denoted as the weight $w_F^{OBMC}(u,v)$ as shown in Fig. 5. The MAD is an indicator of how close the previous SI DCT coefficients were to the ones belonging to the original WZ frame. It has to be noted that the WZ frame is not used in this process. The same procedure can be repeated for OBDC, using B_{OBDC} and B_{Rec}, generating the weight $w_F^{OBDC}(u,v)$. The higher the weight the lower the reliability of the corresponding SI, therefore $w_F^{OBMC}(u,v)$ is used as weighting factor for OBDC, while $w_F^{OBDC}(u,v)$ is used as weighting factor for OBMC.

The set of weights is used to generate the fused SI coefficient:

$$C_F^{b_k}(u,v) = \frac{w_F^{OBMC}(u,v)C_{OBDC}^{b_k}(u,v) + w_F^{OBDC}(u,v)C_{OBMC}^{b_k}(u,v)}{w_F^{OBDC}(u,v) + w_F^{OBMC}(u,v)}, \qquad (11)$$

and the corresponding residual estimation for the fused coefficient of the SI:

$$C_{R,F}^{b_k}(u,v) = \frac{w_F^{QBMC}(u,v)C_{R,QBDC}^{b_k}(u,v) + w_F^{QBDC}(u,v)C_{R,QBMC}^{b_k}(u,v)}{w_F^{QBDC}(u,v) + w_F^{QBMC}(u,v)}. \quad (12)$$

To use the correlation noise model of [6], the coefficients $C_F^{b_k}(u,v)$ need to be divided into the inliers cluster and outliers cluster. Therefore (11) is used to calculate $C_F^{b_l}(u,v)$, $0 \le l < k$. The coefficients $C_F^{b_l}(u,v)$ and the estimation function defined in [6] are used to segment the coefficients $C_F^{b_k}(u,v)$ in the two clusters. The three SIs for $k > 0$, are fused using the distribution fusion framework. The final joint distribution is defined as:

$$f_{Fus}^{b_k,(u,v)} = \lambda f^{b_k,(u,v)} + (1-\lambda)\, f_{X|Y_{Fus}}^{b_k,\ (u,v)}, \quad (13)$$

where

$$\lambda = \frac{1}{2^k}, \quad (14)$$

and $f^{b_k,(u,v)}$ is defined in (10).

The adaptive computation of the λ parameter assures that a low weight to the fused SI is selected when the fused SI is not reliable, but it increases rapidly, in line with the expected increase in reliability of the fused SI. The conditional probability of each bit in the SI can be calculated, taking into account the previously decoded bitplanes and the correlation noise model described by $f_{Fus}^{b_k,(u,v)}$. The decoded bitplanes determine the intervals $[L,U)$ each coefficient belongs to. To reconstruct the coefficient in position (u,v), the optimal reconstruction proposed in [14] is used, which is the expectation of the coefficient given the available SIs:

$$C_{Rec}^{b_k}(u,v) = \frac{\int_L^U x f_{Fus}^{b_k,(u,v)}(x)dx}{\int_L^U f_{Fus}^{b_k,(u,v)}(x)dx}, \quad (15)$$

This procedure is carried out for each band b_k, $0 \le k \le N_b$, where N_b is the maximum number of decoded bands, every time updating the weights $w_F^{QBMC}(u,v)$ and $w_F^{QBDC}(u,v)$. Once the band b_{N_b} is decoded, $C_F^{b_k}(u,v)$ is calculated for each $N_b < k \le 16$, and they are used as coefficients in the reconstructed frame. For what concerns the reconstruction of the bands b_k, $0 \le k \le N_b$, they are

reconstructed a second time, to enhance the quality of the reconstructed frame. The segmentation into the inlier cluster and outlier cluster is calculated using the already reconstructed frame and it is not predicted using the mapping function used in the previous steps. As residual the difference between the previously decoded frame and the fused SI is used. In this case $\lambda = 0$ in the reconstruction since at this stage the reliability of the fused SI is so high that it is not necessary to use the inter-view or temporal SIs.

5 EXPERIMENTAL RESULTS

In this section, the proposed coding tools are evaluated. Before presenting the experimental results obtained, the test conditions are first defined. Then, OBDC is compared with DCVP, demonstrating the gains resulting from the pre-alignment phase. For fairness DCVP employs OBMC for disparity estimation and compensation. Furthermore, the fusion algorithm performance is analysed comparing it with single SI decoders and alternative fusion techniques, using cameras at relatively close distance. Finally, the case of unknown disparity is analysed, examining the RD performance of the proposed decoder for 18 different camera configurations.

5.1 Test Conditions

In the experiments, two sequences with still cameras and two sequences with moving cameras at constant inter-camera distance are analysed, in order to test robustness of the system to global motion. The stream structure for the central view has GOP size 2.

The full length of *Outdoor* and *Book Arrival* [22], 100 frames, is coded, and the first 10s of *Kendo* and *Balloons* [22], i.e. 300 frames, are coded. For what concerns the spatial-temporal resolution, all the sequences are downsampled to CIF resolution.

Table 1 - Characteristics of the test sequences

Sequence	Depth Structure	Motion Content	Moving Cameras	Interval of Used Views	Central View	Frame Rate [fps]
Outdoor	Medium	Complex	No	1-15	8	15
Book Arrival	Complex	Medium	No	1-15	8	15
Kendo	Medium/ Complex	Complex	Yes	1-5	3	30
Balloons	Medium/ Complex	Medium	Yes	1-5	3	30

- **Test Sequences:** *Outdoor, Book Arrival, Kendo and Balloons* [22]. These sequences are characterized by different types of motion content, depth structures and camera arrangements, providing a meaningful and varied set of test conditions as outlined in Table 1 ; in the 'Interval of Used Views' column, '1' corresponds to the rightmost view (among the recommended views [23]). In the experiments, the central view is kept fixed while the distance between the central and the lateral cameras is increased, spanning the intervals detailed in Table 1. The distance between two consecutive cameras is 6.5 cm [24] for *Outdoor* and *Book Arrival*, while the distance between two consecutive cameras in *Kendo* and *Balloons* is 5 cm [22].
- **WZ frames coding:** The WZ frames are encoded at four RD points (Q_i, $i = 1,4,7,8,$) corresponding to four different 4×4 DCT quantization matrices [13]. The RD point Q_1 corresponds to the lowest bitrate and quality and the RD point Q_8 to the highest bitrate and quality. The remaining test conditions associated with the DCT, quantization, noise modelling and reconstruction modules are the same as in [6]. For the LDPCA coding a code length of 6336 bits is used, and a CRC check of 8 bits is employed to check the correctness of the decoded result.
- **KFs coding:** The KFs in the central view are H.264/AVC Intra coded (Main Profile) as it is commonly done in, e.g. [6]. The quantization parameter (QP) of the KFs is selected in order to have a similar decoded quality between WZ frames and KF for the same RD point. In Table 2,

the QPs used for each RD point are reported. As previously said, the lateral views are coded with the same parameters as the KFs of the central view.

Table 2 - Quantization parameters for the test sequences

Sequence	Q_1	Q_4	Q_7	Q_8
Outdoor	38	32	28	23
Book Arrival	39	36	29	25
Kendo	39	36	29	22
Balloons	33	30	24	20

- **Quality and Bitrate:** Only the bitrate and PSNR of the luminance component is considered, as it is commonly done in literature. Both WZ frames and KFs are taken into account in rate and PSNR calculations. The rate and PSNR of the lateral views are not taken into account in order to better assess the performance of the given distributed coding solution.

5.2 OBDC-based SI Performance Assessment

In this section, the RD performance of the DVC solution using OBDC, with the sliding window approach, is assessed and compared with the one achieved when DCVP is used to generate the (inter-view) SI; the only difference between OBDC and DCVP is the pre-alignment phase. Table 3 shows the Bjøntegaard bitrate savings (BD-Rate) and Bjøntegaard PSNR gains (BD-PSNR) [25] between OBDC and DCVP when using as lateral views the ones closest to the central view (lowest disparity case) , i.e. views 7 and 9 for *Outdoor* and *Book Arrival* and views 2 and 4 for *Kendo* and *Balloons*. Both SIs are evaluated using the same single SI decoder [6]. For DCVP, the parameters (e.g. search range, strength of the motion smoothing) are adapted, in order to obtain the best average result in terms of RD performance and then the same parameters are used for OBDC. Such parameters are used in OBDC for all the sequences and for all the configurations (distance of the lateral cameras). As it can be observed from Table 3, OBDC allows improvements of the DVC codec RD performance when compared to DCVP, with PSNR gains up to 1.17 dB for the *Book Arrival* sequence, which is characterized by a complex depth structure. No appreciable gains are reported for *Outdoor*, the sequence displaying the simplest depth structure. Table 4 shows the BD-Rate savings and BD-PSNR gains between OBDC and DCVP when using as lateral views the ones furthest away from the central view (according to the view

interval indicated in Table 1), i.e. views 1 and 15 for *Outdoor* and *Book Arrival*, and views 1 and 5 for *Kendo* and *Balloons*. In this case, the parameters for OBDC are the same as those used for generating the results in Table 3. On the other hand, the performance of DCVP is maximized through extensive simulations, finding, for each sequence, the parameters giving the best RD performance. It was not possible to find parameters which were able to perform well for all the sequences for DCVP, while, with the pre-alignment phase in OBDC, the disparity between views is normalized, leaving to the disparity estimation module the task to accommodate for minor differences.

Table 3 – Improvements when using the closest lateral views

BD-Rate savings and BD-PSNR gains for OBDC with respect to DCVP when using as lateral view the ones closest to the central view

Sequence	**BD-PSNR [dB]**	**BD-Rate [%]**
Outdoor	0.00	0.00
Book Arrival	1.17	-17.29
Kendo	0.18	-2.80
Balloons	0.27	-3.94

Table 4 - Improvements when using the furthest away lateral views

BD-RATE savings and BD-PSNR gains for OBDC with respected to DCVP when using as lateral views the ones furthest away from the central view

Sequence	**BD-PSNR [dB]**	**BD-Rate [%]**
Outdoor	0.63	-9.33
Book Arrival	1.52	-21.26
Kendo	0.90	-13.04
Balloons	0.90	-12.26

5.3 M-DVC RD Performance Assessment

In this section, the RD performance of the proposed M-DVC coding solution is assessed and compared with alternative state-of-the-art monoview coding solutions. The left, right and central views used in the experiments are reported in Table 5.

Table 5 - Views used for assessing the proposed M-DVC coding solution RD performance

Sequence	No. Right View	No. Central View	No. Left View
Outdoor	6	8	10
Book Arrival	6	8	10
Kendo	2	3	4
Balloons	2	3	4

5.3.1 Coding Benchmarks

The proposed M-DVC coding solution (described in Section 4) is compared with the following DVC-based codecs:

- **OBMC:** Single SI decoder, as presented in [6]. It is a single view DVC solution, since it exploits the temporal correlation only;
- **OBDC:** Single SI decoder, OBDC is used as SI (outlined in Section 4.1). It exploits the inter-view correlation for the majority of the frame, while the temporal correlation is used for the rest;
- **MDCD-Lin:** It is summarized in Section 2 and implemented following [8]. The weights (calculated from the on-line residuals) used to fuse the SIs are used to fuse the residuals of the two SIs, in order to take into account that a wrong fusion has repercussions not only on the quality of the SI, but also on the quality of the residual. The SI and the residual estimation are fed into the single SI decoder of [6]. While newer techniques were proposed [9], they were unable to provide consistent gains over MDCD-Lin. Therefore MDCD-Lin is employed as benchmark;
- **DISCOVER:** this DVC-based codec [13] it is still widely used as benchmark in literature. The system used as basis for the codec [6] has a structure which is similar to DISCOVER, but it uses an enhanced SI generation module (OBMC) and an advanced noise modelling algorithm. DISCOVER is reported only for completeness, but the focus will be the comparison with the other DVC coding solutions: the OBMC and OBDC-based baseline decoders, in order to make clear how the proposed tools improve the RD performance of the system.

For comparison, the performance of the proposed method is compared with bounds given by ideal fusion techniques:

- **IF BB:** Summarized in Section 2. The SI and the residual estimation are fed into the single SI decoder detailed in [6]. The weights are used to fuse SIs and estimated residuals of the SIs;
- **IF:** Summarized in Section 2. The SI and the residual estimation are fed into the single SI decoder detailed in [6]. The weights are used to fuse SIs and estimated residuals of the SIs;

The proposed M-DVC decoder is finally compared with the following predictive coding references:

- **H.264/AVC Intra:** it is the H.264/AVC codec (Main profile) with only the Intra modes enabled. It is also used for coding the KFs and lateral views. It is also a low-complexity encoding architecture;
- **H.264/AVC No Motion:** which exploits the temporal redundancy in an IB prediction structure setting the search range of the motion compensation to zero, therefore the motion estimation part, which is the most computationally expensive encoding task, is not performed: the co-located blocks in the backward and/or forward reference frames are used for prediction.

5.3.2 RD Performance

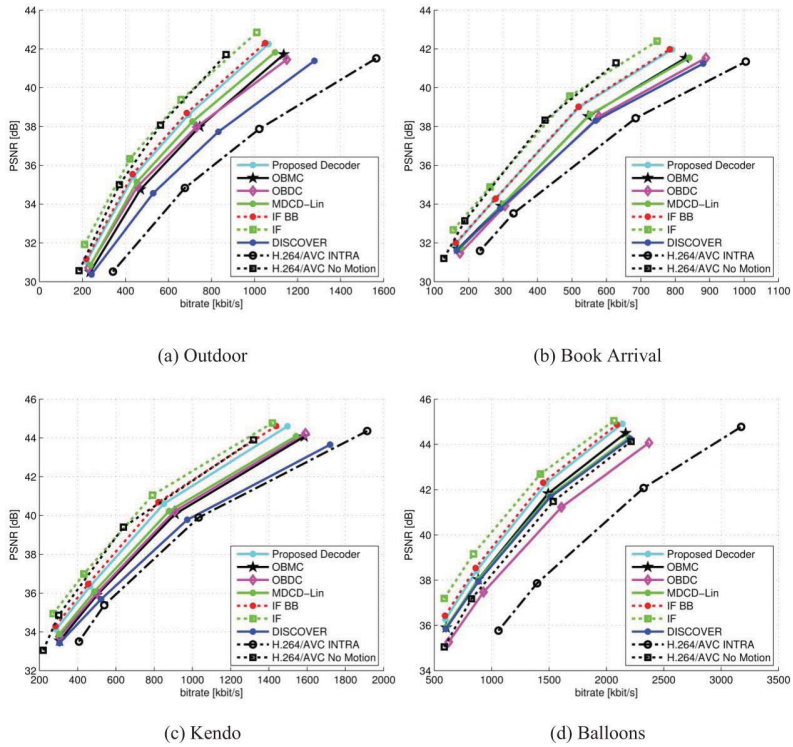
Table 6 – Performance of the proposed solution

BD-Rate savings and BD-PSNR gains for the proposed M-DVC coding solution when compared to the baseline coding solutions (OBDC, OBMC and DISCOVER).

Sequence	OBDC		OBMC		DISCOVER	
	BD-PSNR [dB]	BD-Rate [%]	BD-PSNR [dB]	BD-Rate [%]	BD-PSNR [dB]	BD-Rate [%]
Outdoor	**0.90**	**-12.17**	1.12	-14.55	2.05	-25.36
Book Arrival	1.05	-15.64	**0.72**	**-10.96**	1.01	-15.47
Kendo	**0.79**	**-11.73**	0.94	-13.92	1.53	-22.36
Balloons	1.50	-20.23	**0.50**	**-7.14**	0.68	-9.81
Average	1.06	-14.94	0.82	-11.64	1.32	-18.25

Table 6 reports the BD-Rate savings and BD-PSNR gains for the proposed M-DVC coding solution when compared to the baseline OBMC and OBDC-based DVC coding solutions, using the tools proposed in [6]. For each sequence, the best performing single SI based DVC solution is identified in boldface. The proposed M-DVC video coding solution is able to consistently outperform the best

single SI based DVC solution, with PSNR gains up to 0.9 dB. In the worst case scenario, *Balloons*, the improvement is still significant, allowing a bitrate reduction up to around 7%. The results for the DISCOVER codec are also provided, the average BD-Rate savings are around 18%.



(a) Outdoor (b) Book Arrival

(c) Kendo (d) Balloons

Fig. 6. RD performance for the analysed sequences.

Fig. 6 reports the RD performance results obtained for the *Outdoor*, *Book Arrival*, *Kendo* and *Balloons*, for the nine coding solutions mentioned above. The proposed solution outperforms OBMC, OBDC, DISCOVER and MDCD-Lin, which are all four truly distributed decoders, i.e. they do not require the WZ frame. More specifically, the BD-PSNR gains of the proposed solution are up to 1.5 dB when compared with OBDC and up to 1.12 dB when compared with OBMC. The proposed decoder is able to outperform DISCOVER by up to 2 dB, because DISCOVER uses less advanced SI

generation systems and correlation noise model. MDCD-Lin is able to robustly fuse the SIs for *Outdoor*, *Book Arrival* and *Kendo*, but not for *Balloons.* Furthermore, for the first three sequences, the improvements achieved with MDCD-Lin are lower when compared with the proposed solution, achieving BD-PSNR gains up to 0.33 dB for *Outdoor*. Therefore the proposed solution, leveraging the fusion based on the distributions and the learning process, is able to outperform the other realistic distributed decoders. The use of weights derived from the distributions allows a more precise fusion, because the correlation noise modelling is built on the premise that the residual may be wrong. The learning process allows a refinement of the fused SI, correcting errors during the fusion while decoding the frame. The ideal fusion-based coding solutions: IF and IF BB, require the original WZ frame. Therefore they provide a bound but they are not realistic systems. The BD PSNR gains of IF BB over the proposed coding solution range from 0.02 dB for *Book Arrival* to 0.28 dB for *Kendo.* This shows that the proposed system is able to reach performance close to a block-based fusion technique using block-based tools. IF shows gains by up to 1.14 dB BD PSNR, over the proposed coding solution for the *Outdoor* sequence. For what concerns the reference predictive coders, H.264/AVC Intra is outperformed by every distributed coding solution, regardless the SI generation method. The proposed decoder is able to reach RD performance comparable with H.264/AVC No Motion for *Kendo* and *Balloons.* For *Outdoor* and *Book Arrival*, the only distributed decoder able to compete with H.264/AVC No Motion is IF.

5.4 Camera Distance Impact

This section assesses the impact of varying the distance between the lateral and the central views on the M-DVC codec RD performance. The test conditions are similar to the ones used in the previous sub-section except for the choice of the lateral views. Tables 7 to 10 show the BD-Rate savings and BD-PSNR gains for the proposed M-DVC solution with respect to the baseline OBMC and OBDC-based DVC coding solutions when varying the distance between the cameras for the *Outdoor*, *Book Arrival*, *Kendo* and *Balloons* sequences. The Δ value refers to the difference between the index of the central camera and the index of the right camera. It has to be noted that the same value of Δ may refer to different inter-camera spacing depending on the cameras arrangement. According to the results

obtained, the proposed M-DVC solution is robust to the change of disparity: *Outdoor,* which is characterized by a simpler depth structure, shows a much more stable performance when compared with *Book Arrival*. Only in one case, out of the 18 examined cases, the proposed fusion solution is unable to perform better than the best single SI based DVC solution, but the performance loss is negligible, and the BD between the RD performance of the two single SI decoders (one using OBMC, the other using OBDC) is more than 3 dB, making the problem of increasing the performance by fusion extremely hard.

Table 7 – Performance for variable Δ for the Outdoor sequence

Outdoor: BD-rate savings and BD-PSNR gains for the proposed M-DVC solution with respect to OBDC and OBMC for different Δ values

Δ	**OBDC**		**OBMC**	
	BD-PSNR [dB]	**BD-Rate [%]**	**BD-PSNR [dB]**	**BD-Rate [%]**
1	**0.78**	**-10.53**	1.33	-16.94
2	**0.90**	**-12.17**	1.12	-14.55
3	0.97	-13.10	**0.96**	**-12.69**
4	1.10	-14.87	**0.79**	**-10.58**
5	1.42	-18.76	**0.60**	**-8.22**
6	1.31	-17.66	**0.65**	**-8.82**
7	1.39	-18.56	**0.56**	**-7.71**

Table 8 - Performance for variable Δ for the Book Arrival sequence

Book Arrival: BD-rate savings and BD-PSNR gains for the proposed M-DVC solution with respect to OBDC and OBMC for different Δ values

Δ	**OBDC**		**OBMC**	
	BD-PSNR [dB]	**BD-Rate [%]**	**BD-PSNR [dB]**	**BD-Rate [%]**
1	**0.63**	**-9.45**	1.00	-14.85
2	1.05	-15.64	**0.72**	**-10.96**
3	1.47	-21.48	**0.52**	**-8.02**
4	1.76	-25.39	**0.42**	**-6.55**
5	1.95	-27.85	**0.26**	**-4.02**
6	2.30	-32.31	**0.08**	**-1.25**
7	3.24	-42.82	**-0.06**	**0.99**

Table 9 – Performance for variable Δ for the Kendo sequence

Kendo: BD-rate savings and BD-PSNR gains for the proposed M-DVC solution with respect to OBDC and OBMC for different Δ values

Δ	OBDC		OBMC	
	BD-PSNR [dB]	**BD-Rate [%]**	**BD-PSNR [dB]**	**BD-Rate [%]**
1	**0.79**	**-11.73**	0.94	-13.92
2	1.01	-14.86	**0.62**	**-9.36**

Table 10 – Performance for variable Δ for the Balloons sequence

Balloons: BD-rate savings and BD-PSNR gains for the proposed M-DVC solution with respect to OBDC and OBMC for different Δ values

Δ	OBDC		OBMC	
	BD-PSNR [dB]	**BD-Rate [%]**	**BD-PSNR [dB]**	**BD-Rate [%]**
1	1.50	-20.23	**0.50**	**-7.14**
2	1.90	-24.92	**0.34**	**-4.88**

6 CONCLUSION

In this paper, a novel fusion approach is proposed, based on learning and fusion of the distributions, rather than fusion of the pixels of the SIs. This allows simplifying the problem of estimating the residual of the fused SI, and allows the M-DVC solution to leverage well known techniques for residual estimation and correlation noise model calculation developed for single SI DVC schemes. The proposed M-DVC coding solution proved to be robust to both increments and decrements of the distance between the cameras, which could be a desirable feature in systems where cameras can move with respect to each other or in systems where the distance between cameras is unknown. The proposed learning approach achieved a superior RD performance, on average, when compared with single SI decoders and it showed higher robustness than a residual-based SI fusion technique. The proposed fusion reached performance similar to the performance bounds obtained with a block-based ideal fusion, which relies on the knowledge of the original WZ frame. In case of cameras moving with respect to the scene, but keeping a fixed disparity, the M-DVC solution was able to achieve results

which are close to H.264/AVC No Motion, and in the case of fixed cameras the difference is relatively small, in particular when compared with the gap existing when single SI DVC solutions are used.

COMPETING INTERESTS

All the author(s) declare that they have no competing interests.

REFERENCES

[1] B Girod, A M Aaron, S Rane, and D Rebollo-Monedero: **Distributed Video Coding**. *In Proc. IEEE* 2005, **93**(1):71–83.

[2] R Puri, A Majumdar, and K Ramchandran: **PRISM: A Video Coding Paradigm With Motion Estimation at the Decoder**. *IEEE Trans. Image Process.* 2007, **16**(10): 2436–2448.

[3] C Guillemot, F Pereira, L Torres, T Ebrahimi, R Leonardi, and J Ostermann: **Distributed Monoview and Multiview Video Coding**. *Signal Process. Mag. IEEE*, 2007, **24**(5): 67–76.

[4] D Slepian and J Wolf: **Noiseless Coding of Correlated Information Sources**. *IEEE Trans. Inf. Theory,* 1973 **19**(4):471–480.

[5] A Wyner and J Ziv: **The Rate-Distortion Function for Source Coding with Side Information at the Decoder**. *IEEE Trans. Inf. Theory*, 1976, **22**(1):1–10.

[6] X Huang and S Forchhammer: **Cross-Band Noise Model Refinement for Transform domain Wyner–Ziv Video Coding**. *Signal Process. Image Commun.*, 2012, **27**(1): 16–30.

[7] A Vetro, T Wiegand, and G J Sullivan: **Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard.** *Proc. IEEE*, 2011, **99**(4):626–642.

[8] T Maugey, W Miled, M Cagnazzo, and B Pesquet-Popescu: **Fusion Schemes for Multiview Distributed Video Coding.** *In Proc. European Signal Processing Conference*, 2009:559–563.

[9] F Dufaux: **Support Vector Machine Based Fusion for Multi-View Distributed Video Coding**. *In Proc. Digital Signal Processing (DSP),* 2011:1–7.

[10] M Ouaret, F Dufaux, and T Ebrahimi: **Multiview Distributed Video Coding with Encoder Driven Fusion.** *In Proc.* of *the 2007 European Signal Processing Conference (EUSIPCO-2007)*, 2007.

[11]X Artigas, F Tarrés, and L Torres: **Comparison of Different Side Information Generation Methods for Multiview Distributed Video Coding.** *In Proc. SIGMAP 2007*, 2007.

[12]G Petrazzuoli, M Cagnazzo, and B Pesquet-Popescu: **Novel solutions for side information generation and fusion in multiview DVC.** *EURASIP J. Adv. Signal Process.*, **2013**(1):154.

[13]X Artigas, J Ascenso, M Dalai, S Klomp, D Kubasov, and M Ouaret: **The DISCOVER Codec: Architecture, Techniques And Evaluation.** *In Proc. Picture Coding Symposium (PCS) 2007*, 2007.

[14]D Kubasov, J Nayak, and C Guillemot: **Optimal Reconstruction in Wyner-Ziv Video Coding with Multiple Side Information**. *in Proc. IEEE MMSP 2007,* 2007:183–186.

[15]X Huang, C Brites, J Ascenso, F Pereira, and S Forchhammer: **Distributed Video Coding with Multiple Side Information**. *In Proc. Picture Coding Symposium (PCS) 2009*, 2009:385–388.

[16]Y Li, H Liu, X Liu, S Ma, D Zhao, and W Gao: **Multi-Hypothesis Based Multi-View Distributed Video Coding.** *In Proc. Picture Coding Symposium (PCS) 2009,* 2009:1–4.

[17]M Salmistraro, M Zamarin, and S Forchhammer: **Multi-hypothesis Distributed Stereo Video Coding.** *In Proc. MMPS 2013*, 2013.

[18]H V Luong, L L Raket, X Huang, and S Forchhammer: **Side Information and Noise Learning for Distributed Video Coding Using Optical Flow and Clustering.** *IEEE Trans. Image Process.*, 2012, **21**(12):4782–4796.

[19]C Brites, J Ascenso, and F Pereira: **Learning Based Decoding Approach for Improved Wyner-Ziv video Coding.** *In Proc. PCS 2012*, 2012:165–168.

[20]M Ouaret, F Dufaux, and T Ebrahimi: **Iterative multiview side information for enhanced reconstruction in distributed video coding**," *J Image Video Process*, vol. 2009, pp. 3:1–3:17, Jan. 2009.

[21]D Varodayan, A Aaron, and B Girod: **Rate-Adaptive Codes for Distributed Source Coding**. *EURASIP Signal Process. J.*, 2006, **86**(11):3123–3130.

[22]A Smolic, G Tech, and H Brust: **Report on Generation of Stereo Video Database.** Technical report D2.1, Jul. 2010.

[23] [On-line] "http://www.tanimoto.nuee.nagoya-u.ac.jp/~fukushima/

mpegftv/yuv/Kendo/readme.txt."

[24]I Feldmann, M Mueller, F Zilly, R Tanger, K Mueller, A Smolic, P Kauff, and T Wiegand: **HHI Test Material for 3D Video. ISO**, IEC JTC1/SC29/WG11 MPEG2008, 2008.

[25]G. Bjøntegaard, "Calculation of Average PSNR Differences between RD Curves," in *ITU-T Q6/SG16, Doc. VCEG-M33, in: 13th Meeting*, 2001.

JOINT DISPARITY AND MOTION ESTIMATION USING OPTICAL FLOW FOR MULTIVIEW DISTRIBUTED VIDEO CODING

Matteo Salmistraro, Lars Lau Rakêt†, Catarina Brites‡, João Ascenso‡, Søren Forchhammer**

*DTU Fotonik, Technical University of Denmark, {matsl, sofo}@fotonik.dtu.dk
†DIKU, University of Copenhagen, Denmark, larslau@diku.dk
‡Instituto Superior Técnico, Portugal, {catarina.brites, joao.ascenso}@lx.it.pt

ABSTRACT

Distributed Video Coding (DVC) is a video coding paradigm where the source statistics are exploited at the decoder based on the availability of Side Information (SI). In a monoview video codec, the SI is generated by exploiting the temporal redundancy of the video, through motion estimation and compensation techniques. In a multiview scenario, the correlation between views can also be exploited to further enhance the overall Rate-Distortion (RD) performance. Thus, to generate SI in a multiview distributed coding scenario, a joint disparity and motion estimation technique is proposed, based on optical flow. The proposed SI generation algorithm allows for RD improvements up to 10% (Bjøntegaard) in bit-rate savings, when compared with block-based SI generation algorithms leveraging temporal and inter-view redundancies.

Index Terms— Distributed Video Coding, Multiview Video, Disparity Estimation, Motion Estimation, Optical Flow.

1. INTRODUCTION

In recent years, the Distributed Video Coding (DVC) [1, 2] paradigm has been considered as a promising approach for multiview scenarios [3]. DVC empowers an emerging set of applications, such as visual sensor networks, where each sensing node has limited computational resources, thus requiring low-complexity encoding but also efficient video compression. DVC is based on two information theoretic results from the 1970s, the Slepian-Wolf [4] and Wyner-Ziv (WZ) [5] theorems. In particular, the WZ theorem considers the setup where a source is independently lossy encoded but jointly decoded with a correlated signal, commonly referred to as Side Information (SI). Compared to predictive video coding, DVC exploits the source redundancy partially or totally at the decoder. This enables to leverage inter-camera redundancy without inter-camera communication. In Multiview DVC (M-DVC), the SI creation and fusion techniques play a critical role in the overall compression performance. Inter-view SI is generated by exploiting the inter-view correlation between cameras, and the intra-view SI is generated exploiting the temporal correlation. Once the two estimations are generated, fusion techniques are typically applied to obtain the final SI, i.e. the two estimations are combined according to their reliability [6, 7]. The better the quality of the fused SI frame, the smaller the number of 'errors' the DVC decoder has to correct and, thus, less redundancy bits are transmitted. However, an alternative SI creation approach has been proposed in [8], the MultiView Motion Estimation (MVME) technique. First, MVME estimates the disparity between temporally aligned frames in the central and lateral (left or right) views and then, motion is estimated for each matched block in the lateral view. The motion vectors obtained for the lateral view are then applied to the central WZ frame to generate the SI. To estimate the motion and disparity of each block, MVME uses a block matching algorithm. However, Optical Flow (OF) for motion estimation can lead to higher SI quality when compared with classical block-based SI generation methods [9], such as Overlapped Block Motion Compensation (OBMC) which is an efficient intra-view SI generation algorithm [10], relying on the use of weighted average of multiple candidate blocks. Motion estimation based on OF produces a dense motion field, where the displacement of each pixel is influenced by the displacement of all other pixels through total variation regularization, allowing for higher flexibility in the motion estimation compared with e.g. OBMC [10]. Thus, the optical flow framework is exploited into a novel SI creation solution, called Time-Disparity OF (TDOF) with the following contributions: 1) the use of OF for estimating the motion of the current view given the lateral views in a DVC setup; and 2) the handling of occlusion through filtering and joint interpolation of scattered sets. To allow for better inter-view matching quality, a pre-alignment step is introduced, to handle areas lying outside the field of view of one camera but available in the other view. TDOF shares with MVME the general concept of using the motion of lateral views to generate SI. Finally, the robustness of the proposed TDOF method is analysed using an on-line correlation noise modelling, as opposed to many M-DVC works still relying on off-line modelling [6, 7].

M. Salmistraro, L. L. Rakêt, Catarina Brites, J. Ascenso, S. Forchhammer, "Joint Disparity and Motion Estimation Using Optical Flow for Multiview Distributed Video Coding", *2014 European Signal Processing Conference (EUSIPCO 2014)* (submitted).

The rest of the paper is organized as follows: in Section 2 the adopted DVC architecture is presented, in Section 3 the proposed SI generation method is described and Section 4 assesses its performance.

2. MULTIVIEW DVC CODING ARCHITECTURE

The proposed M-DVC codec is based on the monoview DVC codec presented in [10] and is depicted in Fig. 1. The three-camera multiview setup depicted in Fig. 2 is considered here where central view frames can be WZ or Intra coded according to a fixed GOP structure. The left and right views, $\widetilde{I_r}$ and $\widetilde{I_l}$, are independently encoded, and although they can be coded with any available video coding solution, the H.264/AVC Intra coding scheme has been adopted here, as it is typically done in literature [8]. The encoder of the central view divides the frames into Key Frames (KFs) and WZ frames, X. The KFs are coded independently, using H.264/AVC Intra coding and the WZ frames are DCT transformed, quantized and organized in bitplanes. Each bitplane is fed into a Low-Density Parity Check Accumulate (LDPCA) encoder [11] that generates the syndromes, which are stored in a buffer and sent to the decoder upon request. The M-DVC decoder uses already decoded frames from central and lateral views to generate the SI frame Y and a residual frame R, which corresponds to the estimation of $X - Y$. The soft probabilities of each bitplane are then calculated with a Laplacian correlation noise model, derived from R: all the residuals used in this work are estimated without using X. A feedback channel allows the decoder to request new syndromes (as in the Stanford DVC codec [1]) if the received syndromes are not enough to successfully decode the source (bitplane). To improve the reliability of the decoded bitplane, an additional 8-bit CRC is used to check for any remaining decoding errors. Once all the bitplanes of a given DCT band are decoded the corresponding coefficients are reconstructed [12].

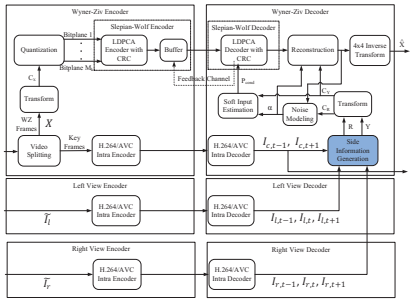


Fig. 1. Proposed M-DVC coding architecture.

3. TDOF SIDE INFORMATION GENERATION

Consider X as the WZ frame to be coded. The TDOF (Time-Disparity OF) approach makes use of three frames in the right view, three frames in the left view and $I_{c,t-1}$ and $I_{c,t+1}$ in the central view. The disparity field between $I_{c,t-1}$ and $I_{l,t-1}$ is first calculated. Then, for each point (which may be a non-integer position) hit by a disparity vector in $I_{l,t-1}$, the motion vector between this point and its corresponding point in $I_{l,t}$ is calculated, as depicted in Fig. 2. The motion vector is then applied to the corresponding pixel, $\boldsymbol{x}$, in $I_{c,t-1}$, obtaining a scattered set of points $S_{l,t-1}$ for the SI frame Y. The set of frames $I_{c,t-1}$, $I_{l,t-1}$ and $I_{l,t}$ constitute a "path". Applying this procedure to the other three paths, three new sets of scattered points can be obtained: $S_{l,t+1}$ (path $I_{c,t+1}$, $I_{l,t+1}$ and $I_{l,t}$), $S_{r,t-1}$ (path $I_{c,t-1}$, $I_{r,t-1}$ and $I_{r,t}$) and $S_{r,t+1}$ (path $I_{c,t+1}$, $I_{r,t+1}$ and $I_{r,t}$). The described solution differs from [8] because it is proposed here to calculate the disparity and motion with an OF technique followed by filtering and joint interpolation, i.e. the fusion of the scattered sets. The TDOF SI generation algorithm can be divided into three steps, corresponding to the three introduced novelties: 1) pre-alignment of the time aligned frames to remove unmatched areas, 2) OF calculation, 3) scattered sets filtering and joint interpolation, to obtain the final SI.

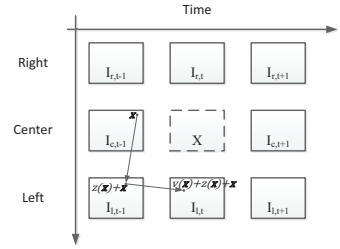


Fig. 2. Three-camera setup, depicting the path for the set $S_{l,t-1}$.

3.1 Pre-alignment

To allow for higher matching quality and higher robustness during the disparity estimation, the two temporally co-located frames, (e.g. $I_{l,t-1}$ and $I_{c,t-1}$ for the set $S_{l,t-1}$), of each path are pre-aligned. The pre-alignment phase removes unmatched lateral (two) bands, one in the lateral frame and one in the central frame to assure that no significant occlusions can occur. This allows for a higher quality match, because wrong estimations in those bands would influence the quality of the whole SI frame, given the particular formulation of the OF problem. Consider the path leading to the calculation of $S_{l,t-1}$ and the frames $I_{l,t-1}$ and $I_{c,t-1}$ with dimension $m \times n$ pixels (with m being the number of col-

umns). The average global disparity d_G between these two frames is calculated by minimizing:

$$d_G = \underset{q}{\operatorname{argmin}} \sum_{i=0+q\chi(q)}^{m-1+q\chi(-q))} \sum_{j=0}^{n-1} \frac{|I_{l,t-1}(i,j) - I_{c,t-1}(i-q,j)|}{(m-|q|)n}, \quad (1)$$

where the indicator function χ is defined as: $\chi(q) = 1$ if $q \geq 0$, and $\chi(q) = 0$ otherwise, and q is the search range employed in (1). In this way, the left (right) lateral band of a frame which has no correspondence in the right (left) band of the other frame is removed, generating the corresponding aligned set of frames $I^a_{l,t-1}$ and $I^a_{c,t-1}$. The relative distance between cameras does not change and it is the same between center and left and between center and right, therefore the same disparity may be used for pre-aligning the other left path, leading to generate set $S_{l,t+1}$, while the opposite value can be used for the two remaining sets.

3.2 Optical Flow Calculation

With the aligned frames $I^a_{l,t-1}$ and $I^a_{c,t-1}$, the disparity field z can be estimated by minimizing the data fidelity term:

$$C_D(\boldsymbol{x}, z) = |I^a_{l,t-1}(\boldsymbol{x} + z(\boldsymbol{x})) - I^a_{c,t-1}(\boldsymbol{x})|. \quad (2)$$

$z(\boldsymbol{x})$ is, in general, a vector-valued function, therefore there are two unknowns at every point in (2). To solve this issue, the TV-L^1 formulation [13] has been adopted. TV-L^1 relies on the L^1-norm of the OF constraint (2), and a Total Variation (TV) regularization term: the 1-Jacobian of the field ($J_1 z(\boldsymbol{x})$) [14] is adopted here. For the TV-L^1 problem, a computationally efficient solution exists to minimize:

$$E_D(z) = \int \lambda_D C_D(\boldsymbol{x}, z) + |J_1 z(\boldsymbol{x})| d\boldsymbol{x}, \quad (3)$$

where $\boldsymbol{x}$ is a 2D point in $I_{c,t-1}$. OF based disparity estimation produces a dense field, since a disparity vector is calculated for each pixel. The calculation of the disparity vector for point $\boldsymbol{x}$ is influenced by the quality of all the other matches (i.e. magnitude of the constraint) and the smoothness of the disparity field. It is here proposed to directly generate a motion field having source in $I^a_{c,t-1}$: the constraint for the motion estimation is defined taking into account the motion field v and the disparity z.

$$C_T(\boldsymbol{x}, z, v) = |I^a_{l,t-1}(\boldsymbol{x} + z(\boldsymbol{x})) - I^a_{l,t}(\boldsymbol{x} + z(\boldsymbol{x}) + v(\boldsymbol{x}))|, \quad (4)$$

The minimization of the energy, leading to the estimation of v is performed jointly with the already calculated z:

$$E_T(v, z) = \int \lambda_T C_T(\boldsymbol{x}, z, v) + |J_1 v(\boldsymbol{x})| d\boldsymbol{x}. \quad (5)$$

With this approach, each point $\boldsymbol{x}$ in $I^a_{c,t-1}$ is directly coupled with its corresponding motion vector $v(\boldsymbol{x})$, which can be used to project $\boldsymbol{x}$ into a new location, obtaining the scattered set $S_{l,t-1}$, composed by the elements $p_{\boldsymbol{x}}$.

$$p_{\boldsymbol{x}} = [I^a_{c,t-1}(\boldsymbol{x}), \boldsymbol{x} + v(\boldsymbol{x}), C_T(\boldsymbol{x}, z, v)]. \quad (6)$$

3.3 Scattered Sets Filtering and Joint Interpolation

Once the four sets are available, it is possible to perform interpolation and obtain four SI frame estimations: $Y_{l,t-1}$, $Y_{l,t+1}$, $Y_{r,t-1}$ and $Y_{r,t+1}$ which averaged could lead to the final SI Y, mimicking the procedure used for MVME [8]. However, the use of the OF based technique presented in the previous Section allows higher granularity. Thus, the following solution is proposed; first, the scattered sets are fused:

$$S_l = S_{l,t-1} \cup S_{l,t+1} \text{ and } S_r = S_{r,t-1} \cup S_{r,t+1}. \quad (7)$$

This fusion allows the handling of holes in SI, due to occlusions and disocclusions. Moreover, when some points are wrongly matched with other points (that occurs when their true match is occluded), the density of points in some areas increases. Therefore, it is proposed to process each fused set (S_l and S_r) to remove points from too dense areas: for each $p_{\boldsymbol{x}}$ having $C_T(\boldsymbol{x}, z, v) \geq \Psi$, where Ψ is a threshold, the Φ closest neighbors are selected, including $p_{\boldsymbol{x}}$. Among them the neighbor having the highest value of $C_T(\boldsymbol{x}, z, v)$ is removed. Once the two sets have been filtered, they are interpolated to obtain the values for the pixel locations, using linear triangular interpolation. The interpolation is divided in two phases: first, using the scattered points, a piecewise triangular surface is generated. Then, for each point having integer coordinates, a bivariate linear interpolation is applied inside the triangle it belongs to [15]. This leads to the generation of the two joint estimations Y_l and Y_r. The final SI Y and its corresponding residual estimate R are calculated as:

$$Y = \frac{1}{2}(Y_l + Y_r) \text{ and } R = Y_l - Y_r. \quad (8)$$

These calculations are carried out only for points which do not belong to the occluded regions identified in the pre-alignment phase. For the pixels belonging to these regions, Y_l or Y_r is used as the final SI, depending on which one is available. For what concerns the correlation noise (or residual) in those regions, if Y_l is available the residual is calculated as $Y_{l,t-1} - Y_{l,t+1}$, otherwise $Y_{r,t-1} - Y_{r,t+1}$ is used.

4. EXPERIMENTAL RESULTS

In Table 1, a detailed description of the test conditions is presented. To ensure a representative set of scenarios, video sequences [16] with still cameras (*Outdoor* and *Book Arrival*) and moving cameras (*Kendo* and *Balloons*), with different depth structures have been selected. All sequences were downsampled to CIF resolution. For the first two sequences, the distance between two consecutive cameras is

6.5cm, while for the latter two the distance is 5cm. To analyze the robustness of the TDOF method, consecutive cameras for the first two sequences are not used, leading to higher disparity. The central view has been coded using a GOP 2 structure; all experiments are conducted only for the luminance component, as usual in DVC. The RD performance of the proposed solution is assessed using four RD points, obtained using four quantization tables (Q_i) of the DISCOVER project [17] and varying the Quantization Parameter (QP) of the KFs accordingly, as shown in Table 2. The KFs are H.264/AVC Intra coded (Main profile). The QPs are chosen to minimize the PSNR variation in the central view between KFs and WZ frames. The left and the right views are H.264/AVC Intra coded (Main profile), with the same QPs used for KFs.

Sequence	Frame Rate	Coded Frames	Views
Outdoor	15 fps	100	6,8,10
Book Arrival	15 fps	100	6,8,10
Kendo	30 fps	300	3,4,5
Balloons	30 fps	300	3,4,5

Table 1. Test Conditions

For the OF calculation, the energies E_D and E_T are minimized, through an iterative procedure, in a coarse-to-fine pyramid, following the general implementation described in [14, 18]: 70 pyramid levels are used, and linear interpolation is used to upscale the flows from a coarser level to a finer one. Since the OF formulation treats a frame like a continuous function, bicubic interpolation is used. After extensive experiments, it has been determined that $\lambda_T = 115$ and $\lambda_D = 25$ are appropriate for good RD performance, and they are the same for all RD points. The λ_D and λ_T values are different since the disparity field is usually much smoother than the motion field, therefore a high value of λ_D is not required to ensure disparity matching, because usually $|J_1 v(\boldsymbol{x})| > |J_1 z(\boldsymbol{x})|$. The proposed TDOF method is compared with alternative SI generation solutions to justify some of the algorithm steps, namely: 1) Y_{avg}: The SI corresponds to the average of the OF-generated estimations $Y_{l,t-1}$, $Y_{l,t+1}$, $Y_{r,t-1}$ and $Y_{r,t+1}$; and 2) Y_l (resp. Y_r): The SI is generated from the left (resp. right) view through OF, scattered set filtering and joint interpolation (see Section 3.3). Table 3 shows the average SI quality for all frames for these three optical flow based solutions.

Sequence	Q_1	Q_4	Q_7	Q_8
Outdoor	38	32	28	23
Book Arrival	39	36	29	25
Kendo	39	36	29	22
Balloons	33	30	24	20

Table 2. QPs for Right and Left Views and KFs

As shown, the scattered set filtering and joint interpolation technique is able to outperform, in SI quality, the simple average Y_{avg} of the OF-generated estimations and a single view (left or right) joint estimation (Y_l or Y_r).

The RD performance of the proposed M-DVC solution (with the TDOF method) is compared with the RD performance of three M-DVC codecs integrating the following benchmark SI generation solutions: 1) OBMC [10]; 2) DCVP (Disparity Compensated View Prediction) [19] applying OBMC between $I_{r,t}$ and $I_{l,t}$; and 3) MVME [8]. The OBMC and DCVP use the correlation noise (or residual) estimation of [10], MVME and TDOF use the residual estimation in (8).

Sequence	Y_{avg}	Y_l	Y_r	TDOF
Outdoor	34.19	36.02	35.63	36.65
Book Arrival	37.51	37.56	37.51	38.48
Kendo	36.38	37.57	37.79	38.93
Balloons	39.69	40.82	40.82	41.47

Table 3. SI PSNR [dB] for alternative OF-based SI Generation Methods

	MVME		OBMC		DCVP	
Sequence	**PSNR [dB]**	**Rate [%]**	**PSNR [dB]**	**Rate [%]**	**PSNR [dB]**	**Rate [%]**
Outdoor	0.30	-4.13	0.43	-5.90	0.44	-6.14
Book Arrival	0.50	-7.46	0.29	-4.45	2.55	-34.21
Kendo	0.71	-10.40	0.58	-8.63	0.61	-9.03
Balloons	0.59	-8.27	0.21	-2.92	1.48	-19.51

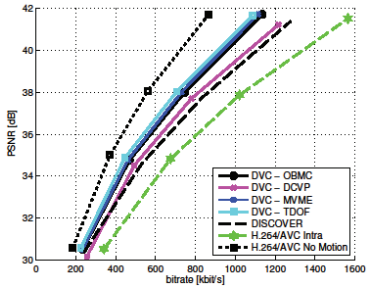
Table 4. BD Gains of the Proposed TDOF Solution Regarding alternative SI Generation Methods

In the filtering of the scattered set, proposed for the TDOF method, Φ=3, and Ψ is chosen such that the number of points having $C_T(\boldsymbol{x}, z, v) \geq \Psi$ is less than or equal to the 1% of the total number of points in the fused scattered sets S_l or S_r. Table 4 shows the Bjøntegaard (BD) [20] PSNR gains and bitrate savings between the TDOF SI generation method and the alternatives MVME, OBMC and DCVP. Rate and PSNR are calculated on all the frames of the central view. The OF-based solution outperforms all the other proposed solutions, for all the sequences; the highest gain when compared with MVME and OBMC is obtained for the *Kendo* sequence (0.71dB and 0.58dB respectively), which has medium complex depth structure and a complex object motion. For what concerns DCVP, the highest gains are obtained for the sequences *Book Arrival* and *Balloons* (2.55dB and 1.49dB respectively) with relatively low motion activity and rather complex depth structure, making the temporal interpolation task much simpler than disparity compensation. In Fig. 3, the RD performance results obtained for *Outdoor* and *Kendo* are presented; again, only the central view KFs and WZ frames rate and PSNR is considered. As shown in Fig. 3, the SI PSNR gains are reflected in

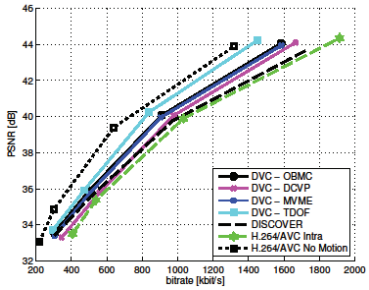
M-DVC codec RD performance improvements. A similar trend is also followed by the other two sequences; RD results are not shown here for these two sequences due to paper length constrains.

5. CONCLUSION

In this paper, a new OF-based method for joint disparity and motion estimation for SI generation in M-DVC, called TDOF, is proposed; the TDOF method includes techniques to filter erroneous interpolations and to jointly interpolate sets of scattered points. The TDOF SI generation method leads to bitrate savings up to 10%, 8.6% and 34% when compared with MVME, OBMC and DCVP, respectively.



(a)



(b)

Fig. 3. RD performance for *Outdoor* (a) and *Kendo* (b) sequences.

REFERENCES

[1] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed Video Coding," *Proc. IEEE*, vol. 93, no. 1, pp. 71–83, Jan. 2005.

[2] R. Puri, A. Majumdar, and K. Ramchandran, "PRISM: A Video Coding Paradigm With Motion Estimation at the Decoder," *IEEE Trans. Image Process.*, vol. 16, no. 10, pp. 2436–2448, Oct. 2007.

[3] C. Guillemot, F. Pereira, L. Torres, T. Ebrahimi, R. Leonardi, and J. Ostermann, "Distributed Monoview and Multiview Video Coding," *IEEE Signal Process. Mag.*, vol. 24, no. 5, pp. 67–76, 2007.

[4] D. Slepian and J. Wolf, "Noiseless Coding of Correlated Information Sources," *IEEE Trans. Inf. Theory*, vol. 19, no. 4, pp. 471–480, Jul. 1973.

[5] A. Wyner and J. Ziv, "The Rate-Distortion Function for Source Coding with Side Information at the Decoder," *IEEE Trans. Inf. Theory*, vol. 22, no. 1, pp. 1–10, Jan. 1976.

[6] T. Maugey, W. Miled, M. Cagnazzo, and B. Pesquet-Popescu, "Fusion Schemes for Multiview Distributed Video Coding," in *Proc. EUSIPCO 2009*, Scotland, 2009, vol. 1, pp. 559–563.

[7] F. Dufaux, "Support Vector Machine Based Fusion for Multi-View Distributed Video Coding," in *Proc. DSP,* 2011, pp. 1–7.

[8] X. Artigas, F. Tarrés, and L. Torres, "Comparison of Different Side Information Generation Methods for Multiview Distributed Video Coding," in *Proc. SIGMAP 2007*, Barcelona, Spain, 2007.

[9] Huynh Van Luong, L. L. Raket, Xin Huang, and S. Forchhammer, "Side Information and Noise Learning for Distributed Video Coding Using Optical Flow and Clustering," *IEEE Trans. Image Process.*, vol. 21, no. 12, pp. 4782–4796, Dec. 2012.

[10] X. Huang and S. Forchhammer, "Cross-Band Noise Model Refinement for Transform domain Wyner–Ziv Video Coding," *Signal Process. Image Commun.*, vol. 27, no. 1, pp. 16–30, Jan. 2012.

[11] D. Varodayan, A. Aaron, and B. Girod, "Rate-Adaptive Codes for Distributed Source Coding," *EURASIP Signal Process. J.*, vol. 86, no. 11, pp. 3123–3130, Nov. 2006.

[12] D. Kubasov, J. Nayak, and C. Guillemot, "Optimal Reconstruction in Wyner-Ziv Video Coding with Multiple Side Information," in *Proc. IEEE MMSP, October, 2007*, 2007, pp. 183–186.

[13] C. Zach, T. Pock, and H. Bischof, "A Duality based Approach for Realtime TV-L^1 Optical Flow," in *Proc. Proceedings of the 29th DAGM conference on Pattern recognition*, Heidelberg, Germany, 2007, pp. 214–223.

[14] L. L. Rakêt, L. Roholm, A. Bruhn, and J. Weickert, "Motion Compensated Frame Interpolation with a Symmetric Optical Flow Constraint," in *Advances in Visual Computing*, vol. 7431 , 2012, pp. 447–457.

[15] I. Amidror, "Scattered Data Interpolation Methods for Electronic Imaging Systems: a Survey," *J. Electron. Imaging*, vol. 11, no. 2, p. 157, Apr. 2002.

[16] A. Smolic, G. Tech, and H. Brust, "Report on Generation of Stereo Video Database," Technical report D2.1, Jul. 2010.

[17] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER Codec: Architecture, Techniques And Evaluation," in *Proc. PCS*, Lisboa, Portugal, 2007.

[18] L. L. Rakêt, "Optical Flow C++ Implementation." [Online]. Available: http://image.diku.dk/larslau/.

[19] M. Ouaret, F. Dufaux, and T. Ebrahimi, "Multiview Distributed Video Coding with Encoder Driven Fusion," in *Proc. EUSIPCO 2007,* Poznan, Poland, 2007.

[20] G. Bjøntegaard, "Calculation of Average PSNR Differences between RD Curves," in *ITU-T Q6/SG16, Doc. VCEG-M33, in: 13th Meeting*, Austin, USA, 2001.

LOW DELAY WYNER-ZIV CODING USING OPTICAL FLOW

Matteo Salmistraro, Søren Forchhammer

DTU Fotonik, Technical University of Denmark, Ørsteds Plads,
2800 Kgs. Lyngby, Denmark. Emails: {matsl, sofo}@fotonik.dtu.dk

ABSTRACT

Distributed Video Coding (DVC) is a video coding paradigm that exploits the source statistics at the decoder based on the availability of the Side Information (SI). The SI can be seen as a noisy version of the source, and the lower the noise the higher the RD performance of the decoder. The SI is usually generated by means of interpolation-based methods, which rely on the availability of a preceding and a following frame with respect to the to-be-decoded one. These methods lead to relatively high RD performance but also high delays. This work is focused on a low-delay codec, relying only on preceding frames for the generation of the SI by means of Optical Flow (OF). OF is also used in the refinement step of the SI for enhanced RD performance. Compared with a state-of-the-art extrapolation-based decoder the proposed solution achieve RD Bjøntegaard gains up to 1.3 dB.

Index Terms— Distributed Video Coding, Optical Flow, Wyner-Ziv Coding, Extrapolation.

1. INTRODUCTION

DVC is a coding paradigm where the exploitation of temporal redundancy in a video is done at the decoder rather than at the encoder. This is appealing for applications like mobile video coding, sensor networks and video surveillance. DVC is based on two information theory results: the Slepian-Wolf [1] and the Wyner-Ziv (WZ) [2] theorems where, in the second case, source data are independently lossy coded but jointly decoded using a correlated source at the decoder, which is commonly referred to as Side Information (SI).

The first practical solutions for the DVC problem have been proposed in [3, 4]. Further improvements on [3] led to the creation of the DISCOVER [5] codec, which is still the state-of-the-art DVC solution. Many coding tools have been proposed so far, mainly addressing the performance gap between DVC and predictive coding solutions, e.g. [6]. Nevertheless the majority of these decoders use an interpolation-based SI generation system. An interpolation-based SI generation system, like OBMC (Overlapped Block Motion Compensation) [6], relies on a coding order different from the order the frames are generated, because the generation of the SI for a WZ frame requires the availability of a preceding and following frame. These frames are referred to as Key Frames (KFs) and are independently encoded and classified with respect to each other and the WZ frames, employing e.g. H.264/AVC Intra Coding. Interpolation-based SI generation systems contributed to reduce the performance gap between predictive coding and distributed coding solutions, in particular a great deal of work has been dedicated to the Group-Of-Picture (GOP) 2 structure: KF-WZ-KF. Nevertheless these requirements make DVC coding solutions not suitable for scenarios where very low-delay encoding is required. Extrapolation, on the other hand, requires only previous frames, usually two, therefore the low-delay requirement can be met using extrapolation-based solutions.

Extrapolation has not been as thoroughly investigated as Interpolation methods, but some works have addressed the problem, first in [4] than notably in [7, 8, 9]. It was proposed to add a refinement step to a block-based extrapolated SI [8]. In [9] an advanced dense motion estimation technique, the Lukas-Kanade algorithm, is used to calculate the motion field, nevertheless, the proposed coding solution requires the encoder to generate and code a mode decision mask, increasing the coding complexity, which may be undesirable. Secondly the Lukas-Kanade algorithm still relies on a block-based matching as basis of the dense motion estimation. Also PRISM [4] is a low-delay encoder, but it requires a classifier, therefore the same comment as for [9] can be made.

The majority of the Interpolation and Extrapolation algorithms presented in literature so far rely on block-based matching and motion compensation, but recently Optical Flow (OF) proved to outperform block-based SI generation systems [10, 11] and it has been suggested that block-based and OF-based SI generation methods could be fused. OF generates a dense motion flow, where a motion vector is estimated for each pixel of the frame. The field estimation for each pixel is influenced by a smoothness constraint calculated on the whole field; thus allowing to take into account the regularity of the field as well as the matching in a joint way, as opposed to block-based systems [6, 8] relying on block-based motion estimation followed by a smoothing step.

The main problems of the current methods for SI generation are the assumption of linear motion and wrong matches due to occlusions, disocclusions or deformations. To address such issues many works have addressed the SI refinement

M. Salmistraro, S. Forchhammer, "Low Delay Wyner-Ziv Coding Using Optical Flow", *2014 IEEE Int'l Conf. on Image Processing (ICIP 2014)*, (submitted).

problem, e.g. [8, 12]. A refined SI is generated by means of a Partially Decoded WZ frame, which is used to refine the motion estimation, leading to a higher quality SI. The majority of the SI refinement methods have been developed for Interpolation-based SI generation systems (with some notable exceptions, e.g. [8]) and they rely on block-based matching: here OF-based SI refinement is explored in the context of an extrapolated SI.

The main contributions of this work are: the use of OF for Extrapolation-based SI generation, namely EX-OF and the use of OF for SI refinement without increasing the delay, denoted as OF-REF. The proposed techniques do not require any modification to the encoder and therefore the encoder complexity if left unchanged.

The paper is organized as follows. In Section 2 the employed codec is presented. Section 3 describes the proposed coding tools and their use in the described architecture. The experimental results are discussed in Section 4 and finally Section 5 concludes this work.

2. DISTRIBUTED VIDEO CODING

The proposed Transform Domain Wyner-Ziv (TDWZ) decoder is built starting from the decoder presented in [6] and depicted in Fig. 1, later enhanced using Multi-Hypothesis (MH) decoding in [10, 13]. The proposed codec is a very low-delay one, relying only on extrapolation-based SI generation, as opposed to the versions proposed in [6, 10, 13] which relied on interpolation-based SI generation. In case

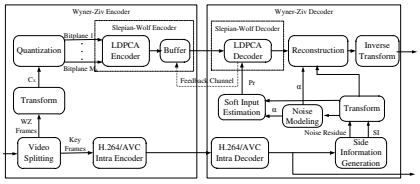


Fig. 1: TDWZ codec used as basis, from [6].

of GOP 2 structure, the odd frames are labelled as KFs and H.264/AVC Intra coded. The even frames are WZ encoded: first a 4×4 DCT transform is applied to the WZ frame, denoted as X, and the DCT coefficients are quantized. The quantization matrices are chosen in order to have similar quality between KFs and WZ frames. Once the coefficients are quantized, the resulting bitstream is re-organized in bitplanes. Each bitplane is then fed into a LDPCA (Low Density Parity-Check Accumulate) encoder [14] and the calculated parity bits are stored in a buffer. The parity bits are then sent in subsets to the decoder, upon request received by mean of a feedback channel.

The decoder has only access to previously decoded frames: supposing that the WZ frame has time index t, the decoder can exploit frames at instants $t-1$ and $t-2$ which can be, in general, WZ frames or KFs. In case of GOP 2 they are a KF and a WZ frame, respectively. First the motion between the frames at $t-1$ and $t-2$ is estimated, then it is used to predict the motion between $t-1$ and t. Applying these newly estimated Motion Vectors (MVs) to $t-1$ it is possible to generate the SI, denoted as Y. Y is then 4×4 DCT transformed. To calculate the soft input, in the form of Log-Likelihood ratios, for the LDPCA decoder the distribution of each DCT SI coefficient should be available. The estimation of the distribution [6] requires the knowledge of the difference $X-Y$. Since X is unknown at the decoder the difference is estimated by R, referred to as Residual, without using X. Once the soft-inputs are calculated the LDPCA decoder tries to decode using the first set of parity bits. If the solution is not acceptable either because it does not fit with the received parity bits or because the 8 bits CRC check fails, a new subset is requested via the feedback channel. The procedure is repeated until the solution is acceptable. When all the bitplanes belonging to a band are decoded the coefficients belonging to the band can be reconstructed. The bands are decoded in zig-zag order. After the k-th band is decoded, the IPDWZ (Inversely-transformed Partially Decoded Wyner-Ziv) frame $\hat{X}_{R_k}$ is available. It is the Inverse-DCT transform of the frame having for each 4×4 DCT block the first k coefficients reconstructed and the remaining $16-k$ coefficients are taken from the EX-OF generated SI Y. $\hat{X}_{R_k}$ is used to calculate the motion field between the frame at instant $t-1$ and itself, applying then the MVs to generate a second, refined SI, Y_{R_k}. Y_{R_k} is available for each coefficient k with $k > 1$. The refined motion field is also used to generate a refined residual estimation R_{R_k}. Therefore for each band $k > 1$ two sets of OF-based soft inputs are available for the LDCPA decoder, the first set calculated starting from Y and R, the second from Y_{R_k} and R_{R_k}. In order to leverage the robustness the block-based approaches have, a third SI is used: a block-based Extrapolated SI [13], denoted as EX-BB, is employed. Therefore the problem can be addressed in the MH framework proposed in [10, 13] using the 2 SIs (EX-BB, EX-OF) version for the DC coefficient and the 3 SIs (EX-BB, EX-OF, REF-OF) version for the remaining AC coefficients. In MH decoding, parallel LDPCA decoders are used, fed with the same parity bits but with different soft input sets, calculated using different weights [10, 13]. For each newly received chunk of bits the decoders are used and new bits are requested only in the case of none of them converging. If one of them converges it is taken as winning solution.

3. SI GENERATION, REFINEMENT AND FUSION

In the proposed architecture OF is employed for both SI generation and SI refinement. To generate the SI, employing the EX-OF method, the decoder uses two previously decoded frames, F_{t-1} and F_{t-2}, at instants $t-1$ and $t-2$, respectively. To generate the refined SI, using OF-REF, F_{t-1} and $\hat{X}_{R_k}$ are used. Y_{R_k} and R_{R_k} are used to decode the $k+1$-th DCT band.

3.1. Optical Flow-based SI Generation

The first step in the EX-OF method is the calculation of the apparent motion between F_{t-1} and F_{t-2}. In general the goal of every motion estimation system is to find the motion field v for every pixel position $\mathbf{x}$ in F_{t-1}, minimizing the data fidelity term, in this context also called OF constraint, denoted as $C(\mathbf{x}, v)$.

$$C(\mathbf{x}, v) = |F_{t-2}(\mathbf{x} + v(\mathbf{x})) - F_{t-1}(\mathbf{x})|. \quad (1)$$

The minimization of (1) is a non well-defined problem, because while the constraint is one, two variables are required to fully characterize the motion field: $v(\mathbf{x}) = [v_1(\mathbf{x}), v_2(\mathbf{x})] \in \mathbb{R}^2$. Therefore a second constraint should be introduced to make the problem well-posed. A well-known solution to this issue is the TV-L^1 OF formulation [15], which uses the L^1 norm of the OF constraint (1) and a Total Variation (TV) regularization term. The chosen regularization term is the 1-Jacobian of the motion field, $J_1(v(\mathbf{x}))$. Therefore the term which has to be minimized in the TV-L^1 is the Energy $E(v)$:

$$E(v) = \int \lambda C(\mathbf{x}, v) + |J_1(v(\mathbf{x}))| d\mathbf{x}, \quad (2)$$

where λ is a weighting factor, allowing to establish a trade-off between a small data fidelity term and a small regularization term (i.e. a smoother motion field). To minimize (2) the implementation outlined in [16] for asymmetrical OF is employed. The implementation relies on an iterative algorithm, which minimizes (2) in a coarse-to-fine pyramid. For all the sequences and all the RD points $\lambda = 20$, the number of pyramid levels is 70. Once the motion field v is calculated the motion is projected towards the time instant t. Given a point $\mathbf{x}$ in F_{t-1}, corresponding to the point $\mathbf{x} + v(\mathbf{x})$ in F_{t-2}, linear motion is assumed, therefore the corresponding point in the instant t has coordinates $\mathbf{x}' = \mathbf{x} - v(\mathbf{x})$, see Fig. 2. This assumption is used to project the value z of the position $\mathbf{x}$ to the point $\mathbf{x}'$.

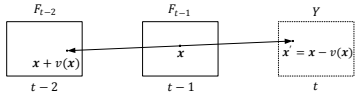


Fig. 2: Linear motion assumption and pixel projection, EX-OF method.

In general $\mathbf{x}' = [x'_1, x'_2] \in \mathbb{R}^2$ is not a position corresponding to a particular pixel, i.e. $\mathbf{x}'$ is not an integer position. To calculate the pixel values in the integer positions it is proposed to address the problem using scattered set interpolation techniques, in particular linear triangular interpolation [17] is employed. The linear triangular interpolation can be divided in two steps: 1) Triangulation of the scattered set; 2) Interpolation. The set is constituted by the points $[x'_1, x'_2, z] \in \mathbb{R}^3$, the triangulation step covers the scattered set with a piecewise triangular continuous surface, whose nodes are the points $[x'_1, x'_2, z]$ and the triangles are joined only by the edges. Among the various triangulation methods the Delaunay [17] triangulation is used. The interpolation step follows: for each integer coordinates point a bivariate linear interpolation is applied to the triangle the point belongs to. Basically a surface is fitted to the three nodes defining the triangle and then interpolation is performed. The residual is also calculated through triangular interpolation, in this case the points in the scattered set are structured as $[x'_1, x'_2, C(\mathbf{x}, v)]$.

3.2. Optical Flow-based SI Refinement

When the IPDWZ frame, $\hat{X}_{R_k}$, is available it can be used as starting point for the OF calculation in the REF-OF SI generation. $\mathbf{x}$ now is an integer position belonging to the IPDWZ frame, see Fig. 3. A new motion field, v_2, is calculated, using

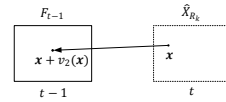


Fig. 3: SI Refinement, REF-OF.

as OF constraint

$$C_2(\mathbf{x}, v_2) = |F_{t-1}(\mathbf{x} + v_2(\mathbf{x})) - \hat{X}_{R_k}(\mathbf{x})|, \quad (3)$$

and solving it using the same algorithm and parameters used for (2). The refined SI and residual are then obtained by:

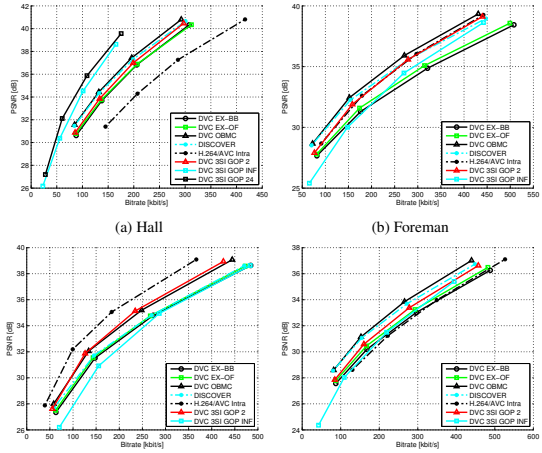
$$Y_{R_k}(\mathbf{x}) = F_{t-1}(\mathbf{x} + v_2(\mathbf{x})), \quad R_{R_k}(\mathbf{x}) = C_2(\mathbf{x}, v_2). \quad (4)$$

3.3. SI fusion

In this work the MH paradigm proposed in [10] is employed for efficient and robust SI fusion. For a given AC coefficient ($k > 1$) three distributions are calculated, one for each available SI: EX-OF, EX-BB, REF-OF. The three distributions are fused in various ways, using 6 sets of weights (provided in [10]). Each of the 6 fused distributions is used to generate the soft input for the LDPCA decoder. Therefore there are 6 different soft input arrays, one for each fused distribution, each array is fed into a different LDPCA decoder. Convergence is declared for the given bitplane when one of the LDPCA decoders converges and the solution is successfully checked against an 8-bits CRC check. The distribution related to the decoder which achieved convergence is then used for SI reconstruction as outlined in [10]. For DC coefficients only two SIs are used (EX-OF, EX-BB), weights and parameters can be found in [10].

4. EXPERIMENTAL RESULTS

For the experiments, the quantization matrices of the DISCOVER project [18] are used. Unless otherwise specified GOP 2 is used for all the DVC-based codecs, which all rely on the same coder. The sequences used (15 fps, QCIF resolution) are: *Hall*, *Foreman*, *Soccer*, *Coast* and the corresponding RD curves are reported in Fig. 4. The RD curves are produced varying the quantization matrices (Q_1, Q_4, Q_7, Q_8) used for the WZ frames and the corresponding QP for



(a) Hall (b) Foreman

(c) Soccer (d) Coast

Fig. 4: RD curves for the chosen test sequences, 15 fps, GOP 2, except in the DVC 3SI GOP INF case.

the KFs, which are H.264/AVC Intra coded following the DISCOVER specifications [18]. The RD curves of a completely intra-coded stream are also presented and denoted as H.264/AVC Intra. Four single SI DVC-based decoders are presented, namely DVC EX-BB, DVC EX-OF, DVC OBMC and DISCOVER. DISCOVER is fully specified in [5], DVC OBMC is presented in [6], they are both interpolation-based systems. DVC EX-BB and DVC EX-OF both use the architecture presented in [6] coupled with the respective SI generation system and they are extrapolation-based systems. EX-BB is used in [13] and it is similar to the one presented in [7]. When comparing the performance of the extrapolation-based systems, DVC EX-OF is able to outperform DVC EX-BB in medium motion speed sequences (*Foreman* and *Coast*) but it is not able to deliver appreciable gains in the other two cases. DVC OBMC and DISCOVER are superior to all the Extrapolation, single SI DVC codecs but they are characterized by higher delay due to Interpolation. The solution denoted as DVC 3SI GOP 2 uses the MH approach to fuse EX-BB, EX-OF and REF-OF, the latter used only for AC coefficients. The DVC 3SI GOP 2 decoder is characterized by high gains when compared with the state-of-the-art EX-BB [7], such gains, expressed by Bjøntegaard PSNR differences [19], ranging from 0.4 dB for the *Hall* sequence to 1.3 dB for the *Foreman* sequence. When comparing the proposed solution with interpolation-based codecs, DVC 3SI GOP 2 is able to provide similar performance to OBMC (an advanced Interpolation based method) for *Soccer* and for the higher rates of *Foreman*. When compared with the DISCOVER codec, characterized by a simpler SI generation and noise modeling, the proposed system achieve superior performance for *Soccer* and similar performance for the higher RD points of the other three sequences. Lastly, for the solution DVC 3SI GOP INF only the first and second frames are H.264 Intra encoded, all the rest are WZ coded frames, the other details are the same used for DVC 3SI GOP 2. While the RD performance drops for all the sequences displaying high motion activity, *Hall* is able to provide significant gains over the other proposed solutions using GOP2. For *Hall* a GOP size of 24 (DVC 3SI GOP 24, Fig. 4a) is able to achieve superior performance due to the balance between the use of WZ frames and KFs. It has to be noted that *Hall* is a typical video surveillance sequence, which is one of the potential application scenarios proposed for DVC.

5. CONCLUSION

In this work, a novel OF-based Extrapolation and OF-based SI refinement methods have been explored in DVC. They have been tested, along with a state-of-the-art block-based Extrapolation SI in a MH decoder, achieving performance comparable, in the low and medium motion case, with a state-of-the-art Interpolation-based codec, without any modification to the low-complexity, original encoder. The use of only WZ frames proved to lead to improved RD performance in a typical video surveillance sequence: *Hall*. Future evolutions of this work will investigate the possibility of using variable GOP sizes for DVC to achieve superior RD performance with lower (average) delay.

6. REFERENCES

[1] D. Slepian and J. Wolf, "Noiseless Coding of Correlated Information Sources," *IEEE Trans. Inform. Theory*, vol. 19, no. 4, pp. 471–480, July 1973.

[2] A. D. Wyner and J. Ziv, "The Rate-Distortion Function for Source Coding with Side Information at the Decoder," *IEEE Trans. Inform. Theory*, vol. 22, pp. 1–10, 1976.

[3] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed Video Coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 71–83, January 2005.

[4] R. Puri, A. Majumdar, and K. Ramchandran, "PRISM: A Video Coding Paradigm With Motion Estimation at the Decoder," *IEEE Transactions on Image Processing*, vol. 16, no. 10, pp. 2436–2448, October 2007.

[5] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER Codec: Architecture, Techniques And Evaluation," *Proc. of PCS*, November 2007.

[6] X. Huang and S. Forchhammer, "Cross-Band Noise Model Refinement for Transform Domain Wyner-Ziv Video Coding," *Signal Processing: Image Communication*, vol. 27, no. 1, pp. 16–30, 2012.

[7] L. Natário, C. Brites, J. Ascenso, and F. Pereira, "Extrapolating Side Information for Low-delay Pixel-domain Distributed Video Coding," in *Proc. of the 9th International Conference on Visual Content Processing and Representation*, 2005, pp. 16–21.

[8] António Tomé and Fernando Pereira, "Low Delay Distributed Video Coding with Refined Side Information ," *Signal Processing: Image Communication*, vol. 26, no. 45, pp. 220 – 235, 2011.

[9] J. Skorupa, J. Slowack, S. Mys, N. Deligiannis, J. De Cock, P. Lambert, C. Grecos, A. Munteanu, and R. Van De Walle, "Efficient low-delay distributed video coding," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 22, no. 4, pp. 530–544, 2012.

[10] X. Huang, L.L. Raket, H. V. Luong, M. Nielsen, F. Lauze, and S. Forchhammer, "Multi-Hypothesis Transform Domain Wyner-Ziv Video Coding Including Optical Flow," in *MMSP 2011*, October 2011, pp. 1–6.

[11] M. Salmistraro, M. Zamarin, L. Lau Raket, and S. Forchhammer, "Distributed Multi-Hypothesis Coding of Depth Maps Using Texture Motion Information and Optical Flow," in *ICASSP 2013*, May 2013, pp. 1685–1689.

[12] C. Brites, J. Ascenso, and F. Pereira, "Learning Based Decoding Approach for Improved Wyner-Ziv Video Coding," in *PCS 2012*, May 2012, pp. 165–168.

[13] X. Huang, C. Brites, J. Ascenso, F. Pereira, and S. Forchhammer, "Distributed Video Coding with Multiple Side Information," in *PCS 2009*, Piscataway, NJ, USA, 2009, PCS 2009, pp. 385–388, IEEE Press.

[14] D. Varodayan, A. Aaron, and B. Girod, "Rate-Adaptive Codes for Distributed Source Coding," *EURASIP Signal Processing Journal*, vol. 86, no. 11, pp. 3123–3130, November 2006.

[15] C. Zach, T. Pock, and H. Bischof, "A Duality Based Approach for Realtime TV-L1 Optical Flow," in *Proceedings of the 29th DAGM Conference on Pattern Recognition*, Berlin, Heidelberg, 2007, pp. 214–223, Springer-Verlag.

[16] L.L. Rakêt, L. Roholm, A. Bruhn, and J. Weickert, "Motion Compensated Frame Interpolation with a Symmetric Optical Flow Constraint," in *Advances in Visual Computing*, vol. 7431 of *Lecture Notes in Computer Science*, pp. 447–457. Springer Berlin Heidelberg, 2012.

[17] Isaac Amidror, "Scattered Data Interpolation Methods for Electronic Imaging Systems: a Survey," *Journal of Electronic Imaging*, vol. 11, no. 2, pp. 157, Apr. 2002.

[18] "[Online] DISCOVER Project Test Conditions," http://www.img.lx.it.pt/~discover/test_conditions.html, December 2007.

[19] G. Bjøntegaard, "Calculation of Average PSNR Differences Between RD Curves," in *ITU-T Q6/SG16, Doc. VCEG-M33, in: 13th Meeting, Austin, USA, April*, 2001.

EDGE-PRESERVING INTRA DEPTH CODING BASED ON CONTEXT-CODING AND H.264/AVC

Marco Zamarin[1], *Matteo Salmistraro*[1], *Søren Forchhammer*[1], *Antonio Ortega*[2]

[1] Dept. of Photonics Eng., Technical University of Denmark, Denmark
[2] Ming Hsieh Dept. of Electrical Eng., University of Southern California, CA (USA)
{mzam, matsl, sofo}@fotonik.dtu.dk, ortega@sipi.usc.edu

ABSTRACT

Depth map coding plays a crucial role in 3D Video communication systems based on the "Multi-view Video plus Depth" representation as view synthesis performance is strongly affected by the accuracy of depth information, especially at edges in the depth map image. In this paper an efficient algorithm for edge-preserving intra depth compression based on H.264/AVC is presented. The proposed method introduces a new Intra mode specifically targeted to depth macroblocks with arbitrarily shaped edges, which are typically not efficiently represented by DCT. Edge macroblocks are partitioned into two regions each approximated by a flat surface. Edge information is encoded by means of context-coding with an adaptive template. As a novel element, the proposed method allows exploiting the edge structure of previously encoded edge macroblocks during the context-coding step to further increase compression performance. Experiments show that the proposed Intra mode can improve view synthesis performance: average Bjøntegaard bit rate savings of 25% have been reported over a standard H.264/AVC Intra coder.

Index Terms— Block-based depth compression, context-coding, edge-based depth representation, video-plus-depth, depth-image-based-rendering.

1. INTRODUCTION

In the recent years the interest in three-dimensional (3D) video technologies has grown considerably in both the academic and industrial worlds. A number of 3D-capable solutions and products are becoming available on the consumer market. One of the key challenges in the implementation of a 3D Video communication system is the decoupling of the capture and transmission format from the display format in order to allow a multitude of acquisition, transmission, and display devices to work together seamlessly [1]. One representation that enables such decoupling is the so-called "Multi-view Video plus Depth" (MVD), in which depth or disparity information is provided together with typically 2 or 3 views. By using depth information together with the input views, the desired output views can be synthesized at the decoder side thus allowing different 3D display devices to operate properly. Efficient coding solutions based on the MVD format are currently being developed by the 3DV group of MPEG [2]. Solutions compatible with both the current H.264/AVC and the upcoming High Efficiency Video Coding (HEVC) standards are being investigated.

As view synthesis algorithms based on Depth-Image-Bases-Rendering (DIBR) typically show a high sensitivity to depth inaccuracies [3], depth coding plays a crucial role in the development of an effective 3D Video system based on the MVD format. Specifically, depth edges should be preserved in order to avoid the appearance of annoying unnatural artifacts in the synthesized views. Due to the fact that standard DCT-based approaches fail to efficiently represent sharp edges, a number of specialized algorithms have been proposed in the literature to cope with this problem, e.g. based on edge-adaptive transforms and transform domain sparsification [4] or local explicit coding of edge information in blocks with sharp discontinuities [5, 6]. An overview of some recent works is provided in the next section.

In this paper a novel algorithm for efficient edge-aware intra depth coding is presented. The proposed scheme is based on the H.264/AVC Intra framework and operates at a MacroBlock (MB) level. While DCT fails to efficiently represent blocks with arbitrarily shaped edges, it provides very compact representations in the case of blocks with (nearly-)uniform values or smooth gradients. Therefore, we propose to modify the reference encoder by adding one Intra mode specifically targeted to edge MBs. When the new mode is selected, the MB is partitioned into two regions and a constant value is assigned to each region and encoded. Partitioning information also needs to be transmitted in order to allow for a proper reconstruction. This is done by encoding a per-pixel binary mask by means of context-coding. The proposed algorithm allows exploiting previously-encoded edge MBs in order to predict the constant values of the current MB being encoded. Moreover, the binary masks of adjacent edge MBs can be combined and jointly context-coded to allow further bit rate savings.

M. Zamarin, M. Salmistraro, S. Forchhammer, A. Ortega, "Edge-preserving Intra Depth Coding based on Context-coding and H.264/AVC", *Proc. of 2013 IEEE Int'l Conf. on Multimedia and Expo (ICME 2013)*, pp. 1 - 6, San Jose, CA, USA, July 15-19, 2013.

The remainder of this paper is organized as follows. In Section 2 related works on edge-based depth compression are briefly discussed. Section 3 describes the proposed coding method highlighting the novelties it introduces. Section 4 discusses the coding performance of the proposed framework on a number of 3D video test sequences.

2. RELATED WORK

As mentioned in Sec. 1, a number of edge-preserving depth coding algorithms have recently been proposed in the literature. Most of them are motivated by the fact that edge information is critical for the achievement of high view synthesis performance and therefore should be preserved in the coding process.

Shen *et al.* [7] introduced a set of edge-adaptive transforms for block-based depth coding. Edge detection was used to identify depth discontinuities and define a local graph structure to be transformed. Even though some view synthesis improvements were achieved, the explicit calculation of the eigenvectors of the graph Laplacian made the algorithm not suitable for fast implementations. Cheung *et al.* [8, 9] introduced the concept of *don't care region*, exploited to obtain sparse depth representations and improve the coding performance of a JPEG coder. The concept was then combined with the edge-adaptive transforms previously mentioned to achieve further improvements [4]. In [10] Kim *et al.* proposed to use graph based transforms as an alternative to DCT for 4×4 edge blocks within the H.264/AVC framework. Significant bit rate savings are reported, but again the calculation of the graph Laplacian caused a considerable increase of the overall algorithm complexity. Morvan *et al.* [11] proposed to extend the wedgelets - which locally approximate a signal with two constant functions separated by a straight line - with piecewise-linear functions (platelets) defined using a quadtree decomposition, reporting good results in terms of depth compression and edge preservation.

Block-based depth coding exploiting co-located texture edge information has been proposed by Merkle *et al.* in [12] for the case of HEVC. When texture is exploited to bi-partite a depth edge block, a constant depth value is assigned to each partition. As the same depth partitioning can be reproduced at the decoder side, only the two constant values need to be encoded. Depth image coding exploiting texture edge information has also been proposed by Milani *et al.* in [13]. In this case, segmentation of reconstructed texture data is exploited to predict shapes in the depth image. Within each segment the depth signal is approximated by a linear function and only the corresponding coefficients are transmitted, thus avoiding the explicit encoding of edge information.

Shimizu *et al.* [5] proposed a depth coding scheme based on H.264/AVC similar to the one proposed in this paper: edge macroblocks are approximated by a palette with two entries and a (binary) object shape map, both predictively encoded exploiting intra/inter correlation. Shape maps are generated by minimizing a prediction error function and encoded by means of context-adaptive binary arithmetic coding as in shape coding in MPEG-4 Part 2. Even though intra/inter neighboring blocks are exploited in order to properly define the context at any pixel position, the edge structure of previously encoded edge macroblocks is not directly exploited to improve the coding efficiency of context coding. If the edge structure correlation of neighboring edge macroblocks is taken into account as in the method proposed in this paper, lower bit rates can be achieved for the coding of shape information, which can be of benefit especially at low bit rates.

A different approach based on a similar framework has been proposed for the case of HEVC by Lan *et al.* in [6]. In this case a wider range of block sizes are exploited, namely from 32×32 down to 4×4. Blocks are partitioned in up to 8 regions each approximated by a flat surface. Arithmetic coding is then employed to encode the region map and surface values. Significant gains are reported for the cases of both estimated and acquired depth sequences. However, higher performance are expected if the region map overhead is reduced. This can be achieved by exploiting the correlation between the edge structures of the current and previously encoded region maps during the encoding process as proposed in this work.

3. PROPOSED METHOD

As mentioned in Sec. 1, a new Intra mode for H.264/AVC - referred to as "EDGE" in the following - able to efficiently represent arbitrarily-shaped edges is proposed. Differently from other approaches, the proposed mode aims at combining the encoding of depth discontinuities that span multiple adjacent MBs in order to increase the coding efficiency of edge information. The method operates at a MB level as part of the Rate-Distorion (RD) optimization strategy: the EDGE mode is tested together with the standard intra modes and the mode with the lowest RD cost is selected.

The EDGE mode partitions the whole depth MB into two regions and associates one Constant Value (CV) to each of them. A per-pixel binary map M is used to identify pixels belonging to the two regions. In order to allow a perfect reconstruction at the decoder side, the mask M is losslessly encoded by means of context-coding together with the two CVs. A detailed description of the EDGE algorithm for a depth MB is provided below (see Fig. 1).

1. The input depth MB is partitioned into two regions. A point in 3D is defined for each pixel from its spatial coordinates and depth value. 3D-points are connected with the closest (in an Euclidean sense) n neighbors, n being a parameter initially set to 1. If more than two disconnected components are formed, n is increased and the procedure iterated. A 16×16 binary mask M

is defined.

2. One CV is associated to each region of M by selecting the median value of the corresponding pixels in the input depth MB.[1]
3. The mask M is losslessly encoded by means of context-based arithmetic coding (see Subsection 3.1 for more details). Let R_M be the number of bits spent to encode M, R_{CV} the number of bits spent to encode the two CVs, and $R = R_M + R_{CV}$ the total number of bits.
4. The distortion D is computed as MSE between the input depth MB and the binary EDGE MB.
5. The EDGE RD cost is computed as follows: $RD^{intra}_{EDGE} = D + \lambda \cdot R$, λ being the same Lagrangian multiplier used in the RD cost calculation of the standard Intra modes.
6. If the left-neighboring MB is available and encoded as EDGE MB:
   (a) Steps 3-5 are repeated using the same CVs used in the left-neighboring MB. In this way no bits are spent to encode the CVs for the current MB. Let RD^{left1}_{EDGE} be the corresponding RD cost.
   (b) Steps 3-5 are repeated using the same CVs and coding parameters for the mask M used in the left-neighboring MB (see Subsection 3.1). Let RD^{left2}_{EDGE} be the corresponding RD cost.
7. As Step 6 but using the top-neighboring MB. If it is available and encoded as EDGE MB, the costs RD^{top1}_{EDGE} and RD^{top2}_{EDGE} are defined.
8. The RD_{EDGE} cost is defined as the lowest cost among RD^{left1}_{EDGE}, RD^{left2}_{EDGE}, RD^{top1}_{EDGE}, RD^{top2}_{EDGE} (if available), and RD^{intra}_{EDGE}.

If RD_{EDGE} is lower than the RD cost obtained with standard H.264/AVC Intra coding, the EDGE mode is selected and the corresponding data (mask M, CVs and coding parameters) are included in the bit stream. At the decoder side, if the usage of the EDGE mode is detected, the binary mask is decoded by means of context-based decoding, CVs are decoded from the bit stream and the output EDGE MB is produced. If the EDGE mode is not used, standard Intra decoding is performed.

3.1. Binary mask encoding

The encoding of a binary mask M is done by means of context-based arithmetic coding. The encoding can be done in two different ways: *intra* (i.e. without exploiting previously-encoded EDGE MBs) and *inter* (i.e. exploiting previously-encoded EDGE MBs in the same slice). The two methods are described here in detail.

[1]The choice of the median over the mean - even though not necessarily optimal from a MSE point of view - reduces the influence of outliers, i.e. noisy pixels in the MB partition.

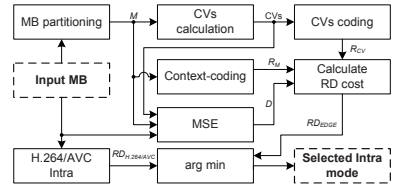


Fig. 1: Encoder block scheme

Fig. 2: Template selection: pixels p_1 and p_2 are fixed; one additional pixel among p_3, p_4, and p_5 can be included in the template. "?" indicates the current pixel being encoded.

3.1.1. *Intra EDGE coding*

In order to make the context prediction efficient, it is important to select the most relevant template pixels for the edge structure of the mask M being encoded. For this reason, an adaptive template inspired by [14] is used. The template is defined among a set of 5 candidate pixels (see Fig. 2) and has thus a smaller size than the one used in [14] and in MPEG-4 Part 2 shape coding. As shown in Fig. 2, the template includes the pixels on the left and on top of the one being encoded. A third pixel chosen among a set of three candidates can be included if of benefit in terms of code length: the binary mask is encoded 4 times - with the 2-pixel template and with the three 3-pixel templates - and the template providing the lower rate is selected.

In order to allow the decoder to correctly decode the mask M, the selected template is explicitly signalled in the bit stream. This is done by encoding two parameters: the number of pixels in the template and, in case of a 3-pixel template, the third pixel index. The bits needed to encode these parameters (referred to as $R_{\#}$ and R_p, respectively) are also considered in the template selection process.

For the *intra* case, the EDGE rate R introduced in Section 3, Step 3 is therefore given by

$$R^{intra} = R_{\#} + R_p + R_M + R_{CV} + R^{i/p}_{flag}, \qquad (1)$$

where $R^{i/p}_{flag}$ is the rate of a flag encoded only when at least one between the left and top neighboring MBs is available and encoded as EDGE MB, to signal whether *intra* on *inter* EDGE coding is used.

3.1.2. *Inter EDGE coding*

As specified in Section 3, Steps 6 and 7, if the left or top neighboring MB is available and encoded as EDGE MB, the

current MB can be inter encoded. This can be done in two ways referred to as *partial inter coding* and *full inter coding*, corresponding to Steps 6a-6b, and 7a-7b, respectively.

In the first case the encoding is done as in the intra case with the only difference that the CVs are not calculated and encoded but simply copied from the predicting MB, under the assumption that adjacent parts of the same edge refer to the same background and foreground objects. In this case the EDGE rate is given by

$$R^{p.inter} = R_{\#} + R_p + R_M + R^{i/p}_{flag} + R^{p/f}_{flag} + R^{l/t}_{flag}, \quad (2)$$

where $R^{p/f}_{flag}$ is the rate of a flag specifying whether *partial* or *full* inter coding is used, and $R^{l/t}_{flag}$ is the rate of a flag indicating which of the two neighboring MBs is used for prediction when both of them are available and encoded as EDGE MBs.

When *full inter coding* is used, together with the CVs also the template pixels are copied from the predicting MB. Moreover, since the context-coding will be based on the same template, it is possible to use the edge statistics of the predicting MB - which are stored for each EDGE MB after the encoding of the mask M - as a starting point for the context-coding, thus allowing to exploit the statistics of the edge in the predicting MB. When *full inter coding* is used, adjacent blocks are therefore encoded as part of a unique binary mask. Note that edge statistics do not need to be explicitly transmitted as they can be generated during the decoding process. Finally, in order to properly initialize the boundaries of the binary mask M for the context coding step, the encoded binary mask of the predicting MB is used (the same is done for the *partial inter coding* too). In case of *full inter coding* the EDGE rate is given by

$$R^{f.inter} = R_M + R^{i/p}_{flag} + R^{p/f}_{flag} + R^{l/t}_{flag}. \quad (3)$$

The availability of these two inter modes allows for efficient representation of depth MBs that separate between the same background and foreground of the previous MB (i.e. they share the same CVs) and present different edge direction/structure (*partial inter coding*) or similar structure (*full inter coding*).

4. EXPERIMENTAL RESULTS

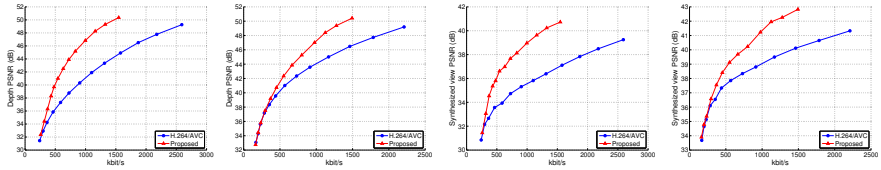
The proposed method has been implemented on the H.264/AVC reference software JM 17.1 and evaluated on the following 3D video sequences: *Ballet* and *Breakdancers* [15], *Book Arrival*, *Lovebird1*, *Dancer*, *Cafe*. The first 15 frames of each sequence have been encoded. Two views have been selected for each sequence (reported in Table 1) and the corresponding depth images have been encoded with both H.264/AVC Intra and the proposed method. Depth maps have been encoded with every second QP in the interval (24, 48). All the standard Intra modes of H.264/AVC have been enabled and RD-based mode decision was selected for the Intra 16×16 mode. In order to avoid introducing blurring artifacts, the deblocking filter has been disabled. In order to evaluate the effectiveness of the proposed algorithm, not only the RD performance on the depth signals have been evaluated (Figs. 3a and 3b for the sequences *Ballet* and *Breakdancers*), but also the view synthesis performance using reconstructed depth data and uncompressed texture as done in [16] (Figs. 3c and 3d) have been considered. In this case, PSNR values are measured against reference virtual views synthesized from uncompressed depth and texture data. Virtual views have been synthesized using the MPEG VSRS 3.5.

Table 1: Sequence resolutions and input left (L), virtual (V) and right (R) views.

Sequence	Resolution	L	V	R
Ballet	1024×768	5	4	3
Breakdancers	1024×768	5	4	3
Book Arrival	1024×768	8	7	6
Lovebird1	1024×768	6	7	8
Dancer	1920×1088	2	3	5
Cafe	1920×1088	2	3	4

Table 2 reports the Bjøntegaard bit rate savings [17] between the proposed method and H.264/AVC for both the depth signals and the sythesized views. As it can be noticed, when comparing the view synthesis performance the proposed method improves depth compression efficiency for all the sequences, confirming that it is of benefit in a MVD scenario. However, the improvement depends on the depth accuracy of the particular test sequence and on its resolution. Major bit rate savings in terms of both depth coding and view synthesis performance are reported in the case of *Ballet* and *Breakdancers* due to the sharp and clean edges that characterize depth images of these two sequences. Experimental results show that the proposed *inter EDGE coding* is often selected: the binary masks of adjacent edge macroblocks are therefore jointly encoded as part of a single binary image allowing additional bit rate savings (see Fig. 4b for an example). *Book Arrival* shows a comparable bit rate reduction in terms of view synthesis, but a smaller gain in terms of depth compression due to the presence of less sharp object discontinuities in the depth images which favor DCT coding. In the case of *Lovebird1*, a smaller gain in terms of synthesis performance is observed, together with a minor increase of the depth bit rate. This sequence shows that even though the proposed algorithm might not always be more efficient than H.264/AVC in terms of depth MSE, it does provide a benefit when reconstructed depth data are used for view synthesis.

The performance on the two high definition sequences are also satisfactory but lower gains are noticed, even for the *Dancer* sequence which features high quality synthetic depth data. One reason can be found in the fact that edge mac-

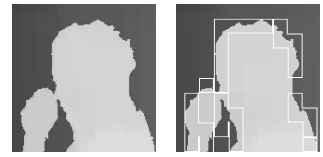


(a) *Ballet*: depth PSNR vs. depth bit rate (b) *Breakdancers*: depth PSNR vs. depth bit rate (c) *Ballet*: synth. view PSNR vs. depth bit rate (d) *Breakdancers*: synth. view PSNR vs. depth bit rate

Fig. 3: Rate-Distortion performance comparisons

Table 2: Bjøntegaard bit rate savings (BDBR) between the proposed method and H.264/AVC for depth signals and corresponding synthesized views. Note that the minus sign means bit rate reduction.

Sequence	Depth BDBR (%)	Synth. BDBR (%)
Ballet	-38.33	-46.82
Breakdancers	-21.94	-34.54
Book Arrival	-7.91	-34.39
Lovebird1	1.45	-10.78
Dancer	-3.60	-20.84
Cafe	0.45	-4.21
Average	**-11.65**	**-25.26**



(a) (b)

Fig. 4: Reconstructed depth comparison: detail of the Ballet sequence, left view, frame 0. (a) H.264/AVC Intra, 38.85 dB @ 361 kbit/s, (b) Proposed, 43.83 dB @ 366 kbit/s. White squares indicate MBs that have been encoded as EDGE MBs. EDGE MBs encoded by means of EDGE prediction are connected to the corresponding prediction MBs.

roblocks in high resolution depth images present on average less arbitrarily shaped discontinuities. Therefore, the standard H.264/AVC Intra modes can predict well the edge structures most of the times. For high resolution data, bigger block sizes - such as those defined in the upcoming HEVC standard - would hence be beneficial for the proposed algorithm especially at very low bit rates, where the overhead due to DCT/EDGE signalling can significantly affect the RD performance (as in the *Dancer*, *Cafe*, *Book Arrival*, and *Lovebird1* sequences in which H.264/AVC slightly outperforms the proposed method at very low bit rates).

Figures 4 and 5 provide a visual quality comparison between H.264/AVC Intra and the proposed method for both depth and synthesized data at similar bit rates, highlighting the advantages of the latter over the former.

Coding performance considering total bit rate (texture plus depth) versus synthesized view PSNR have also been evaluated, as done in [18]. Even though a direct comparison with the plane segmentation based coding method in [18] is not possible due to different reference settings and test conditions, the proposed method shows similar Bjøntegaard bit rate savings over the reference encoder for the common sequences, ranging between -11.50% and -3.98%.

Finally, Fig. 6 compares the RD performance of the proposed approach in terms of depth quality against H.264/AVC, JPEG2000, the method proposed by Milani *et al.* in [13], and the Platelet-based algorithm proposed by Morvan *et al.* in [11]. The comparison is made on the *Teddy* depth image from the stereo sets at [19]. As it can be noticed, the proposed method is able to outperform H.264/AVC Intra, the planar fitting method proposed in [13], which exploits reconstructed texture data in order to predict depth shapes, and the Platelet-based scheme. JPEG2000 appears as the least efficient still image coder in the case of depth images due to the heavy ringing artifacts it introduces.

Compared with Shimizu *et al.* [5] in which the number of bits needed to encode a shape/binary mask is around 50 per MB, the proposed method exhibits a more flexible behavior thanks to the *intra/inter EDGE coding* methods. The total number of bits needed to encode an EDGE MB (including binary mask, constant values, and flags) ranges from an average of 51 at high rates to an average of 45 at low rates, where *inter EDGE coding* is often selected.

5. CONCLUSIONS AND FUTURE WORK

In this paper a novel edge-preserving H.264/AVC Intra mode for efficient depth coding has been presented. Edge macroblocks are partitioned in two regions each approximated

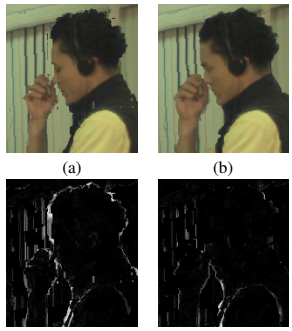


Fig. 5: Visual quality comparison versus total depth bit rate: detail of the synthesized view from uncompressed texture, Ballet sequence, frame 0. H.264/AVC Intra, 34.73 dB @ 726 kbit/s (a), Proposed, 37.67 dB @ 723 kbit/s (b), absolute errors on the luminance components (with same scale) between synthesized views obtained with uncompressed depth data and H.264/AVC Intra (c) and Proposed (d).

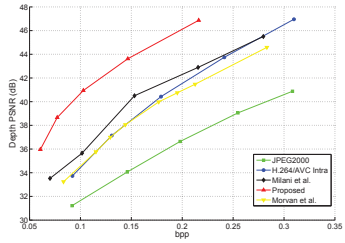


Fig. 6: *Teddy* depth image: RD performance comparison

by a flat surface. A binary mask identifying the two regions is defined and encoded by means of adaptive context-coding exploiting previously encoded edge macroblocks to increase the compression efficiency. Experimental results show that adjacent edge macroblocks can be jointly encoded in an efficient manner. Compared with a standard H.264/AVC Intra coder, the proposed algorithm achieved average Bjøntegaard bit rate savings of about 12% in terms of depth compression, and about 25% in terms of synthesized view quality versus depth bit rate. The complexities of the proposed and standard Intra modes are comparable, thus the overall complexity is not affected significantly. Future developments include binary mask pre-filtering for efficient low bit rate coding, prediction from non-EDGE macroblocks and quantization of flat surface values, and extension to depth video coding.

6. REFERENCES

[1] L. Onural, *3D Video Technologies: An Overview of Research Trends*, Society of Photo Optical, 2010.

[2] ISO/IEC JTC1/SC29/WG11, "Applications and Requirements on 3D Video Coding," Doc. N12035, Geneva (CH), Mar. 2011.

[3] K. Müller, P. Merkle, and T. Wiegand, "3-D Video Representation Using Depth Maps," *Proc. of the IEEE*, vol. 99, no. 4, pp. 643–656, Apr. 2011.

[4] G. Cheung, W.-S. Kim, A. Ortega, J. Ishida, and A. Kubota, "Depth Map Coding using Graph based Transform and Transform Domain Sparsification," in *IEEE MMSP 2011*, Oct. 2011, pp. 1–6.

[5] S. Shimizu, H. Kimata, S. Sugimoto, and N. Matsuura, "Block-adaptive Palette-based Prediction for Depth Map Coding," in *IEEE ICIP 2011*, Sep. 2011, pp. 117–120.

[6] C. Lan, J. Xu, and F. Wu, "Improving Depth Compression in HEVC by Pre/Post Processing," in *2012 Int'l Work. on Hot Topics in 3D Multim. (Hot3D 2012)*, July 2012, pp. 611–616.

[7] G. Shen, W.-S. Kim, S.K. Narang, A. Ortega, J. Lee, and H. Wey, "Edge-adaptive Transforms for Efficient Depth Map Coding," in *PCS 2010*, Dec. 2010, pp. 566–569.

[8] G. Cheung, A. Kubota, and A. Ortega, "Sparse Representation of Depth Maps for Efficient Transform Coding," in *Picture Coding Symposium (PCS) 2010*, Dec. 2010, pp. 298–301.

[9] G. Cheung, J. Ishida, A. Kubota, and A. Ortega, "Transform Domain Sparsification of Depth Maps using Iterative Quadratic Programming," in *IEEE ICIP 2011*, Sep. 2011, pp. 129–132.

[10] W.-S. Kim, S.K. Narang, and A. Ortega, "Graph based Transforms for Depth Video Coding," in *IEEE ICASSP 2012*, Mar. 2012, pp. 813–816.

[11] Y. Morvan, D. Farin, and P. de With, "Depth-Image Compression based on an R-D Optimized Quadtree Decomposition for the Transmission of Multiview Images," in *Proc. of IEEE ICIP 2007*, 2007.

[12] P. Merkle, C. Bartnik, K. Muller, D. Marpe, and T. Wiegand, "3D Video: Depth Coding based on Inter-component Prediction of Block Partitions," in *Picture Coding Symposium (PCS) 2012*, May 2012, pp. 149–152.

[13] S. Milani, P. Zanuttigh, M. Zamarin, and S. Forchhammer, "Efficient Depth Map Compression Exploiting Segmented Color Data," in *IEEE ICME 2011*, July 2011, pp. 1–6.

[14] M. Zamarin and S. Forchhammer, "Lossless Compression of Stereo Disparity Maps for 3D," in *2012 Int'l Work. on Hot Topics in 3D Multim. (Hot3D 2012)*, July 2012, pp. 617–622.

[15] L.C. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-Quality Video View Interpolation using a Layered Representation," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 600–608, 2004.

[16] P. Merkle, Y. Morvan, A. Smolic, D. Farin, K. Müller, P.H.N. de With, and T. Wiegand, "The Effects of Multiview Depth Video Compression on Multiview Rendering," *Image Commun.*, vol. 24, no. 1-2, pp. 73–88, 2009.

[17] G. Bjøntegaard, "Calculation of Average PSNR Differences between RD-Curves," VCEG-M33, Apr. 2001.

[18] B.T. Oh, H.-C. Wey, and D.-S. Park, "Plane segmentation based intra prediction for depth map coding," in *PCS 2012*, May 2012, pp. 41–44.

[19] "Repository vision.middlebury.edu: Stereo datasets," `http://vision.middlebury.edu/stereo`.

TEXTURE SIDE INFORMATION GENERATION FOR DISTRIBUTED CODING OF VIDEO-PLUS-DEPTH

Matteo Salmistraro[◊] *Lars Lau Rakêt*[⋆] *Marco Zamarin*[◊] *Anna Ukhanova*[◊] *Søren Forchhammer*[◊]

[◊]DTU Fotonik, Technical University of Denmark, Ørsteds Plads,
2800 Kgs. Lyngby, Denmark. Emails: {matsl, mzam, annuk, sofo}@fotonik.dtu.dk
[⋆]Department of Computer Science, University of Copenhagen, Universitetsparken 5,
2100 Copenhagen, Denmark. Email: larslau@diku.dk

ABSTRACT

We consider distributed video coding in a monoview video-plus-depth scenario, aiming at coding textures jointly with their corresponding depth stream. Distributed Video Coding (DVC) is a video coding paradigm in which the complexity is shifted from the encoder to the decoder. The Side Information (SI) generation is an important element of the decoder, since the SI is the estimation of the to-be-decoded frame. Depth maps enable the calculation of the distance of an object from the camera. The motion between depth frames and their corresponding texture frames (luminance and chrominance components) is strongly correlated, so the additional depth information may be used to generate more accurate SI for the texture stream, increasing the efficiency of the system. In this paper we propose various methods for accurate texture SI generation, comparing them with other state-of-the-art solutions. The proposed system achieves gains on the reference decoder up to 1.49 dB.

***Index Terms*—** Distributed Video Coding, Depth Map, Wyner-Ziv Coding, Optical Flow, Multi-Hypothesis

1. INTRODUCTION

In the recent years Distributed Video Coding (DVC) has received a great amount of interest, due to the possibility of shifting complexity from the encoder to the decoder.

In this paper we address DVC of video-plus-depth streams in a monoview scenario and propose methods to exploit the correlation between the streams in order to produce more accurate Side Information (SI). Depth maps can be used in single view scenarios for activity detection, object tracking and background/foreground separation [1].

DVC is based on two information theoretic results, the Slepian-Wolf [2] and Wyner-Ziv [3] (WZ) theorems, where, in the second case, source data are independently lossy coded but jointly decoded using a correlated source at the decoder, commonly referred to as SI. DVC could be an appealing solution for the video-plus-depth coding problem, in particular if we require low-complexity encoders. It is possible, in this way, to independently code the two streams and then jointly decode them. This is especially convenient when separated texture and depth cameras are used, in which case inter-camera communication is difficult or perhaps infeasible. The DVC decoder used as basis of our system is the one presented in [4], employing the approach first proposed in [5] and then improved in [6]. As can be seen in Fig. 1, the frames are divided into Key-Frames (KFs) and WZ frames at the encoder. The KFs are encoded independently with respect to each other and with respect to the WZ frames, using a H.264/AVC Intra coder. The KFs are used at the decoder to calculate the SI, which is a prediction of the to-be-decoded WZ frame. At the encoder the WZ frame is DCT-transformed, the coefficients are grouped and divided in bitplanes. Each bitplane is encoded using an LDPCA encoder [7], and a subset of the calculated syndromes is sent to the decoder. The decoder uses the syndromes to correct the errors in the corresponding SI bitplanes, bitplane by bitplane. If the syndromes are not enough, others are requested via a feedback channel. The LDCPA decoder also requires the calculation of the reliability of the bits of the bitplanes. Ideally, it is possible to calculate such reliability from the residual, which is the difference between the SI and the original WZ frame, but since WZ frames are not available at the decoder, a residual estimation method have to be devised.

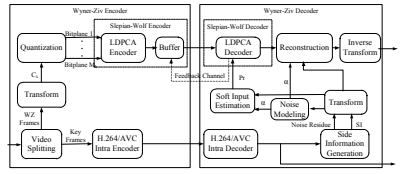


Fig. 1: DVC Codec [4].

Depth maps are images allowing the calculation of the distance of an object from the camera. While texture frames contain the luminance and chrominance components of the

M. Salmistraro, L.L. Rakêt, M. Zamarin, A. Ukhanova, S. Forchhammer, "Texture Side Information Generation for Distributed Coding of Video-Plus-Depth", *Proc. of 2013 IEEE Int'l Conf. on Image Processing (ICIP 2013)*, pp. 1699 - 1703, Melbourne, Australia, Sep. 15-18, 2013.

(a) Texture Frame (b) Depth Frame

Fig. 2: A texture frame (a) and its correspoding depth frame (b), from the Ballet [12] sequence.

scene (Fig. 2a), depth maps describe depth information (Fig. 2b). Depth information can be used to calculate the distance of a given point in the 3D scene from the camera. The depth and texture frames of the same scene, referring to the same time instant, are strongly correlated and the motion between texture frames is highly correlated with the motion between the depth frames [8]. This gives rise to the video-plus-depth coding problem, in which the redundancy between the two streams is used to achieve efficient coding, as for example in [8]. This approach has also been used in depth map coding architectures based on DVC [9, 10] where the depth motion estimation has been carried out exploiting texture data. In DVC for texture frames, depth data have been used to generate intra-view SI through view synthesis [11]. View synthesis can be used in Multiview DVC but it is not suitable for single view systems or in the cases in which the Rate-Distortion (RD) performance of the intra and inter-view SIs are too different to obtain improvements from the fusion of the SIs.

This paper proposes methods for exploiting depth maps in the texture SI generation. We consider that an independently coded depth stream is already available and it is used to improve the WZ coding performance of the texture stream. We introduce Optical Flow (OF) based techniques, extending the framework proposed in [13] and introducing a new OF technique based on two distinct data terms. We benchmark these techniques against well-known block-based systems. Finally we consider the use of a multi-hypothesis decoder [14] for efficient and robust SI fusion. OF-based SI generation [15, 16] has been previously used in DVC as a way to create accurate SI for texture streams. In this paper we use OF to extract accurate motion estimates from the depth stream. We also propose a joint stream calculation, taking into account, at the same time, both KFs and depth frames, employing an OF formulation with two constraints.

2. SIDE INFORMATION GENERATION

Let D_i and T_i, with temporal index i, denote depth and texture frames, respectively. The to-be-decoded frame is T_t, all the other frames in Fig. 3 are assumed to be known at the decoder. We estimate the motion between D_t and D_{t-1} and use it to motion compensate T_{t-1} obtaining Y_{t-1}. We also calculate the motion between D_t and D_{t+1}, then T_{t+1} is motion compensated obtaining Y_{t+1}.

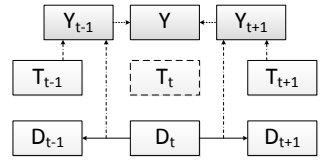


Fig. 3: The video stream structure, Group-Of-Pictures 2.

Once these two components have been calculated, the final SI Y can be calculated as their average, and the residual R can be calculated as their difference. We propose three new methods for this basic setup (Fig. 3).

The first two, "D2T BB" and "D2T OF", calculate the motion using the depth frames only, then this motion is used to motion compensate the texture frames, generating Y_{t-1} and Y_{t+1}. The difference between the two is the Motion Estimation (ME) algorithm: D2T BB uses a Block-Based (BB) method, while D2T OF uses an OF method.

For what concerns D2T BB, we consider the so-called "Adaptive Rood Pattern Search" (ARPS) ME algorithm [17]. While this approach may not provide the lowest Mean-Squared-Error between the motion compensated depth frame and the original one on average, it is able to capture the motion between the frames in a robust way, leading to fewer artefacts in the warped (texture) frame. ARPS has been proposed as a way to reduce the complexity of the ME process in state-of-the-art predictive coding, but thanks to the adaptive nature of the pattern and the refinement step, it produces superior results compared with full search ME in the given setup. The final method that we propose, "DT2T", is an OF method that uses both texture and depth information. This method employs the symmetric (texture) data term proposed in [13], but also adds the asymmetric information given by the depth maps in the motion estimation. In addition, we consider the symmetric texture based SI generation method presented in [16], which produces state-of-the-art results, as an alternative that does not use depth maps. This method is denoted as "T2T".

2.1. Optical Flow based SI generation

As opposed to BB motion estimation, OF gives a dense result, calculated by means of a global regularization process. Typical SI generation methods are based on calculating motion using texture KFs [14, 16]. Here we extend the symmetric OF method of [13] to also include asymmetric depth information. A novelty of our approach is the introduction of a new OF-based SI generation system, in which two data terms are jointly minimized.

Given a set of pixel-domain (texture) key frames and depth frames T_{t-1}, T_{t+1}, D_t, and $D_{t'}$, $t' = t-1$ or $t' = t+1$,

we want to estimate the dense flow field v such that the following optical flow constraints

$$C_T(\boldsymbol{x}, v) \triangleq T_{t+1}(\boldsymbol{x} + v(\boldsymbol{x})) - T_{t-1}(\boldsymbol{x} - v(\boldsymbol{x})), \quad (1)$$

$$C_D(\boldsymbol{x}, v) \triangleq D_{t'}(\boldsymbol{x} + v(\boldsymbol{x})) - D_t(\boldsymbol{x}), \quad (2)$$

are minimized, where $\boldsymbol{x}$ denotes a 2D point in the image.

The OF constraints are not sufficient for the motion estimation, and in order to make the problem well-posed, one has to penalize irregular behaviour. Here we focus on the TV-L^1 energy [18], where data fidelity between two frames is measured by L^1-norms of the optical flow constraints, and the global regularization term penalizes the total variation E of the estimated motion:

$$E(v) = \int \lambda_1 \|C_T(\boldsymbol{x}, v)\| + \lambda_2 \|C_D(\boldsymbol{x}, v)\| + \|\mathscr{D}v(\boldsymbol{x})\| \, \mathrm{d}\boldsymbol{x}. \quad (3)$$

With two data terms, this energy cannot be minimized as proposed in [13], unless $\lambda_1 = 0$ (D2T) or $\lambda_2 = 0$ (T2T). However, an extension to a sum of two 1-norm data terms (including the cases $\lambda_1 = 0$ or $\lambda_2 = 0$) is presented in [19]. This solution is used to substitute the original data term solution in [13], giving an algorithm that minimizes (3). The first data term produces flows that are symmetric through the interpolated frame, while the other term allows non-symmetric motion vectors. This combination should produce motion vectors where smaller details are matched using depth information, while bigger details (including lighting changes and shadows, which are not visible from depth data) should be matched using the texture frames. With the given formulation (3) we consider three distinct cases:

T2T: $\lambda_1 = 40$, $\lambda_2 = 0$,
D2T: $\lambda_1 = 0$, $\lambda_2 = 30$,
DT2T: $\lambda_1 = 5$, $\lambda_2 = 40$.

It has to be noted that DT2T can only be calculated by using the new OF introduced here, while D2T and T2T could have been calculated by using the method presented in [13]. The final estimate of the motion v is recovered following the general implementation described by [13], with the following exceptions: 65 pyramid levels are used, and 90 warps with 10 inner iterations are performed on each level; the Gaussian smoothing of input images prior to downsampling has standard deviation 0.5 for T2T and DT2T, and 0.35 for D2T; after linear upsampling of the flows, they are filtered using a 3×3 median filter. OF naturally leads to the non pixel location problem, in which a target position in T_{t-1} and T_{t+1} does not have integer coordinates. In this case bicubic interpolation is used.

2.2. Side Information Fusion

As previously outlined, our system is based on [4], and therefore also uses the Overlapped Block Motion Compensation (OBMC) SI. We propose the use of a multi-hypothesis decoder as a way of fusing the produced SIs [14]. For each SI the probability distribution of the bits of the bitplanes is calculated using the SI and its estimated residual. The distributions are then combined together using fixed weighting coefficients. Six different coefficients are used, and the resulting distributions are fed into six LDPCA decoders. For each new received chunk of syndromes the decoding is tried; if one of the decoders converges, its result is taken as final result and its combined distribution is used to reconstruct the corresponding DCT coefficients. This process can be also seen as a decoder-based rate-optimization since the chosen solution is the one requiring less bits. We employed the 2 SIs decoder (denoted as "2SI") and a 3 SIs decoder (denoted as "3SI"). For the 2SI decoder the first SI is OBMC and the second is chosen between the ones presented here. For the 3SI decoder we use OBMC, DT2T and T2T as SIs.

Table 1: QPs used with the given quantization matrices Q_i.

Sequence \ $\mathbf{Q_i}$	Q_1	Q_4	Q_7	Q_8
Dancer	33	30	27	23
Ballet	38	31	24	19
Breakdancers	40	33	26	22

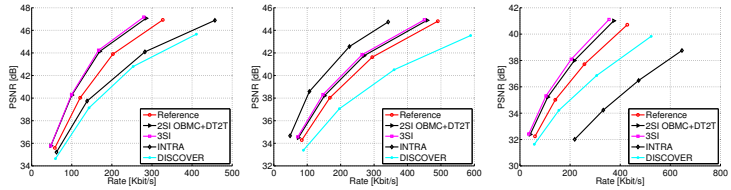
3. EXPERIMENTAL RESULTS

The system has been tested on a single view of the sequences "Breakdancers" and "Ballet" from Microsoft Research [12], and "Dancer" from Nokia Research [20]. We used the central view of the three sequences, at 15 fps downsampled to CIF resolution. The quantization matrices Q_i, $i = 1, 4, 7, 8$ of the DISCOVER project [21] are employed. The (texture) KFs are H.264/AVC Intra encoded using the QPs in Table 1. We have tested the first 100 frames of each sequence and reported the results for Group-Of-Pictures (GOP) 2, using as reference the decoder presented in [4] on texture frames. All the results and the graphs show only the WZ frames performance, since the KFs are encoded in the same manner for all the sequences. The rate of the depth frames is not taken into account, since we suppose that they are already required by the system and not only used to improve the coding performance.

The Bjøntegaard PSNR distances and bit-rate savings [22] for the 2SI decoder have been reported in Table 2, using uncompressed depth maps (denoted as $\mathrm{QP}_D = U$), and H.264/AVC Intra coded depth maps with quantization parameter $\mathrm{QP}_D = \{20, 40\}$. In Table 2, each 2SI decoder is denoted with the second employed SI, since the first one is always OBMC. DT2T is an extension of T2T, hence we report also the latter for the ease of comparison. From Table 2, in the case of uncompressed depth maps, we can see that DT2T is the best performing method for Dancer and Ballet, both medium motion sequences. For Ballet the second best method is D2T OF, while for Dancer the difference between D2T OF and T2T is negligible. It has to be noted that Ballet is a real-world sequence and depth maps have been estimated from texture data. Dancer, on the other hand, is a computer-

Table 2: Bjøntegaard Distances between the reference decoder [4] and the proposed decoders.

Sequence		T2T	$QP_D = U$			$QP_D = 20$			$QP_D = 40$		
			D2T BB	D2T OF	DT2T	D2T BB	D2T OF	DT2T	D2T BB	D2T OF	DT2T
Dancer	ΔRate[%]	15.44	11.12	17.57	23.82	9.48	17.16	24.55	5.41	12.57	19.29
	ΔPSNR[dB]	0.74	0.48	0.78	1.12	0.42	0.76	1.15	0.23	0.56	0.88
Ballet	ΔRate[%]	2.76	11.03	17.42	19.11	9.46	17.69	18.97	6.20	15.36	17.03
	ΔPSNR[dB]	0.19	0.74	1.19	1.32	0.63	1.22	1.32	0.41	1.05	1.17
Break-dancers	ΔRate[%]	3.84	8.50	12.43	11.61	7.71	12.30	11.83	5.03	11.15	10.95
	ΔPSNR[dB]	0.24	0.52	0.76	0.71	0.47	0.75	0.73	0.30	0.68	0.67



(a) Ballet (b) Breakdancers (c) Dancer

Fig. 4: RD curves, WZ frames only, uncompressed depth maps.

generated sequence in which depth maps have been generated using the actual distances of the 3D object models from the virtual camera. The depth maps of Dancer are smoother compared with those of Ballet, hence the SI of Dancer has lower quality compared with Ballet. Nevertheless, the novel DT2T approach outperforms both D2T OF and T2T, improving over the single SI decoder [4] by up to 1.32 dB. Breakdancers shows a much higher temporal activity making the motion estimation more difficult. In this case D2T OF greatly outperforms T2T. DT2T shows a negligible performance loss compared with D2T OF. In all the aforementioned cases D2T BB is not able to achieve the same performance as D2T OF due to the lack of flexibility of the block-based approach. The proposed OF-based methods show high resilience to the quantization noise of the depth maps: the performance in the case of $QP_D = 20$ are basically the same as in the uncompressed case; while in the case of $QP_D = 40$ we can notice a performance degradation, but the DT2T method is still able to achieve improvements ranging from 0.67 to 1.17 dB over [4]. It may also be noted that DT2T works correctly even when T2T has better performance compared with D2T OF (see Dancer, $QP_D = 40$) and it is still superior to the best of them. For what concerns the 3SI decoder (Table 3), it is able to correctly fuse the SIs leading to good and robust improvements, always superior to any 2SI decoder, with gains ranging from 0.90 dB to 1.49 dB. No case-specific optimization has been performed, i.e. the parameters used for the OF-based methods are fixed for all the sequences and for all the QP_D values. The RD-curves for the three sequences in the case of uncompressed texture frames are also depicted in Fig. 4,

Table 3: Bjøntegaard Distances between the reference decoder [4] and the 3SI decoder.

Sequence \ QP_D		U	20	40
Dancer	ΔRate[%]	30.69	30.80	28.40
	ΔPSNR[dB]	1.48	1.49	1.35
Ballet	ΔRate[%]	20.70	20.63	18.96
	ΔPSNR[dB]	1.45	1.45	1.32
Break-dancers	ΔRate[%]	15.15	15.05	14.49
	ΔPSNR[dB]	0.94	0.93	0.90

where the performance of [4] is denoted as "Reference". As it can be seen the decoder in [4] is able to greatly outperform the DISCOVER [6] decoder on textures in all the settings, making it more fair to compare the proposed systems with the one in [4].

4. CONCLUSION

In this work we investigated the possibility of using depth maps for improved SI generation in single-view video-plus-depth DVC. The proposed system is able to achieve good and robust improvements over one of the best single SI DVC decoders available in literature [4], with improvements ranging from 0.90 dB to 1.49 dB. OF-based methods showed clear superiority to conceptually similar block-based methods. The DT2T method was able to successfully combine the symmetrical OF approach [16] and the D2T approach introduced in this work. Finally, the multi-hypothesis decoder was able to successfully and robustly fuse the SIs here presented.

5. REFERENCES

[1] S. Mehrotra, Z. Zhang, Q. Cai, C. Zhang, and P.A. Chou, "Low-complexity, near-lossless coding of depth maps from Kinect-like depth cameras," in *Proc. of IEEE MMSP*, October 2011, pp. 1 –6.

[2] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. 19, no. 4, 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 Trans. Inform. Theory*, vol. 22, pp. 1–10, 1976.

[4] X. Huang and S. Forchhammer, "Cross-band noise model refinement for transform domain Wyner-Ziv video coding," *Signal Processing: Image Communication*, vol. 27, no. 1, pp. 16 – 30, 2012.

[5] B. Girod, A.M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," in *Proc. of the IEEE*, January 2005, vol. 93, pp. 71 –83.

[6] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: Architecture, techniques and evaluation," in *Proc. of PCS*, November 2007.

[7] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive codes for distributed source coding," *EURASIP Signal Processing Journal*, vol. 86, no. 11, pp. 3123–3130, November 2006.

[8] M. Winken, H. Schwarz, and T. Wiegand, "Motion vector inheritance for high efficiency 3D video plus depth coding," in *Proc. of PCS*, May 2012, pp. 53 –56.

[9] G. Petrazzuoli, M. Cagnazzo, F. Dufaux, and B. Pesquet-Popescu, "Wyner-Ziv coding for depth maps in multiview video-plus-depth," in *Proc. of IEEE ICIP*, September 2011, pp. 1817 –1820.

[10] M. Salmistraro, M. Zamarin, L. L. Rakêt, and S. Forchhammer, "Distributed multi-hypothesis coding of depth maps using texture motion information and optical flow," in *Proc. of IEEE ICASSP*, May 2013, *accepted*.

[11] X. Artigas, E. Angeli, and L. Torres, "Side information generation for multiview distributed video coding using a fusion approach," in *Proc. of NORSIG 2006*, June 2006, pp. 250 –253.

[12] L.C. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 600–608, 2004.

[13] L.L. Rakêt, L. Roholm, A. Bruhn, and J. Weickert, "Motion compensated frame interpolation with a symmetric optical flow constraint," in *Advances in Visual Computing*, George Bebis *et al.*, Ed., vol. 7431 of *Lecture Notes in Computer Science*, pp. 447–457. Springer Berlin Heidelberg, 2012.

[14] X. Huang, L.L. Rakêt, H.V. Luong, M. Nielsen, F. Lauze, and S. Forchhammer, "Multi-hypothesis transform domain Wyner-Ziv video coding including optical flow," in *Proc. of IEEE MMSP*, October 2011, pp. 1 –6.

[15] H.V. Luong, L.L. Raket, X. Huang, and S. Forchhammer, "Side information and noise learning for distributed video coding using optical flow and clustering," *IEEE Trans. Image Process.*, vol. 21, no. 12, pp. 4782 –4796, December 2012.

[16] L.L. Rakêt, J. Søgaard, M. Salmistraro, H. V. Luong, and S. Forchhammer, "Exploiting the error-correcting capabilities of low density parity check codes in distributed video coding using optical flow," in *Proc. of SPIE*, 2012, vol. 8499, pp. 84990N–84990N–15.

[17] Y. Nie and K.-K. Ma, "Adaptive rood pattern search for fast block-matching motion estimation," *IEEE Trans. Image Process.*, vol. 11, no. 12, pp. 1442 – 1449, December 2002.

[18] C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime TV-L^1 optical flow," in *Ann. Symp. German Association Patt. Recogn*, 2007, pp. 214–223.

[19] A. Wedel, T. Pock, J. Braun, U. Franke, and D. Cremers, "Duality TV-L^1 flow with fundamental matrix prior," in *Image and Vision Computing*, Auckland, New Zealand, November 2008, pp. 1–6.

[20] "Extension of existing 3DV test set toward synthetic 3D video content," ISO/IEC JTC1/SC29/WG11, Doc. M19221, Daegu, Korea, January 2011.

[21] "DISCOVER project test conditions," December 2007, http://www.img.lx.it.pt/~discover/test_conditions.html.

[22] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," in *VCEG Meeting*, Austin, USA, April 2001.

DISTRIBUTED MULTI-HYPOTHESIS CODING OF DEPTH MAPS USING TEXTURE MOTION INFORMATION AND OPTICAL FLOW

Matteo Salmistraro[◊] *Marco Zamarin*[◊] *Lars Lau Rakêt*[⋆] *Søren Forchhammer*[◊]

[◊]DTU Fotonik, Technical University of Denmark, Ørsteds Plads,
2800 Kgs. Lyngby, Denmark. Emails: {matsl, mzam, sofo}@fotonik.dtu.dk
[⋆]Department of Computer Science, University of Copenhagen, Universitetsparken 1,
2100 Copenhagen, Denmark. Email: larslau@diku.dk

ABSTRACT

Distributed Video Coding (DVC) is a video coding paradigm allowing a shift of complexity from the encoder to the decoder. Depth maps are images enabling the calculation of the distance of an object from the camera, which can be used in multiview coding in order to generate virtual views, but also in single view coding for motion detection or image segmentation. In this work, we address the problem of depth map video DVC encoding in a single-view scenario. We exploit the motion of the corresponding texture video which is highly correlated with the depth maps. In order to extract the motion information, a block-based and an optical flow-based methods are employed. Finally we fuse the proposed Side Informations using a multi-hypothesis DVC decoder, which allows us to exploit the strengths of all the proposed methods at the same time.

Index Terms— Distributed Source Coding, Depth Map Coding, Wyner-Ziv Coding, Optical Flow, Distributed Video Coding.

1. INTRODUCTION

In this work we address the coding of depth maps, using DVC [1, 2] as basis of our coding architecture.

Depth maps are particular images enabling the calculation of the distance of an object from the camera. A video representation format that is gaining popularity is the so-called "video-plus-depth", where in addition to texture data (the luminance and chrominance information of the scene), per-pixel depth information is also provided [3, 4]. Depth data allows fast generation of virtual views using the so-called Depth-Image-Based-Rendering (DIBR) algorithms [4], which makes the video-plus-depth format suitable for 3DTV and free viewpoint system implementations [5]. Moreover, it can be used for a number of purposes that can be of interest in modern video surveillance scenarios such as scene matting, activity detection and object tracking [3].

DVC is a video coding paradigm that allows shifting the complexity from the encoder side to the decoder side due to the fact that Motion Estimation (ME)—which heavily contributes to the computational complexity in state-of-the-art video codecs—can be performed at the decoder. Typical DVC scenarios feature strict power consumption constraints at the transmitter side, requiring low-complexity encoders, while the requirements are less stringent at the decoder. A multi-camera video surveillance scenario is a good example of a system with such requirements [2]. In a typical DVC architecture [1] inter-coded frames (i.e. frames coded by means of motion estimation and compensation) are substituted by the so-called Wyner-Ziv (WZ) frames. WZ frames are encoded in a different manner: parity check data are calculated and transmitted. An Intra-coded frame is referred to as Key Frame (KF) and is encoded and transmitted as in traditional video coding. At the decoder side KFs are used to estimate WZ frames by means of ME. The estimated frame, called Side Information (SI), can be corrected using parity bits from the encoder. The SI generation algorithm is therefore of crucial importance as the quality of the estimated frames directly affects the amount of additional parity bits required, and consequently the Rate-Distortion (RD) performance of the system. The core part of the proposed decoder is the Transform Domain WZ (TDWZ) codec [6]. At the encoder the WZ frame is DCT transformed and quantized. Each DCT coefficient is organized in bitplanes, and for each bitplane a LDPCA [7] encoder calculates the parity bits. At the decoder the SI is generated using an interpolation-based technique, for example Overlapped Block Motion Compensation (OBMC) [6]. A subset of the parity bits are sent to the decoder. The decoder tries to correct the errors present in the corresponding bitplane of the SI using the parity bits. If the decoding is not successful new bits are requested. Another key element of the decoder is the noise modelling, which is important in order to provide the LDPCA decoder with the likelihood of the value of each bit. The errors present in the SI are modelled as Laplacian distributed errors. In order to calculate the distribution, an estimation of the residual is needed. The residual is the difference between the SI and the original frame, which can not be directly calculated in practice. For more information on

978-1-4799-0356-6/13/$31.00 ©2013 IEEE

M. Salmistraro, M. Zamarin, L.L. Rakêt, S. Forchhammer, "Distributed Multi-Hypothesis Coding of Depth Maps using Texture Motion Information and Optical Flow", *Proc. of 2013 IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2013)*, pp. 1685 - 1689, Vancouver, Canada, May 26-31, 2013.

DVC coding the reader is referred to [1, 2, 6].

Since texture and depth represent different aspects of the same 3D scene, the two components show a high correlation [8]. In a video-plus-depth DVC scenario such correlation can be exploited to improve the overall coding efficiency e.g. by refining the depth SI generation using texture motion information.

In this paper Transform Domain WZ coding of depth maps is addressed in a mono-view video-plus-depth scenario. This scenario is interesting when addressed with DVC because two dependent streams can be independently encoded but dependently decoded. This approach can be generalized to a multi-camera scenario, where a depth camera and a texture camera are used together, making inter-camera cooperation difficult or not feasible. Texture data are supposed to be available at the decoder and are used to improve the WZ decoding of depth data. Three different SIs are generated and fused using a multi-hypothesis approach [9]. The first SI is generated by applying block-based texture motion vectors to the depth component; the second one is obtained by applying the texture optical flow to the depth component; finally the third one is generated by means of motion estimation from depth data only. The three SIs present different characteristics and provide accurate estimation of the to-be-decoded depth frame in different regions.

1.1. Related Works

The use of texture motion information for depth compression purposes has been explored in conventional predictive coding in [10] and more recently in [11]. The same concepts can be exploited in a DVC decoder for accurate SI generation, as done in [12] in which multiple decoded texture frames are used. In our work we suppose that the decoder has access to the corresponding texture frame of the to-be-decoded depth frame, while in [12] only the texture frames corresponding to the depth KFs are used. Moreover, we investigate optical-flow-based methods, while [12] investigate only block-based methods. The multi-hypothesis decoder employed is the same as in [9] where OBMC was used with block-based extrapolation and optical flow-based interpolation in order to improve a texture-based DVC decoder. We use the same approach in order to effectively fuse three different SIs.

A preliminary study of the aforementioned problem has been performed in [13] but only the block-based method was presented and no fusion technique was proposed.

Optical flow-based SI generation has already been used for example in [9]. In this case the flows were used to interpolate an unknown texture frame given the previous and successive texture frames. In our framework we use the flow to extract the motion information from texture frames.

The remainder of this paper is organized as follows: Section 2 describes the proposed SI generation algorithms and the relative SI fusion method. In Section 3 experimental results are discussed. Finally, Section 4 summarizes the presented work.

2. SI GENERATION AND FUSION

In this section we describe the two proposed SI generation algorithms for depth maps, exploiting texture motion information. We also analyse the employed fusion procedure. In addition a third SI based on OBMC [6] on depth video is included in the fusion procedure. This decoder is used as basis for evaluating the performance of the two texture-based SIs and the performance of the fusion procedure. It has to be noted however, that OBMC has not been devised for depth maps and it has not been modified in this work.

2.1. Texture-based SI generation algorithms

The main idea behind the proposed methods is that the motion of the texture is highly correlated with the one encountered in the depth data. For the to-be-decoded depth frame X at instant t, assume that the depth maps at instants $t-1$, and $t+1$ (D_{t-1} and D_{t+1}, respectively) are known. We can use the motion information of the texture to warp D_{t-1} and D_{t+1} towards X obtaining the SI Y. In order to perform the aforementioned procedure the texture frames at instants $t-1$, t, and $t+1$ (C_{t-1}, C_t, C_{t+1}, respectively) are available at the decoder. The Motion Vectors (MVs) are calculated from C_t to C_{t-1} and from C_t to C_{t+1}. The MVs are used in turn to motion compensate D_{t-1} and D_{t+1}, obtaining two depth SIs Y_1 and Y_2, respectively. The final SI, Y, is calculated as the arithmetic average of Y_1 and Y_2. The residual, R_Y, is calculated as the absolute difference between Y_1 and Y_2. The argument behind this simple choice is that if a region in X presents simple motion, it will be well predicted. Hence Y_1 and Y_2 will agree in the particular area, leading to low residual estimation. If on the contrary the two estimated frames disagree, the residual will be higher.

The methods used to calculate the motion from the texture data is of central importance to the SI quality. We have selected two different ME approaches.

2.2. Block-Based Side Information Generation

We consider the so-called "Adaptive Rood Pattern Search" (ARPS) ME algorithm proposed in [14]. This approach may not provide the lowest MSE (Mean-Squared-Error) between the motion compensated texture frame and the original one, however, it is able to capture the motion between the frames in a robust way, leading to fewer artefacts in the warped (depth) frame. ARPS has been proposed as a way to reduce the complexity of the ME process in state-of-the-art predictive coding, but thanks to the adaptive nature of the pattern and the refinement step, it produces superior results compared with

full search in the given setup. This Block-Based SI generation is referred as BB.

2.3. Optical Flow Side Information Generation

As an alternative to BB, we consider an Optical Flow (OF) [15] SI generation. As opposed to BB, the OF based ME is global, in the sense that individual motion vectors are estimated for every pixel. Given a set of texture frames C_t and $C_{t'}$, $(t' = t+1, t-1)$, in pixel domain, we want to estimate the dense flow field v such that the optical flow constraint

$$D(\boldsymbol{x}, v) \triangleq C_{t'}(\boldsymbol{x} + v(\boldsymbol{x})) - C_t(\boldsymbol{x}), \tag{1}$$

where $\boldsymbol{x}$ denotes a point in the image, is close to zero.

The optical flow constraint (1) will not be sufficient for motion estimation, and in order to make the problem well posed, one has to penalize irregular behavior. Here we focus on the TV-L^1 energy, where data fidelity between two frames is measured by the L^1-norm of the optical flow constraint, and the regularization term penalizes the total variation of the estimated motion:

$$E(v) = \int \lambda \|D(\boldsymbol{x}, v)\| + \|\mathscr{D}v(\boldsymbol{x})\| \, \mathrm{d}\boldsymbol{x}. \tag{2}$$

The total variation of a vector valued function is not uniquely defined, and several definitions have been used for this problem [16, 17, 18]. Here we use the definition of [19], since this method does not suffer from the channel smearing (i.e. independent optimization of the two channels of the motion vectors, the x- and y-components) of other definitions.

The final estimate of the motion v is recovered from iteratively minimizing a linearized version of (2) using the duality based splitting of [16]. The minimization is performed in a coarse to fine pyramid. We use 65 pyramid levels wit a scaling factor of 1.05, and Gaussian blurring of C_t and $C_{t'}$ with standard deviation 0.5, and on each level we perform 90 warps, with 1 outer and 10 inner iterations [16]. Furthermore we remove outliers by performing a median filtering of the flow for each warp. The parameter λ was set to 480. Compared to optical flow based interpolation [9] this value may seem high, however for the given test setup with higher temporal and spatial resolution, as well as the direct knowledge of the texture state, this higher weight on data fidelity is adequate. For more details on the implementation we refer to [18]. OF may lead to the non pixel location problem, in which a target position in D_{t-1} and D_{t+1} does not have integer coordinates. In this case bicubic interpolation is used.

2.4. Side Information Fusion

In order to exploit all the presented SIs (BB, OF, OBMC) a robust fusion technique is needed. In [9] it has been demonstrated that a multi-hypothesis decoder can be used to effectively combine block-based and pixel-based motion estimation techniques. In our work, we use the three SI decoders approach (referred as 3SI) as a way to fuse the three SIs. The multi-hypothesis decoder allows implementing a rate-based optimization strategy by using a number of parallel LDPCA decoders. Each LDPCA decoder is fed with a different weighted combination of the conditional probabilities for a given bitplane, and the syndromes coming from the encoder. Each bitplane contains the co-located bits of a given DCT coefficient. The decoded sequence of the first converging decoder is chosen as solution, and the corresponding weights used to combine the SIs are also used in the reconstruction process to improve the PSNR of the decoded frame. This method, thanks to the multi-decoder structure, shows robust gain, good performance and is therefore employed in this work as the fusion technique. However, the 3SI approach will increase the complexity of LDPC decoding up to 6 times.

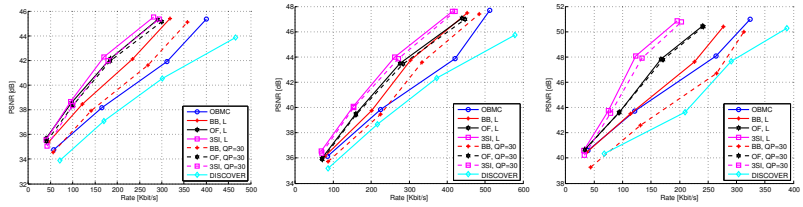
3. EXPERIMENTAL RESULTS

The system has been tested on the sequences "Breakdancers" and "Ballet" from Microsoft Research [20], and "Dancer" from Nokia Research [21]. We used the central view of the three sequences, at 15 fps downsampled to CIF resolution. The quantization matrices Qi = 1, 4, 7, 8 of the DISCOVER [22] project are employed. The KFs are H.264/AVC Intra encoded using QP = 40, 37, 31, 29 and are matched with the quantization matrices. We have tested the first 100 frames of each sequence and reported the results for Group-Of-Pictures (GOP) 2, 4, 8. The performance of the WZ frames has been evaluated and compared with the single SI OBMC decoder. In Tables 1-3 we list the Bjøntegaard differences [23] between the single SI OBMC decoder and the 3SI decoder. The results for lossless coded textures are listed as "QP = L", while the results using compressed textures, are listed with the QP used for compression. Texture compression has been performed with a standard H.264/AVC Intra coder[1]. In Figs. 1a-1c the RD curves for GOP2 are reported. We have also reported the performance of the single SI system and the performance of DISCOVER. Only the performance for WZ frames is reported. It has to be noted that the parameters of the 3SI decoder are the same for all the sequences and the quality level of the textures.

In Section 2, the SI generation for GOP2 has been outlined. In the cases of GOP4 and GOP8 a hierarchical coding structure [6] is used. First the SI for the central WZ frame is generated using C_{t-k}, C_{t+k}, D_{t-k}, and D_{t+k}, where k corresponds to half of the GOP size. The decoded WZ frame splits the GOP in two smaller GOPs in which the procedure can be iterated until all the WZ frames have been decoded.

From the results presented, it can be seen that the OF outperforms all the other single SI methods, showing also high robustness against texture quantization, while the BB

[1] JM 18.1 Reference Software, available at `iphome.hhi.de/suehring/tml`



(a) Ballet (b) Breakdancers (c) Dancer

Fig. 1: RD curves, WZ frames only, GOP2.

method suffers at lower qualities of the texture frames. The single SI OBMC-based decoder outperforms DISCOVER [24] codec in all the studied conditions and for all the investigated sequences. The 3SI is able to correctly fuse the three SIs, performing, on average, better or as well as the best available SI for the particular RD point. The improvements between the single SI OBMC decoder and the 3SI decoder ranges from 1.50 to 4.95 dB and from 21.24% to 49.06% bit-rate Bjøntegaard savings. Interestingly, the improvements for GOP8, are higher in the case of compressed textures for the Ballet and Breakdancers sequences (Table 3). A justification can be found in the non-linear low-pass filtering nature of the quantization, leading to more robust results, which in case of complex motion can be of benefit.

Sequence	**QP**	**ΔPSNR** [dB]	**ΔRate** [%]
Ballet	L	2.98	−46.46
	20	2.85	−44.99
	30	2.40	−39.79
Breakdancers	L	2.12	−34.02
	20	2.07	−33.28
	30	1.87	−31.16
Dancer	L	2.05	−42.90
	20	2.04	−40.41
	30	1.82	−36.24

Table 1: Bjøntegaard Distances between OBMC and the proposed methods, GOP2.

Sequence	**QP**	**ΔPSNR** [dB]	**ΔRate** [%]
Ballet	L	3.16	−44.71
	20	3.06	−43.32
	30	2.57	−38.11
Breakdancers	L	1.71	−23.87
	20	1.68	−23.52
	30	1.50	−21.24
Dancer	L	2.47	−42.54
	20	2.38	−42.03
	30	2.00	−38.81

Table 2: Bjøntegaard Distances between OBMC and the proposed methods, GOP4.

Sequence	**QP**	**ΔPSNR** [dB]	**ΔRate** [%]
Ballet	L	3.03	−42.80
	20	3.46	−46.62
	30	2.98	−41.53
Breakdancers	L	1.80	−23.95
	20	1.95	−25.55
	30	1.76	−23.37
Dancer	L	4.95	−49.06
	20	4.74	−47.61
	30	4.43	−45.04

Table 3: Bjøntegaard Distances between OBMC and the proposed methods, GOP8.

4. CONCLUSION

In this work we addressed the problem of DVC-based depth-map coding. We devised algorithms to produce higher quality SIs, employing the texture frames. We used two methods in order to extract the motion information from the texture frames: a block-based method and an optical flow-based one. The optical flow achieved better performance and superior robustness to quantization of the textures compared with the block-based system. The multi-hypothesis decoder proved to be an effective and robust way to fuse the three generated SIs outperforming the best single SI available. The improvements between the single SI OBMC decoder and the multi-hypothesis decoder ranges from 1.50 to 4.95 dB and from 21.24% to 49.06% Bjøntegaard bit-rate savings.

5. REFERENCES

[1] B. Girod, A.M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 71 –83, January 2005.

[2] F. Pereira, "Distributed video coding: basics, main solutions and trends," in *Proc. of IEEE ICME*, Piscataway, NJ, USA, 2009, ICME'09, pp. 1592–1595, IEEE Press.

[3] K. Müller, P. Merkle, and T. Wiegand, "3-D video representation using depth maps," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 643 –656, April 2011.

[4] P. Kauff, N. Atzpadin, C. Fehn, M. Müler, O. Schreer, A. Smolic, and R. Tanger, "Depth map creation and image-based rendering for advanced 3DTV services providing interoperability and scalability," *Signal Processing: Image Communication, Special issue on three-dimensional video and television*, vol. 22, no. 2, pp. 217 – 234, 2007.

[5] M. Tanimoto, M.P. Tehrani, T. Fujii, and T. Yendo, "Free-Viewpoint TV," *Signal Processing Magazine, IEEE*, vol. 28, no. 1, pp. 67 –76, January 2011.

[6] X. Huang and S. Forchhammer, "Cross-band noise model refinement for transform domain Wyner-Ziv video coding," *Signal Processing: Image Communication*, vol. 27, no. 1, pp. 16 – 30, 2012.

[7] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive codes for distributed source coding," *EURASIP Signal Processing Journal*, vol. 86, no. 11, pp. 3123–3130, November 2006.

[8] I. Daribo, C. Tillier, and B. Pesquet-Popescu, "Motion vector sharing and bitrate allocation for 3D video-plus-depth coding," *EURASIP J. Appl. Signal Process.*, vol. 2009, pp. 1–13, January 2008.

[9] X. Huang, L.L. Rakêt, H.V. Luong, M. Nielsen, F. Lauze, and S. Forchhammer, "Multi-hypothesis transform domain Wyner-Ziv video coding including optical flow," in *Proc. of IEEE MMSP*, October 2011, pp. 1 –6.

[10] H. Oh and Y.-S. Ho, "H.264-based depth map sequence coding using motion information of corresponding texture video," in *Proc. of PSIVT*, Berlin, Heidelberg, 2006, pp. 898–907, Springer-Verlag.

[11] M. Winken, H. Schwarz, and T. Wiegand, "Motion vector inheritance for high efficiency 3D video plus depth coding," in *Proc. of IEEE PCS*, May 2012, pp. 53 –56.

[12] G. Petrazzuoli, M. Cagnazzo, F. Dufaux, and B. Pesquet-Popescu, "Wyner-Ziv coding for depth maps in multiview video-plus-depth," in *Proc. of IEEE ICIP*, September 2011, pp. 1817 –1820.

[13] M. Salmistraro, M. Zamarin, and S. Forchhammer, "Wyner-Ziv Coding of Depth Maps Exploiting Color Motion Information," *Proceedings of SPIE, the International Society for Optical Engineering*, vol. 8666, pp. 8666–14, 2013.

[14] Y. Nie and K.-K. Ma, "Adaptive rood pattern search for fast block-matching motion estimation," *Image Processing, IEEE Transactions on*, vol. 11, no. 12, pp. 1442 – 1449, December 2002.

[15] R. Szeliski, *Computer Vision: Algorithms and Applications*, Springer-Verlag New York, Inc., New York, NY, USA, 1st edition, 2010.

[16] C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime TV-L^1 optical flow," in *In Ann. Symp. German Association Patt. Recogn*, 2007, pp. 214–223.

[17] L.L. Rakêt, L. Roholm, M. Nielsen, and F. Lauze, "TV-L^1 optical flow for vector valued images," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*, Yuri Boykov *et al.*, Ed., vol. 6819 of *Lecture Notes in Computer Science*, pp. 329–343. Springer, 2011.

[18] L.L. Rakêt, L. Roholm, A. Bruhn, and J. Weickert, "Motion compensated frame interpolation with a symmetric optical flow constraint," in *Advances in Visual Computing*, George Bebis *et al.*, Ed., vol. 7431 of *Lecture Notes in Computer Science*, pp. 447–457. Springer Berlin Heidelberg, 2012.

[19] B. Goldluecke, E. Strekalovskiy, and D. Cremers, "The natural vectorial total variation which arises from geometric measure theory," *SIAM Journal on Imaging Sciences*, vol. 5, no. 2, pp. 537–563, 2012.

[20] L.C. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 600–608, 2004.

[21] "Extension of existing 3DV test set toward synthetic 3D video content," ISO/IEC JTC1/SC29/WG11, Doc. M19221, Daegu, Korea, January 2011.

[22] "DISCOVER project test conditions," December 2007, http://www.img.lx.it.pt/ discover/test_conditions.html.

[23] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," in *VCEG Meeting*, Austin, USA, April 2001.

[24] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: Architecture, techniques and evaluation," *Proc. of IEEE PCS*, November 2007.

2012 Data Compression Conference

Rate-adaptive BCH coding for Slepian-Wolf coding of highly correlated sources

Søren Forchhammer, Matteo Salmistraro, Knud J. Larsen,
Xin Huang and Huynh Van Luong
DTU Fotonik, Technical University of Denmark
Ørsteds Plads, Building 343, 2800 Kgs. Lyngby, Denmark
Email:{sofo,matsl,knjl,xhua,hulu}@fotonik.dtu.dk

Abstract

This paper considers using BCH codes for distributed source coding using feedback. The focus is on coding using short block lengths for a binary source, X, having a high correlation between each symbol to be coded and a side information, Y, such that the marginal probability of each symbol, X_i in X, given Y is highly skewed. In the analysis, noiseless feedback and noiseless communication are assumed. A rate-adaptive BCH code is presented and applied to distributed source coding. Simulation results for a fixed error probability show that rate-adaptive BCH achieves better performance than LDPCA (Low-Density Parity-Check Accumulate) codes for high correlation between source symbols and the side information.

1 Introduction

In this paper we address the use of BCH codes in distributed source coding (DSC) with feedback. In recent years distributed source coding [1] has gained increasing interest e.g. for distributed video coding [2]. The coding is referred to as Slepian-Wolf coding and based on the Slepian-Wolf theorem [3]. The relation between Slepian-Wolf (SW) coding and syndrome decoding of error-correcting codes was observed by Wyner in [4]. Applying and designing practical SW coding schemes of finite block length poses challenges. Turbo and LDPC codes were considered in [5] using block lengths of 10^4 and 10^5 bits, which may be too long for some practical applications. Our aim is to use feedback for (SW) coding using shorter blocks. We shall consider the case where each symbol, X_i, to be coded is strongly correlated with the side information Y. Thus the conditional entropy $H(X|Y)$ is low. We shall analyze the case, where the difference between X_i and the corresponding symbol in the side information, Y_i, is modeled as a Bernoulli process having (a small) error probability, p. Distributed coding of strongly correlated sources was treated in [6], but the bit error rate was not reduced much, in the results reported, compared to what would be obtained simply by selecting the most likely values of X_i given the corresponding side-information, Y_i. They considered distributed arithmetic coding as an alternative to Turbo and LDPC coding. Without a feedback channel, using these codes as well as BCH codes for DSC [2], [7], fixed rate coding is used, leading to an inferior performance since the code used has to be designed for the relatively large variation in number of errors in the short blocks. To alleviate this we shall consider a system with feedback as in [1] where LDPC(A) coding is used, but here we shall use BCH coding in a rate-adaptive manner. Syndromes for the BCH code are requested one by one through the feedback channel and the requests are stopped when a sufficiently reliable decoded results is reached. To increase the reliability, a check of the decoded result may be requested

1068-0314/12 $26.00 © 2012 IEEE
DOI 10.1109/DCC.2012.31

S. Forchhammer, M. Salmistraro, K. J. Larsen, X. Huang, H. V. Luong, "Rate-adaptive BCH Coding for Slepian-Wolf Coding of Highly Correlated Sources", *2012 Data Compression Conference (DCC 2012)*, pp. 237 - 246, Snowbird, UT, USA, Apr. 10-12, 2012.

and performed based on additional syndromes of the BCH code or CRC checking. The problem is addressed both with or without knowledge of the error probability, p. In Section 2 rate-adaptive BCH coding and distributed decoding is presented, as well as the use of extra syndromes for checking. Section 3 presents expressions and an algorithm for analyzing the performance. Section 4 deals with the code selection in the case of unknown error probability. To reduce the cost of these check bits, Section 5 extends the scheme to one or more CRC checks over multiple BCH blocks. Simulation results are presented in Section 6.

2 Slepian-Wolf coding using rate-adaptive BCH

For use in distributed source coding we consider a specific version of Slepian-Wolf coding [3] and present a rate-adaptive BCH code for DSC over a communication system with error free transmission including an error free feedback channel. A block, $\boldsymbol{X}$, of length l bits from the source sequence is encoded using a parity check matrix for a linear code, and it is decoded with help of side information, $\boldsymbol{Y}$, correlated with $\boldsymbol{X}$. Let $\boldsymbol{E}$ denote the (modulo 2) difference between $\boldsymbol{X}$ and $\boldsymbol{Y}$. We proceed using the formulation in [4]. Syndromes, $\boldsymbol{s}_X(s)$, are calculated as shown below at the encoder and received at the decoder without errors:

$$Encoding: \boldsymbol{s}_X(s) = H(s)\boldsymbol{X}^t, \tag{1}$$

$$Side\ information: \ \boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{E} \tag{2}$$

$$Decoding: \boldsymbol{s}(s) = \boldsymbol{s}_X(s) + H(s)\,\boldsymbol{Y}^t = H(s)\boldsymbol{E}^t, \tag{3}$$

where the calculations are done modulo 2 and the parity check matrix is denoted *H(s)* with index s to indicate an adaptive rate given by using an increasing number of rows in *H(s)*, and thus providing incremental redundancy. The syndromes are requested by the decoder from the source encoder through a feedback channel. The decoder [4] (ideally) performs a maximum-likelihood decoding of $\boldsymbol{s}$(s) to find the estimate $\hat{\boldsymbol{E}}$ which is used to find an estimate of $\boldsymbol{X}$ as $\hat{\boldsymbol{X}} = \boldsymbol{Y} + \hat{\boldsymbol{E}}$. Since the code is of finite length, errors in $\hat{\boldsymbol{E}}$ cannot be completely avoided. We consider the case that the difference $\boldsymbol{E}$ is an i.i.d. Bernoulli process with error probability, p, given the side information $\boldsymbol{Y}$. We analyze performance assuming that X and Y are equiprobable i.i.d. In this case the likelihood may be expressed by the Hamming distance. We consider a particular code and decoding algorithm below, and later analyze the probability of error.

2.1 Rate-adaptive BCH codes

We shall here consider the use of BCH codes as the linear codes [8] and describe how to use them in a rate-adaptive way. The length of the blocks $\boldsymbol{X}$ and $\boldsymbol{Y}$ is assumed to be $l \leq 2^M - 1$ for an integer M. Let α be a primitive element in $GF(2^M)$. For fixed s and given set of syndromes, $\boldsymbol{s}_X(s)$, the BCH code is sure to correct t errors if $\alpha, \alpha^2, ..., \alpha^{2t}$ are roots of the codewords regarded as binary polynomials. Syndromes for the source block *X(z)* are calculated in $GF(2^M)$ as $X(\alpha^i) = r_i(\alpha^i)$ where $r_i(z)$ is the remainder of $X(z)$ divided in $GF(2)$ by $m_i(z)$, the minimal polynomial for α^i. For binary BCH codes some syndromes are easily calculated from others, so there is only a need to know the independent syndrome values. For the decoding, a maximum-likelihood decoder should be used [4]. In [7] list decoding of BCH and similar codes is suggested (without considering feedback) to extend the capabilities of the usual bounded distance decoder, but the list decoding algorithms work best for low rate codes, i.e. for high $H(X|Y)$.

The structure of BCH codes makes them suited for rate adaptation as the number of syndromes is freely selectable. We introduce rate adaptation of BCH codes for DSC driven by the feedback channel to improve on bounded-distance decoding. For

decoding we use the Berlekamp-Massey algorithm [8] since it operates on the syndromes for increasing powers of α just as the rate adaption. The algorithm may be stopped at any step and the result evaluated, i.e. does the error locator provide an acceptable error pattern? We shall discuss this evaluation further below. If a new independent syndrome is needed, it is requested from the encoder, and the algorithm may continue from the stopping point since all previous syndromes are already included in the current result. A similar approach for adapting BCH codes was used in a more complex way by Shiozaki for an adaptive ARQ system [9]. As above, we let the index s indicate the number of independent syndromes known at a certain step in the rate-adaptation. We denote the number of bits in the remainder giving independent syndrome number s as $m(s)$ (usually $m(s) = M$) and we denote the number of errors correctable by having the s independent syndromes as $t(s)$.

2.2 Reliability of decoding and strategy for checking

A particular decoding of s independent syndromes may have two undesired results: *Decoder failure* or *Decoder error*. In case of failure, the rate-adaptive scheme just continues by requesting a new independent syndrome, but for a decoder error, the error pattern is wrongly accepted. For Reed-Solomon codes, the probability that a received vector with more than t errors is erroneously decoded is known [10] to be close to $1/t!$. A similar argument may be used for the probability of decoding error for a BCH code. Thus, if $t(s)$ is reasonably high, there is no need to test the reliability further, but for smaller $t(s)$, a test for the reliability has to be added. A separate CRC check may be used for this purpose, but here we shall instead use one or more subsequent independent syndrome(s). This is done simply by requesting syndrome(s) and letting the Berlekamp-Massey algorithm continue one or more steps where the discrepancies should evaluate as zero, i.e. the result also fits the extra check syndrome(s). We denote the number of bits received for checking as $c(s)$. The reliability is improved by (about) $2^{-c(s)}$ when using a CRC check [11]. If the check fails, a new error pattern may be calculated based on the syndromes already available, including the checks, and if needed, extra check syndromes may be requested. Since the decoding with few syndromes are rather unreliable, we start with s_{min} syndromes and thereafter one syndrome at a time. We may also impose a maximum number of syndromes s_{max}, e.g. if we want to limit the number of requests/syndromes for practical or numerical reasons. Finally we may at some point decide the strength of BCH decoding to be sufficient (i.e. the probability of erroneous decoding is sufficiently low) and therefore drop the CRC check, i.e. $c(s) = 0$ after this point. If we were to use a conventional CRC, once we had checked for a given s and rejected the decoding, we could not later back off to a check with fewer bits. Generally we require $c(s)$ extra syndrome bits after decoding using s syndromes.

To summarize a rate-adaptive BCH DSC codec is specified by the length, l, of the BCH code, the decoding algorithm applied at each s and the check strategy, $c(s)$. The codec requires an error free feedback channel.

3 Performance - known error probability

We shall analyze the performance for i.i.d. equiprobable binary sequences X and Y, with fixed probability of error, $p(x_i \neq y_i) = p$. Let $P_e(t)$ denote the (binomially distributed) probability of t errors among the l positions. For a strategy, $c(s)$, we may calculate the expected number of errors when stopping at s syndromes as follows. All errors (apart from errors remaining at s_{max}) are due to erroneous decoding. The other difference compared to analysis of fixed rate and t is that there may have been an erroneous decoding at an earlier stage. We assume that this instance is independent

of the number of errors and denote the residual probability of not being able to have a correct decoding after s syndromes by $P_R(s)$. This is given by $P_e(> t(s))$ but decreased by the (small) probability of erroneous decoding prior to $t(s)$ which is not caught by the CRC. Due to the independence assumption, the relative probabilities of decoding, failure and erroneous decoding at $t(s)$, remain the same as before. We shall now express approximative bounds on the expected number of errors, $B(s)$, after erroneous decoding with s syndromes. First we express contributions $r(s)$ and $b(s)$ to the bit-rate, $R(s)$, and $B(s)$, respectively. To simplify the expressions we introduce the expected number of errors beyond $t(s)$ given that there are more errors:

$$\overline{e}(s) = \frac{\sum_{e=t(s)+1}^{l} eP_e(e)}{P_e(> t(s))} = \frac{pl - \sum_{e=0}^{t(s)} eP_e(e)}{P_e(> t(s))} \tag{4}$$

and we introduce $P_E(s)$ as the probability of an erroneous decoding with s syndromes and using the arguments from [10] we get a heuristic bound

$$P_E(s) \leq \frac{\sum_{e=0}^{t(s)} \binom{l}{e}}{2^{N(s)}} P_e(\geq t(s)+1), \tag{5}$$

where $N(s)$ is the total number of bits in the s independent syndromes. First, we initialize lower and upper bounds and an approximate value for $B(s)$ at $s_{min} \geq 0$ after adding a $c(s_{min})$ CRC:

$$P_E(s_{min})2^{-c(s_{min})}\overline{e}(s_{min}) \leq B(s_{min}) \leq P_E(s_{min})2^{-c(s_{min})}(\overline{e}(s_{min}) + t(s_{min})) \tag{6}$$

$$B(s_{min}) \approx P_E(s_{min})2^{-c(s_{min})} max\{2t(s_{min}) + 1; \overline{e}(s_{min})(1 - 2\frac{t(s_{min})}{l}) + t(s_{min})\} \tag{7}$$

Equation (6) is derived using the probability of a decoding error with s_{min} syndromes, $P_E(s_{min})$, which is reduced due to the checking with $c(s_{min})$ check bits. In this case there are at least $\overline{e}(s_{min})$ errors and the decoding may add up to $t(s_{min})$ errors more. When the $t(s_{min})$ is larger than the mean number of errors in the block, pl, $\overline{e}(s_{min})$ is around $t(s_{min})+1$, and the decoding usually makes $t(s_{min})$ errors more, so the first element in the maximization in (7) gives an approximate value. The other element in the maximization is for $t(s_{min}) < pl$ where some of the corrections may accidently hit the $\overline{e}(s_{min})$ positions in error. For ease of calculation, we assume that checking using $c(s)$ bits by extra syndromes has the same effect as a CRC ($2^{-c(s)}$), and that $P_E(s)$ may also be used in the following steps. Now let $P_R(s_{min})$ denote the probability of more errors without stopping at s_{min}.

$$P_R(s_{min}) = P_e(> t(s_{min})) - P_E(s_{min})2^{-c(s_{min})} \tag{8}$$

$$R(s_{min}) = \sum_{s=1}^{s_{min}} m(s) + (1 - P_R(s_{min}))c(s_{min}) \tag{9}$$

Equation (9) uses $P_R(s_{min})$ in (8) to give the starting point for the rate expression in (13) below. Using the same arguments as for s_{min}, we get the contributions for each syndrome, s, following s_{min} as:

$$\frac{P_R(s-1)}{P_e(> t(s-1))} P_E(s)2^{-c(s)}\overline{e}(s) \leq b(s) \leq \frac{P_R(s-1)}{P_e(> t(s-1))} P_E(s)2^{-c(s)}(\overline{e}(s) + t(s)) \tag{10}$$

$$b(s) \approx \frac{P_R(s-1)P_E(s)2^{-c(s)}}{P_e(> t(s-1))} max\{2t(s) + 1; \overline{e}(s)(1 - 2\frac{t(s)}{l}) + t(s)\} \tag{11}$$

$$P_R(s) = P_R(s-1)(1 - \frac{P_e(t(s-1) < e \leq t(s)) + P_E(s)2^{-c(s)}}{P_e(> t(s-1))}) \tag{12}$$

$$r(s) = P_R(s-1)m(s) + (P_R(s-1) - P_R(s))c(s) \tag{13}$$

Equations (10-13) are applied for the syndromes $s_{min} < s \leq s_{max}$ and all contributions are summed to evaluate $B(s_{max})$ and $R(s_{max})$. $P_R(s_{max})$ is now treated as failure which gives an extra contribution, denoted $b_\Delta(s_{max})$, to (10)-(11):

$$b_\Delta(s_{max}) = P_R(s_{max})\overline{e}(s_{max}) \tag{14}$$

Finally, summing all of these contributions gives:

$$B(s_{max}) = B(s_{min}) + \sum_{s=s_{min}+1}^{s_{max}} b(s) + P_R(s_{max})\overline{e}(s_{max}) \tag{15}$$

The expression in (11) is used for $b(s)$. Summing $r(s)$ gives

$$R(s_{max}) = R(s_{min}) + \sum_{s=s_{min}+1}^{s_{max}} r(s) \tag{16}$$

Thus for a given length, l, of the BCH code and checking strategy, $c(s)$, $s_{min} \leq s \leq s_{max}$, (15-16) express the expected rate and bit error rate performance. Calculating the performance for a number of strategies, $c(s)$, the convex hull of the bit-rate vs. bit-error-rate may be used to define an optimal strategy.

4 Code selection for unknown error probability

In applications the error probability may be unknown. Here we consider adaptive solutions for the case that p is not known. For efficient bit rate, $c(s)$ should be high for low values of s and may be set to 0 if s is big enough. We represent $c(s)$ by a set of thresholds $\{T_i\}$, which decides the required strength of check as a function of how many syndromes were used to decode. In our system if $s < T_2$, we request $\Delta s = 3$ extra syndromes, if $T_2 \leq s < T_1$, $\Delta s = 2$, if $T_1 \leq s < T_0$, $\Delta s = 1$ and otherwise $\Delta s = 0$. Using the model previously introduced it is possible to find the optimal strategy (i.e. the set of values $\{T_i\}$) through exhaustive search. This section addresses the problem of finding the optimal strategy based on estimates of p, in case of unknown error probability. We shall assume that the error probability is distributed following a beta distribution ($p \in B(\delta_e, \delta_c)$). In this case, the optimal estimator ($p_e(k)$) is a modification of [12] used for (universal) adaptive source coding and here used for estimating the error probability given the number of errors up to and including the k-th block is [13]:

$$p_e(k) = \frac{n_e(k) + \delta_e}{k \cdot l + \delta_e + \delta_c} \tag{17}$$

$$n_e(k) = \sum_{i=1}^{k} e(i) \tag{18}$$

where $n_e(k)$ is the number of errors from the first to the k-th block and $e(i)$ is the number of errors in the i-th block. Unfortunately, we can not observe the errors directly, but only count the difference between the side information sequence and decoded sequence, which may be incorrectly decoded. For the i-th block this value is denoted $\hat{e}(i)$ and the total number of estimated errors after k blocks is $\hat{n}'_e(k)$.

$$\hat{p}'_e(k) = \frac{\hat{n}'_e(k) + \delta_e}{k \cdot l + \delta_e + \delta_c} \tag{19}$$

$$\hat{n}'_e(k) = \sum_{i=1}^{k} \hat{e}(i) \tag{20}$$

Using a bounded distance (or maximum-likelihood) decoding, a decoding error will mean that the actual number of errors is larger than the estimated, $e(i) \geq \hat{e}(i)$ and hence (19-20) will underestimate the actual error probability and not provide a

central estimator. In order to counteract this bias, we modify (19-20) by adding a correction factor $d(i)$ to $\hat{e}(i)$ obtaining

$$\hat{p}_e''(k) = \frac{\hat{n}_e''(k) + \delta_e}{k \cdot l + \delta_e + \delta_c} \tag{21}$$

$$\hat{n}_e''(k) = \sum_{i=1}^{k} (\hat{e}(i) + d(i)) \tag{22}$$

where $d(i)$ is the expected value of the non-negative term $\Delta(i) = e(i) - \hat{e}(i)$,

$$d(i) = \mathrm{E}[\Delta(i)|t(s_i)] = \sum_{j=0}^{(l-t(s_i))} j p_\Delta(\Delta = j|t(s_i)) \tag{23}$$

We know that $t(s_i - 1) < \hat{e}(i)$, since if not we would have decoded at $s_i - 1$, which would not have been rejected by the check, as this is a subset of that at s_i. In most cases for low values of p in our simulations the main contributions are for s_i for which $\hat{e}(i) = t(s_i)$. We sum these contributions and disregard the rest using

$$p_\Delta(\Delta = j|t(s_i)) = \begin{cases} \frac{P_e(t(s_i))}{P_e(t(s_i)) + P_E(s_i) 2^{-c(s_i)}}, & \text{If } j = 0 \\ \frac{P_E(s_i, t(s_i)+j) 2^{-c(s_i)}}{P_e(t(s_i)) + P_E(s_i) 2^{-c(s_i)}}, & \text{If } j > 0 \end{cases} \tag{24}$$

where $P_E(s_i)$ is given by (5), P_e is as in Sec. 2 and

$$P_E(s, i) = \frac{\sum_{e=0}^{t(s)} \binom{l}{e}}{2^{N(s)}} P_e(i) \tag{25}$$

In (21) the contributions to $\hat{p}_e''(k)$ from all blocks have equal weight. A weighting factor can assign more importance to the more reliable contributions,

$$\hat{p}_e'''(k) = \frac{\hat{n}_e'''(k) + \delta_e}{\sum_{i=1}^{k} a(i) l + \delta_e + \delta_c} \tag{26}$$

$$\hat{n}_e'''(k) = \sum_{i=1}^{k} a(i)(\hat{e}(i) + d(i)) \tag{27}$$

$$a(i) = p_\Delta(\Delta = 0|t(s_i)) \tag{28}$$

The formulas (21-28) each requires the knowledge of p, which they shall estimate. To overcome this problem, we take a simple approach: at the Step k we first calculate $\hat{p}_e'(k)$ (19), then use this for calculating $\hat{p}_e''(k)$ (21), which is used for calculating $\hat{p}_e'''(k)$ (21), which in turn is used to calculate the optimal strategy for step $k + 1$.

5 Hierarchical CRC

Using extra syndromes to check, whether a decoding is correct, costs a non-neglectable amount of bits for a BCH code of short block length and low $H(X|Y)$. In this case, the cost of correcting an error is most likely a remainder of M bits. So while long blocks will reduce check overhead they also increase the cost of correcting each error. If possible, a CRC check may be applied across a number of blocks, forming a longer block.

For f blocks, a CRC check of c' bits is performed across the blocks. We refer to this as a c' CRC check and also assume that these c' bits are received without errors. The check is performed when the decoder has accepted a decoding for each of the f blocks. If the CRC check is not satisfied, an extra syndrome is requested for each of the individual blocks in turn, one at a time, until a block decodes to a different sequence than before. Thereafter the c' CRC check is performed again. This continues until the c' CRC check is satisfied. This may be generalized to a hierarchical CRC check system. Upon acceptance at one level, i, a new CRC check is applied at

the next level, $i+1$, across a number f_{i+1} of blocks at the current level, i. Starting with level 0, $f_0 = f$. If a CRC check at some level i detects an error, more syndromes are requested for the blocks until a new information word is retrieved. It is assumed that the CRC is independent of the so far received BCH syndromes.

5.1 Performance estimation

The effect of the hierarchical CRC check is evaluated for a first level CRC. The c' bits of the CRC check are distributed among f blocks, increasing the code length by c'/f for each. The performance of the scheme may readily be simulated on f codes in parallel. For faster and simpler simulation, the decoding of one block according to the given check rules specified by $c(s)$ is extended with an analysis of the expected contribution to bit error rate and increased rate due to potential additional syndromes requested for the f blocks. For low block error rates and modest values of f, when a decoding error is detected by the c' CRC, with high probability, only one of the blocks was wrongly decoded. This gives a conservative estimate. The additional contributions to rate and improvements of bit error rate are expressed below.

In a given simulation, let e denote the number of errors in the side information and let $e(s)$ denote the number of errors in the decoded sequence in case of a decoding error. Let s_0 denote the number of syndromes after which the simulation stops the first time, having a block that has been decoded and accepted. The contribution to the rate is $R(s_0)+c(s_0)$. If the decoding is correct, the contribution to bit error rate is 0. Now consider the case that the decoded block was incorrect. The simulation continues until it has sufficient syndromes to correct the e errors, i.e. $t(s) = e$. Along the way it will calculate the expected contributions to the rate and error probability. The c' CRC check will not detect this error with probability $2^{-c'}$, leading a contribution to the bit-error rate,

$$b(s_0) = 2^{-c'} e(s_0). \tag{29}$$

The residual probability is $P_R(s_0) = 1-2^{-c'}$. Trying to decode with the syndromes specified by $c(s)$ will not change the decoding. A new syndrome is requested. Let s_1 denote the number of syndromes received in total after this extra syndrome and further let s_i denote the total number of syndromes if i extra syndromes are received.

If syndrome s_1, does not lead to a stop, $P_R(s_1) = P_R(s_0)$ and an extra syndrome has been requested by the $f-1$ other blocks. Adding the $m(s_0)$ bits of the next syndrome gives a bound on the additional rate of

$$r(s_1) \leq P_R(s_0)((f-1)M + m(s_1)) \tag{30}$$

If s_1 leads to a correct decoding, the expected contribution is bounded by $\leq P_R(s_0)(f-1)M/2$ when averaging over the order in which the blocks receive syndromes. Adding the BCH syndrome $m(s_1)$ and the check bits, $c(s_1)$ gives

$$E[r(s_1)] \leq P_R(s_0)((f-1)M/2 + m(s_1) + c(s_1)) \tag{31}$$

where $E[]$ denotes expectation. If s_1 leads to an incorrect decoding, also after a check using the $c(s_1)$ syndromes bits, the contribution to the error rate in case the c' bit CRC does not detect this error is

$$b(s_1) = P_R(s_0) 2^{-c'} e(s_1) \tag{32}$$

and the contribution to the rate is

$$E[r(s_1)] \leq P_R(s_0) 2^{-c'} ((f-1)M/2 + m(s_1) + c(s_1)) \tag{33}$$

but there is a higher probability that the error is detected and that a new syndrome is requested leading to $P_R(s_1) = P_R(s_0)(1-2^{-c'})$, and the contribution

$$r(s_1) = m(s_1) \tag{34}$$

in this case. Thereafter a new syndrome is requested until decoding. The contributions are given by replacing, s_1 with s_i and s_0 with s_{i-1} above. For a given simulation with an accepted decoding error at s_0, the expected bit error rate is

$$E[B] = \sum_{s=s_0}^{t(s)=e} b(s) \tag{35}$$

where $b(s)$ are given by (29) or (32) and 0 otherwise. The overall bit error rate is obtained by averaging over all simulations. Likewise the corresponding expected rate for a simulation with an accepted error at s_0 is

$$E[R] = R(s_0) + c(s_0) + c'/f + \sum_{s=s_1}^{t(s)=e} r(s) \tag{36}$$

where $r(s)$ are given by (30), (31), (33) or (34). The overall expected rate is obtained by averaging over all simulations.

We may derive an optimum value of f given the block error rate, P_E. The saving by sharing check bits are $c'(f-1)/f$ and the extra cost given by extra bits for the other blocks is upper bounded by $a(f-1)P_E M$, where a is the expected number times each correct block will request a syndrome. Selecting $c' = M$, i.e. replacing a syndrome check of M bits (for all values of s) with a shared check of $c' = M$, we may calculate the difference and take the derivative with respect to f. This leads to a optimum value of $f = \sqrt{(aP_E)^{-1}}$.

6 Results

Simulations were performed using a fixed i.i.d. error probability, $p(x_i \neq y_i) = p(x_i \neq y_i|y) = p$ for the side information, y and equiprobable i.i.d. X and Y, but no errors on the syndromes, $\mathbf{s}_X(s)$, used for SW decoding. Thus we have a BSC (Binary Symmetric Channel) relation symbol by symbol between the information bits, x_i and the side information bits, y_i.

6.1 Results for known conditional probability, p

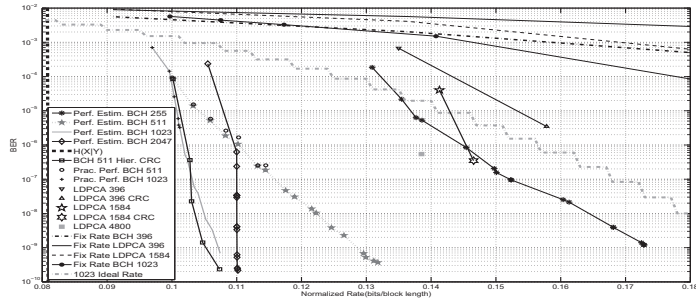


Figure 1: Performance analysis for $p = 0.01$.

The performance estimation of rate-adaptive BCH codes described in Section 3 and the extension with hierarchical CRC introduced in Section 5 are evaluated in this Section. Since we are aiming at strongly (inter-)correlated sources, we focus on values of $H(X|Y)$ below 0.3 and initially evaluate performance at a low bit error probability, $p = 0.01$, i.e. $H(X|Y) = 0.0808$. The performance of BCH codes with different block lengths, $l \in \{255, 511, 1023, 2047\}$, is evaluated and shown in Fig. 1

as a function of the bit-rate normalized with respect to l. The performance values depicted belong to the convex hull (using linear scales) of the results obtained with different check strategy, $c(s)$. The lower bound on bit-rate is given by the Slepian-Wolf bound, $H(X|Y)$. BCH of length 511 with hierarchical CRC is also evaluated in Fig. 1, for $c' \in \{8, 12, 16\}$ and number of blocks, $f \in \{2, 4, 6, 8\}$. The performance of rate-adaptive BCH with block lengths 511 and 1023 is also simulated. The simulated BCH use the $c(s)$ strategy obtained by the convex hull of the performance estimations. The simulated results (BCH 511 and 1023) match the estimated results very well. We also compare with simulation results for a popular rate-adaptive code, LDPCA code [1] for block lengths of 396, 1584 and 4800. An 8-bit CRC is optionally utilized after LDPCA decoding. As shown in Fig. 1, rate-adaptive BCH achieves a better performance than LDPCA in this test. Changing the block length shows that BCH with longer block length gives a steeper decrease of error rate. However, the basic performance of BCH 2047 is slightly worse. The BCH with block length 1023 gives the best result. First level hierarchical CRC, improves the performance of the rate-adaptive BCH 511. The effective blocks length may be considered to be $f \times 511$ in this case.

For further comparison, the performance for fixed rate BCH codes with block lengths 396 and 1023 are given, as well as the performance of fixed rate LDPCA with block lengths 396 and 1584. The achievable performance with a fixed-rate error-correcting code of length 1023 (Theorem 33 in [14] for block error rate was modified to give a bound for the bit error rate) and also applicable for our distributed coding (1-3), is also included. We see that the rate-adaptive BCH clearly provides better results than even what can be hoped for using a fixed rate code for short block lengths.

In Fig. 2, the performance of various BCH codes are compared with an LDPCA code having a code length of 1584. For each probability, p, examined, we have chosen the strategy having the lowest rate out of the set giving a BER not exceeding that of the LDPCA evaluated at p. Up to $H(X|Y) = 0.28$, the rate-adaptive BCH can provide better performance than the (longer) LDPCA code. Adding an 8-bit CRC to the LDPCA code gave similar relative results. In this case rate-adaptive BCH was better up to $H(X|Y) = 0.25$.

6.2 Results for unknown conditional probability p

In this section we present the results obtained using the adaptive method designed for unknown error probability, p, on the BSC. We measured the difference ($\Delta Rate$, ΔBER) between the performance of the optimal strategy for given p and the performance achieved by our algorithm estimating the probability and use this to select the strategy. The results were obtained using 10^5 blocks, a block length of 511 and a target BER=10^{-6}. In Table 1, we used a windowed approach: for the first 100 blocks we use a strategy designed for $p = 0.01$ and then we start estimating p (26-28) and based on this updating the strategy every 20 blocks. In the second part of Table 1, the performance for the same simulation is reported when statistics is collected only after the first 5000 blocks. In Table 1, the results are a weighted average performance assuming that the prior distribution of p is Beta distributed. As the results show our method is able to reach a reasonable precision compared to the optimal strategy. Using the correction and the weighting factor allow us to achieve better convergence to the target BER compared with estimating p by (19-20) where, for example, $\Delta BER = -4.04 \cdot 10^{-6}$ and $\Delta Rate = 4.39 \cdot 10^{-5}$ and with (21-22) where, for example, $\Delta BER = -1.63 \cdot 10^{-6}$ and $\Delta Rate = 4.43 \cdot 10^{-5}$. In both the cited cases $\delta_e = 1$, $\delta_c = 99$.

The estimation of p given by (26-28) was observed to provide fast convergence. As an example, a test with $p = 0.02$, was performed (without the initial window), calculating a new estimate after each block and reconsider the strategy after every 10 blocks. After 18 blocks $p - \hat{p}_e'''(k)$ was smaller than $p_e(k) - \hat{p}_e'''(k)$ when calculating $p_e(k)$ with the actual number of errors, and beyond 20 blocks both errors were below 10^{-3}.

Table 1: Adapting strategy compared with the optimal strategy

		All blocks		Last 95000 blocks	
Prior	**Rate**	**ΔRate**	**ΔBER**	**ΔRate**	**ΔBER**
$\delta_e = 0.1$, $\delta_c = 9.9$	0.09	$8.45 \cdot 10^{-6}$	$-1.68 \cdot 10^{-6}$	$8.54 \cdot 10^{-6}$	$-1.58 \cdot 10^{-6}$
$\delta_e = 1$, $\delta_c = 99$	0.11	$5.58 \cdot 10^{-6}$	$-5.77 \cdot 10^{-7}$	$1 \cdot 10^{-6}$	$-1.77 \cdot 10^{-7}$
$\delta_e = 10$, $\delta_c = 990$	0.11	$-1.36 \cdot 10^{-6}$	$-1.18 \cdot 10^{-8}$	$2.99 \cdot 10^{-9}$	$-5.87 \cdot 10^{-10}$

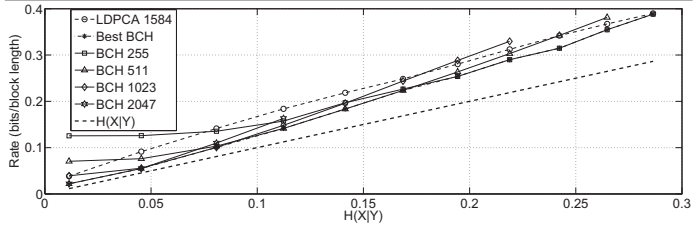


Figure 2: BCH performance compared with LDPCA

References

[1] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive distributed source coding using low-density parity-check codes," *EURASIP Signal Process. Journal, Special Section Distributed Source Coding*, vol. 86, pp. 3123–3130, Nov. 2006.

[2] C. Yeo, K. Ramchandran, "Robust Distributed Multiview Video Compression for Wireless Camera Networks," *IEEE Trans. Image Proc.,*, vol. 19, April 2010.

[3] D. Slepian and J.K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. 19, pp. 471–480, July 1973.

[4] A.D. Wyner, "Recent results in the Shannon theory," *IEEE Trans. Inform. Theory*, vol. 20, pp. 2–10, Jan. 1974.

[5] V. Stanković, A. D.Liveris, Z. Xiong, and C. N. Georghiades, "On code design for the Slepian-Wolf problem and lossless multiterminal networks,"*IEEE Trans. Inform. Theory*, vol. 52, no. 4, pp. 1495–1507, April 2006.

[6] M. Grangetto, E. Magli, and G. Olmo, "Distributed arithmetic coding for the Slepian-Wolf problem,"*IEEE Trans. Sign. Proc.*, vol. 57, no. 6, pp. 2245–2257, June 2009.

[7] M. Ali and M. Kuijper, "Source coding with side information using list decoding," *in Proc. IEEE Int. Symp. Inform. Theory*, Austin, Texas, USA, pp. 91–95, June 2010.

[8] R.E. Blahut, *Algebraic Codes for Data Transmission*. Cambridge University Press, UK, 2003.

[9] K.A. Shiozaki, "Adaptive type-II hybrid ARQ system using BCH codes," *Trans. IEICE*, vol. E75-A, pp. 1071–1075, Sept. 1992.

[10] R. McEliece and L. Swanson, "On the error probability for Reed-Solomon codes," *IEEE Trans. Inform. Theory*, vol. IT-32, pp. 702–703, Sept. 1986.

[11] T. Kløve and V.I. Korzhik, *Error Detecting Codes*. Kluwer Academic, Boston, MA. 1995.

[12] R. E. Krichevsky and V. K. Trofimov, "The Performance of Universal Encoding," *IEEE Trans. Inform. Theory*, vol.27, no.2, pp.199-207, March 1981.

[13] J. Justesen and S. Forchhammer, *Two-Dimensional Information Theory and Coding - With Application to Graphics and High-Density Storage Media*. Cambridge University Press, UK, 2010.

[14] Y. Polyanskiy, H.V. Poor, S. Verdu, "Channel Coding Rate in the Finite Blocklength Regime," *IEEE Trans. Information Theory*, vol.56, no.5, pp.2307-2359, May 2010.

Salmistraro *et al. EURASIP Journal on Advances in Signal Processing* 2013, **2013**:166
http://asp.eurasipjournals.com/content/2013/1/166

EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

Rate-adaptive BCH codes for distributed source coding

Matteo Salmistraro*, Knud J Larsen and Søren Forchhammer

Abstract

This paper considers Bose-Chaudhuri-Hocquenghem (BCH) codes for distributed source coding. A feedback channel is employed to adapt the rate of the code during the decoding process. The focus is on codes with short block lengths for independently coding a binary source X and decoding it given its correlated side information Y. The proposed codes have been analyzed in a high-correlation scenario, where the marginal probability of each symbol, X_i in X, given Y is highly skewed (unbalanced). Rate-adaptive BCH codes are presented and applied to distributed source coding. Adaptive and fixed checking strategies for improving the reliability of the decoded result are analyzed, and methods for estimating the performance are proposed. In the analysis, noiseless feedback and noiseless communication are assumed. Simulation results show that rate-adaptive BCH codes achieve better performance than low-density parity-check accumulate (LDPCA) codes in the cases studied.

Keywords: Distributed source coding; Rate-adaptive error-correcting codes; Rate-adaptive BCH codes; BCH codes

1 Introduction

In this paper, we address the use of Bose-Chaudhuri-Hocquenghem (BCH) codes in distributed source coding (DSC) with feedback. In recent years, DSC [1,2] has gained increasing interest, e.g. for distributed video coding (DVC) [3-6]. The coding is referred to as Slepian-Wolf (SW) coding and is based on the SW theorem [1]. The relation between SW coding and syndrome decoding of error-correcting codes was observed by Wyner in [7].

Applying and designing practical SW coding schemes of finite block length pose challenges. Turbo and low-density parity-check (LDPC) codes have been applied and studied, e.g. in [8] using block lengths of 10^4 and 10^5 bits, but this may be too long for some practical applications.

We shall consider SW coding of shorter blocks within an architecture, where the decoder can provide feedback to the encoder. Therefore, we shall compare the proposed codes with low-density parity-check accumulate (LDPCA) codes [9]. We shall focus on the case where each symbol to be coded, X_i, is strongly correlated with the side information Y, and thus the conditional entropy $H(X|Y)$ is low. We shall analyze the case where the difference between X_i, and the corresponding symbol in the side information, Y_i, is modelled as a Bernoulli process having (a small) error probability, $p = P(X_i \neq Y_i)$.

In DSC, using short block length may be of interest, e.g. in the case of delay restrictions relative to the bit rate or for adaptive coding. Context-based adaptive coding as used, e.g. in conventional image and video coding may, in principle, adapt after every symbol. Using short code lengths in DSC, it is possible to obtain decoded bits at a fine granularity, allowing, in turn, the parameters used to model the source to adapt and/or converge faster, when performing adaptive DSC. In transform domain DVC [6], bit planes of DCT coefficients are coded: for QCIF resolution this means 1584 source bits in each bit plane to encode, and it may be desired to adapt with even finer granularity in an adaptive DSC architecture. Distributed coding of strongly correlated sources was treated in [10], where arithmetic codes were used in place of LDPC or Turbo codes. However, in the reported results, the bit error rate was not reduced much when compared with simply selecting the most likely values of X_i given the corresponding side information, Y.

In the case of feedback-free DSC coding [3,4,11], the code has to be designed to cope with the relatively large variation in number of errors in case of short blocks. As

*Correspondence: matsl@fotonik.dtu.dk
DTU Fotonik, Technical University of Denmark, Ørsteds Plads, 2800 Kgs. Lyngby, Denmark

M. Salmistraro, K. J. Larsen, S. Forchhammer; "Rate-adaptive BCH Codes for Distributed Source Coding", *EURASIP Journal on Advances in Signal Processing*, Vol. 2013, 166, 2013.

Salmistraro *et al. EURASIP Journal on Advances in Signal Processing* 2013, **2013**:166
http://asp.eurasipjournals.com/content/2013/1/166

we also demonstrate in the result section, the performance of a feedback-free non-rate-adaptive code is limited by the block length and for short- and medium-length codes clearly inferior compared with the performance of its rate-adaptive counterpart. In the context of non-rate-adaptive codes, quasi-arithmetic codes for DSC have been investigated in [12], and in the field of real-number codes, BCH-DFT codes [13] have been proposed.

In general, a rate-adaptive code is an error-correcting code having the capability to vary its strength (i.e. increase or decrease the number of parity symbols) in order to adapt to the number of errors in the given block. In our case, rate adaptation is performed incrementally and controlled by the decoder by means of a feedback channel. This is a quite common assumption in LDPCA and Turbo code-based DVC [5,6]. BCH codes with feedback channel have also been used in [14] in order to perform the quantum key reconciliation step in the quantum key distribution protocol. In this system, the rate of the BCH code is fixed and it is decided based on the noise on the quantum channel. The feedback is used to allow the receiver to inform the sender whether the quantum key has been correctly reconciled, and therefore, it does not need to be discarded. Nevertheless, the feedback channel is not used for rate adaptation purposes.

We shall consider a system with feedback as in [9] where LDPCA coding is used, but here we shall use BCH coding in a rate-adaptive manner (RA BCH). Syndromes (blocks of syndrome bits) are requested one by one through the feedback channel, and the requests are stopped when a sufficiently reliable decoded result is reached, see Figure 1. To increase the reliability, a check of the decoded result may be requested and performed based on additional syndromes of the RA BCH code or cyclic redundancy checking (CRC). The main motivations of the study on RA BCH codes is the relatively low efficiency of LDPCA (and Turbo) codes when using a short packet length in a high-correlation scenario and the fact that the analysis of the performance of the RA BCH codes is simpler. An initial study on RA BCH codes was presented in [15], where we proposed a model for RA BCH codes: we demonstrated that BCH codes were able to outperform LDPCA codes in the high-correlation scenario, and we validated the correctness of the results of our model. Nevertheless, the model we proposed was based on some rough approximations; in particular, the checking process of the results was only crudely modelled. Secondly, we did not provide a complete theoretical model for the hierarchical check procedure, and we did not analyze other possible checking procedures for RA BCH codes. In this work, we provide a review of the basic concepts presented in [15], and we present new models based on a detailed analysis of BCH performance and report new results in order to better analyze, demonstrate, and evaluate the features of our system.

In this work, we will demonstrate that our RA BCH codes are able to outperform LDPCA codes if $p < 0.04$ ($H(X|Y) < 0.24$). When using efficient side information generation methods in DVC, e.g. OBMC or optical flow-based systems [16], the most significant bit planes of the coded coefficients have error rates comparable with those in our scenario when low-motion sequences are analyzed. For example, for the *Hall Monitor*, sequence is coded using side information produced by optical flow, and the maximum error rate among the first three bit planes of the first five DCT coefficients is $p = 0.038$. In some of the cases, it is less than 0.01. In [16] it has been noted that in low-motion sequences there is a consistent gap between the performance of an ideal code and the real performance of the LDPCA code. Our system could be considered as part of a Wyner-Ziv decoder in order to reduce the gap for the easy-to-decode (most significant) bit planes in low-motion sequences. It should be noted that we do not think that our codes can substitute LDPCA (or Turbo) codes generally in DVC, but we think that a hybrid system, using both BCH and LDPCA codes, chosen accordingly to the correlation of the bit planes, can improve the performance of current DSC architectures. For example, in [17] the authors presented a DVC codec able to perform rate decision at the decoder, achieving superior performance through the use of different coding modalities: skip, arithmetic, and intra-coding. We think that our codes could be used in a similar way. The proposed scheme can also be used for other DSC scenarios,

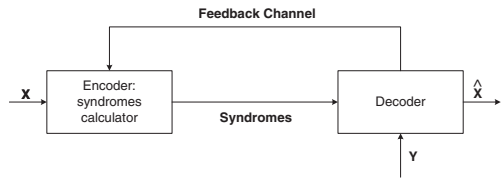


Figure 1 The addressed scenario with transmission and feedback channels.

e.g. in cases such as the one examined in [14], where LDPC codes require very long block lengths and the correlation is so high that the RA BCH codes are an efficient solution for many of the considered cases.

The rest of the paper is organized as follows: In Section 2 RA BCH coding and distributed decoding is presented. Section 3 presents expressions for analyzing the performance. In order to increase the reliability without heavily affecting the rate, Section 4 extends the scheme by an extra CRC check over multiple blocks decoded using RA BCH. Simulation results are presented in Section 5.

2 Rate-adaptive BCH codes

We consider a specific version of SW coding [1] employing RA BCH codes over a communication system with error free transmission and error-free feedback channel. We shall describe SW coding using linear block codes in the next subsection and thereafter describe the adaptive algorithm using BCH codes which belong to the class of linear codes.

2.1 Slepian-Wolf coding with linear block codes

A block, X, of length l bits from the source sequence is encoded using the parity check matrix of a linear code, and it is decoded with the help of side information, Y, which is correlated with X. Let E denote the difference between X and Y where all the calculations are done bitwise modulo 2. This is expressed by

$$Y = X + E. \tag{1}$$

We proceed using the formulation in [7]. A syndrome, $s_X(s)$, is calculated at the encoder using the parity check matrix $H(s)$, where s is an index to indicate the adaptive code rate to be introduced later:

$$s_X(s) = H(s)X. \tag{2}$$

As in [9] and commonly in DVC literature, $s_X(s)$ is assumed to be received without errors. Based on this syndrome and the side information Y, we first calculate

$$s_R(s) = s_X(s) + H(s)Y = H(s)E. \tag{3}$$

The decoder [7] (ideally) performs a maximum-likelihood decoding of $s_R(s)$ to find the estimate $\hat{E}$ which is used to find an estimate of X, denoted as $\hat{X}$

$$\hat{X} = Y + \hat{E}. \tag{4}$$

Since the code is of finite length, errors in $\hat{E}$ cannot be completely avoided. We consider the case that the difference E is an independent and identically distributed (i.i.d.) Bernoulli process with error probability p, given the side information Y. We analyze the performance assuming that X (and Y) are equiprobable i.i.d. In this case, the likelihood may be expressed by the Hamming distance, and thus, the decoding performance of the instance above follows from the usual performance analysis for linear block codes [18,19].

2.2 Rate adaptation

The block coding scheme described above may be made rate adaptive using an increasing number of rows in $H(s)$, thus providing incremental redundancy. This requires that the error-correcting code is chosen from a family of codes where the H matrix is extensible with more rows while the previous rows are kept, i.e. where more syndrome bits may be produced, without changing the previous ones. In this way, when more syndrome bits are requested by the decoder, the already received bits can be reused in the next decoding attempt, see Figure 1.

BCH codes [18] form such a family and allow simple algebraic decoding. We shall describe how to use them in a rate-adaptive way. The length of the blocks X and Y is $l \leq 2^M - 1$ for an integer M. Let α be a primitive element in $GF(2^M)$. For fixed s and a given syndrome, $s_X(s)$, the BCH code is sure to correct $t(s)$ errors if $\alpha, \alpha^2, \ldots, \alpha^{2t(s)}$ are roots of the codewords regarded as binary polynomials. The syndrome (2) is calculated in blocks of bits, which we hereafter name *syndromes*, and each of these is calculated in $GF(2^M)$ as $X(\alpha^i) = r_i(\alpha^i)$, where $r_i(z)$ is the remainder of $X(z)$ divided in $GF(2)$ by $m_i(z)$, the minimal polynomial for α^i. For binary BCH codes, for some values of i, syndromes do not increase the error-correcting capability; therefore, there is only a need to know the other syndromes which we call *independent* syndromes. We denote the number of bits in $r_i(\alpha^i)$ as $m(s) \leq M$. The structure of BCH codes makes them suited for rate adaptation as the number of syndromes is freely selectable up to the maximum number of syndromes.

We shall, as usual, decode the BCH codes using bounded distance decoding, i.e. correct up to $t(s)$ errors, whereas a maximum-likelihood decoder would have been more powerful. Rate adaptation through the use of the feedback channel allows to counterbalance the coding loss due to having a short block length. For decoding, we use the Berlekamp-Massey algorithm [18] to determine the error locator polynomial since it operates on the syndromes for increasing powers of α just as the rate adaptation. The next step in the decoding determines the errors from the roots in the error locator polynomial, and this result may be evaluated to assess its acceptability. Thus, the rate adaptation algorithm may be stopped if the result is acceptable. If a new independent syndrome is needed (i.e. the result is not acceptable), it is requested from the encoder, and the Berlekamp-Massey algorithm may continue from the stopping point since all previous syndromes are already included in the current result. A similar approach for adapting BCH codes was used to

do error correction in an adaptive ARQ communication system by Shiozaki [20]. This approach also assumes error-free syndromes, but the calculations involved are unnecessarily complex and no analysis of the (adaptive) BCH is provided.

The index s indicates the number of independent syndromes known at a certain step in the rate adaptation. We will also refer to s also as the *state* of the system.

2.3 Checking strategies

The bounded distance decoding of s independent syndromes may have two different results if the actual number of errors is higher than $t(s)$. If the actual error pattern E does not come closer than $t(s)$ to any codeword, the decoder declares a *decoder failure* and the rate-adaptive scheme just continues by requesting a new independent syndrome. If the error pattern is at most $t(s)$ from another non-zero codeword, we have a *decoder error* and the error pattern is wrongly accepted. For Reed-Solomon codes, the probability that a received vector with more than $t(s)$ errors is erroneously decoded is known to be close to $1/t(s)!$ [21]. A similar argument may be used for the probability of decoding error for a BCH code. Thus, if $t(s)$ is reasonably high, there is no need for further testing of the reliability, but for smaller $t(s)$, a test for reliability has to be added. We suppose that the BCH decoder has initially accepted a given decoded word as correct, employing s syndromes: now we add a procedure to check it. We shall detail the three coding strategies we analyze.

The most common way of addressing the problem is to use additional CRC check bits [11] after the BCH decoder has accepted a decoded word. If the CRC check fails, the BCH decoder is forced to start again. We refer to this approach as *Fixed CRC check*. This is also used in, e.g. DVC codecs [6]. It has to be noted that in common communication systems a CRC check is usually employed to check the correctness of a decoded block, and in case of failure, the block is requested again by the receiver if this is possible. In the rate-adaptive case, the CRC check is used to allow the decoder to improve the reliability of the decoded result through the request of other syndromes when the CRC check rejects the decoded string.

Since the reliability of the decoded sequence varies greatly with respect to the state s in which the decoder is, we can employ the knowledge of state s of the decoder (known at the encoder by means of the feedback) to use extra bits to perform the checking, i.e. we perform a strong check (requiring more bits) if $t(s)$ is low but a weaker check, or even no check, (requiring fewer or no bits) if $t(s)$ is high enough. We can perform the check using more syndromes or CRC bits.

In this scenario, we can request a CRC check, which in strength is matched with the desired resulting reliability and the reliability of the result at the time of the request. We denote the number of extra bits required to check the result as $c(s)$ which is a nonincreasing function ($c(s) \geq c(s+1)$). The reliability is improved by (about) $2^{-c(s)}$ when using a CRC check [22]. In case of a decoder error at s', the $c(s')$ bits used for the checking are stored and used for the next result the BCH decoder accepts. Hence, in general, the number of bits used to check a result when the system is in state $s \geq s'$ is greater than or equal to $c(s)$. We call this approach *Variable CRC check* (see Appendix).

When performing the check through the request of extra syndromes, we can simply request $\delta(s)$ extra syndrome(s) (whose transmission requires $c(s)$ bits) and let the Berlekamp-Massey algorithm continue one or more steps: if the result is not consistent with the extra check syndrome(s), the RA BCH decoder is forced to start the decoding process again. If the check fails, a new error pattern may be calculated based on the syndromes already available, including the checks, and if needed, extra check syndromes may be requested. We call this third solution *Syndrome check* method.

To summarize, we investigate and analyze three different checking strategies to be performed after a word has been accepted by the BCH decoder:

- Fixed CRC check: request a fixed amount of CRC check bits to check the result (analysis in Section 3.1)
- Variable CRC check: request a variable amount of CRC check bits to check the result; the strength of the CRC check is matched with the reliability of the decoded result (analysis is in the Appendix)
- Syndrome check: request a variable amount of syndrome bits to check the result; the number of syndromes is related to the reliability of the result (analysis is in Section 3.2)

It is quite straightforward to notice that in the two latter cases, the algorithm deciding the value of $\delta(s)$ or $c(s)$ dictates the performance of the code. In this paper, the parameters of the algorithm are specified by a set of thresholds $\mathbf{T} = \{T_0, T_1, T_2, T_3, T_{\max}\}$ where $T_0 < T_1 < \ldots < T_{\max}$. The decision on the strength of the check is based on comparing the state s with the thresholds. We will refer to $\mathbf{T}$ to as the *strategy*.

Since decoding with few syndromes is rather unreliable, we start with T_0 syndromes and thereafter one syndrome at a time is requested. We may also impose a maximum number of syndromes $T_{\max}$, e.g. if we want to limit the number of requests/syndromes for practical reasons. A general comment is that, using conventional CRC, once we have checked for a given s and rejected the decoding, we cannot back off later to a check with fewer bits. Using the Syndrome check approach, in case the check at s' rejects the decoded result, the syndromes at s' are reused for decoding and new (usually fewer) extra syndromes are

required for the later check, allowing the system to back off in practice. The function we employ to calculate $\delta(s)$ is

$$\delta_{\mathbf{T}}(s) = \begin{cases} 3 \text{ if } T_0 \leq s \leq T_1, \\ 2 \text{ if } T_1 < s \leq T_2, \\ 1 \text{ if } T_2 < s \leq T_3, \\ 0 \text{ if } T_3 < s \leq T_{\max}. \end{cases} \tag{5}$$

As can be seen, in the results to be presented, the maximum number of extra syndromes $\delta_M = 3$. In order to have a more compact notation, we make the dependence on $\mathbf{T}$ implicit, denoting $\delta_{\mathbf{T}}(s)$ simply as $\delta(s)$. In the case of variable CRC, we define $c(s) = M\delta(s)$, while in the case of Syndrome check, in general, $c(s) \leq M\delta(s)$. To summarize, a RA BCH DSC codec is specified by the length, $l = 2^M - 1$, of the RA BCH code and the strategy, $\mathbf{T}$, used to decide the values of $c(s)$.

3 A model for the performance of rate-adaptive BCH codes

Developing a model to predict the performance of a code, with parameters l and $\mathbf{T}$ given p, is important not only for performance analysis but also for optimizing the strategy of the code with respect to p. We can devise a simple estimate of the code length by noticing that whereas $m(s) \leq M$ in general, in most cases $m(s) = M$. Secondly, we can also notice that when having few errors, $t(s+1) = t(s)+1$; therefore, for correcting one more error, we need up to M more bits, which implies that in order to correct N_e errors (i.e. N_e ones in E), we need approximately MN_e bits, which leads to concluding that, on average, plM bits are needed to correct errors (plus an overhead for checking) because $E[N_e] = pl$. Below, we shall present more accurate estimations. We introduce a compact notation for the probability of having more than $k \in \mathbb{N}$ errors: $P_e(> k) \triangleq P(N_e > k)$. To simplify the expressions, in the next part of this section, we introduce $\overline{e}(s)$ as the expected number of errors beyond $t(s)$ given that there are more than $t(s)$ errors:

$$\overline{e}(s) = \frac{\sum_{e=t(s)+1}^{l} eP(N_e = e)}{P_e(> t(s))} = \frac{pl - \sum_{e=0}^{t(s)} eP(N_e = e)}{P_e(> t(s))}, \tag{6}$$

we introduce $P_E(s)$ as the probability of an erroneous decoding with s syndromes, and using an argument from [21] developed for fixed-rate codes, we get a heuristic bound, which we use as an estimate

$$P_E(s) \approx \frac{\sum_{e=0}^{t(s)} \binom{l}{e}}{2^{N(s)}} P_e(> t(s)), \tag{7}$$

where $N(s)$ $(= \sum_{s'=1}^{s} m(s'))$ is the total number of bits in the s independent syndromes.

The probability of having more than $t(s)$ errors and thereby not having the correct result after bounded distance decoding is expressed by $P_e(> t(s))$. The argument presented in [21], which we use to express (7) is based on the assumption that, when there are more than $t(s)$ errors, the error pattern, E, is completely random. Therefore, we may apply a combinatorial analysis. The ratio in (7) relates the possible decoder errors, i.e. cases with up to $t(s)$ Hamming distance to a wrong codeword of the BCH code to the total number of possible syndromes. There are $N_{\hat{E}}$ distinct error patterns, which may be output as accepted by the code. Actually one of these is the correct pattern, E. To reflect this, the expression should be multiplied by the ratio of the number of possible patterns which are decoder errors $(N_{\hat{E}} - 1)$ and divided by $N_{\hat{E}}$, but we assume $(N_{\hat{E}} - 1)/N_{\hat{E}} \approx 1$.

We approach the analysis of the RA BCH decoding process by defining two probabilities $P_B(s)$ and $P_A(s)$. Let $P_A(s)$ denote the probability of not ending the decoding (not accepting a previously decoded result) given that s syndromes are employed, and let $P_B(s)$ denote the probability of requesting s syndromes. For each state s between T_0 and $T_{\max}$, we calculate these two probabilities and use them to calculate the estimated expected bit error rate (BER) contribution $b(s)$ and the estimated expected rate contribution $r(s)$ related to state s. Finally, the estimated total BER can be calculated as

$$B = \sum_{s=T_0}^{T_{\max}} b(s), \tag{8}$$

and the estimated total rate can be calculated as

$$R = \sum_{s=T_0}^{T_{\max}} r(s). \tag{9}$$

We are going to present models to analyze the three check methods we previously introduced. Based on $m(s)$, we define $m_T(s)$ as the contribution to the rate given that we passed from state $s-1$ to s:

$$m_T(s) = \begin{cases} m(s) & \text{if } s > T_0, \\ \sum_{k=1}^{T_0} m(k) & \text{if } s = T_0. \end{cases} \tag{10}$$

Finally, we need to estimate the expected number of errors given that we are accepting a result:

$$e_B(s) = \max\{2t(s) + 1, \overline{e}_M(s)\}, \tag{11}$$

where $\overline{e}_M(s)$ is the estimation of the number of errors in the (wrongly) decoded word if $t(s) < pl$. In this case, it is possible that the wrongly decoded error pattern corrects some bits which are in error. This number can be approximated by $\overline{e}(s)t(s)/l$. Hence, the number of original errors is decreased to $\overline{e}(s) - \overline{e}(s)t(s)/l$, but $t(s) - \overline{e}(s)t(s)/l$ new

Salmistraro *et al. EURASIP Journal on Advances in Signal Processing* 2013, **2013**:166
http://asp.eurasipjournals.com/content/2013/1/166

errors are introduced. Summing the two contributions, we obtain the estimate

$$\overline{e}_M(s) = \overline{e}(s)\left(1 - \frac{2t(s)}{l}\right) + t(s). \quad (12)$$

In the case of $t(s) > pl$, we use $2t(s) + 1$ in the estimate (11) since it is the minimum distance between two valid words of the code, one of which is the correct error pattern and the other is the wrongly decoded error pattern. Summarizing, given a code with block length $l = 2^M - 1$ and a strategy $\mathbf{T}$ and for a given p, we want to analyze the performance by estimating the rate and the BER.

3.1 Rate-adaptive BCH codes using Fixed CRC check

In the case of the more conventional Fixed CRC check $c(s) = C$, $\forall s$, after each new syndrome is received, the result (if not a decoding failure) is checked with the CRC and accepted only if the CRC check succeeds. If the CRC check does not succeed, a new syndrome is requested. In this scenario, $T_0 = 1$ and $T_1 = T_2 = T_3 = T_{\max}$. We will use the thresholds in the formulas in this section in order to have a general formulation which can be used in the successive sections. We can start with modelling $P_A(s)$ which is the probability of having more than $t(s)$ errors reduced by the probability of accepting a wrong result:

$$P_A(s) = \frac{P_e(> t(s)) - P_E(s)2^{-C}}{D(s)}, \quad (13)$$

where

$$D(s) = \begin{cases} 1 & \text{if } s = T_0, \\ P_e(> t(s-1)) & \text{if } s > T_0. \end{cases} \quad (14)$$

For $s > T_0$, $D(s)$ takes into account that there are more than $t(s-1)$ errors because we have arrived at state s; otherwise, the results would have been accepted in a previous state. $D(s) = 1$ when we have no knowledge of the past, i.e. when $s = T_0$. The expression for $P_B(s)$ for this system is

$$P_B(s) = \begin{cases} P_A(s-1)P_B(s-1) & \text{if } s > T_0, \\ 1 & \text{if } s = T_0 \end{cases} \quad (15)$$

since we can arrive at state s from state $s-1$ due to a decoding failure or to an error revealed by the CRC check. Now we can estimate the expected contribution to the BER, $b(s)$:

$$b(s) = P_B(s)\frac{P_E(s)2^{-C}e_B(s)}{D(s)}, \quad (16)$$

and the expected rate contribution, $r(s)$:

$$r(s) = m_T(s)P_B(s) + C_{T_{\max}}(s), \quad (17)$$

where

$$C_{T_{\max}}(s) = \begin{cases} C & \text{if } s = T_{\max}, \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

since we need to take into account the rate contribution of the CRC check only once because it does not depend on the state s of the decoder.

3.2 Rate-adaptive BCH codes using Syndrome check

The analysis of the proposed RA BCH scheme is based on an accurate analysis of the possible situations at each syndrome request. There are two types of requests: normal syndrome request and check request. As we can see from the rate adaptation algorithm depicted in Figure 2, up to $\delta(s)$ extra checks are performed after decoding with s syndromes. We call this process a *check procedure* starting in s. After every check request, the decoder verifies if the new syndrome satisfies the next step of the Berlekamp-Massey algorithm. If the new syndrome is not compatible with the previously decoded result, the check procedure is stopped, and the latest check is regarded as a normal syndrome request and checked with a new check procedure if needed (i.e. if it is not a decoding failure).

First of all, we shall redefine the estimation of $P_A(s)$:

$$P_A(s) = \frac{P_e(> t(s)) - P_E(s)}{D(s)}. \quad (19)$$

We shall define $P_F(s,i)$, $0 \le i \le \delta(s)$ as an estimate of the probability of failure in detecting that the decoded word is wrong using i extra syndromes given that using $i-1$ extra syndromes, it was not possible to detect the error. We assume that this probability can be approximated by

$$P_F(s,i) = \begin{cases} \Pi(s,i)\Phi(s,i)\Upsilon(s,i) & \text{if } 0 < i \le \delta(s), \\ P_E(s) & \text{if } i = 0, \end{cases} \quad (20)$$

where

$$\Pi(s,i) = \frac{P_e(> t(s+i))}{P_e(> t(s+i-1))}, \quad (21)$$

$$\Phi(s,i) = \frac{2^{N(s+i-1)}}{2^{N(s+i)}} \quad (22)$$

$$\Upsilon(s,i) = \frac{\sum_{k=t(s+i-1)}^{t(s+\delta(s))} P(N_e = k)}{\sum_{k=0}^{t(s+\delta(s))} P(N_e = k)}. \quad (23)$$

For $i \neq 0$, (20) is expressed using three terms (21-23); $\Pi(s,i)$ (21) is the probability of having more than $s+i$ errors given that we have more than $s+i-1$ errors. This knowledge comes from the fact that we made a mistake in the previous check. Increasing the number of syndromes from $s+i-1$ to $s+i$ increases the strength of the code; this phenomenon is expressed using the ratio between the number of words belonging to the code when using $s+i-1$ syndromes and $s+i$ syndromes (22). These two terms can be derived from the same heuristic reasoning used for (7). The last term (23) is a correction factor. Consider the sphere having centre in the correct word

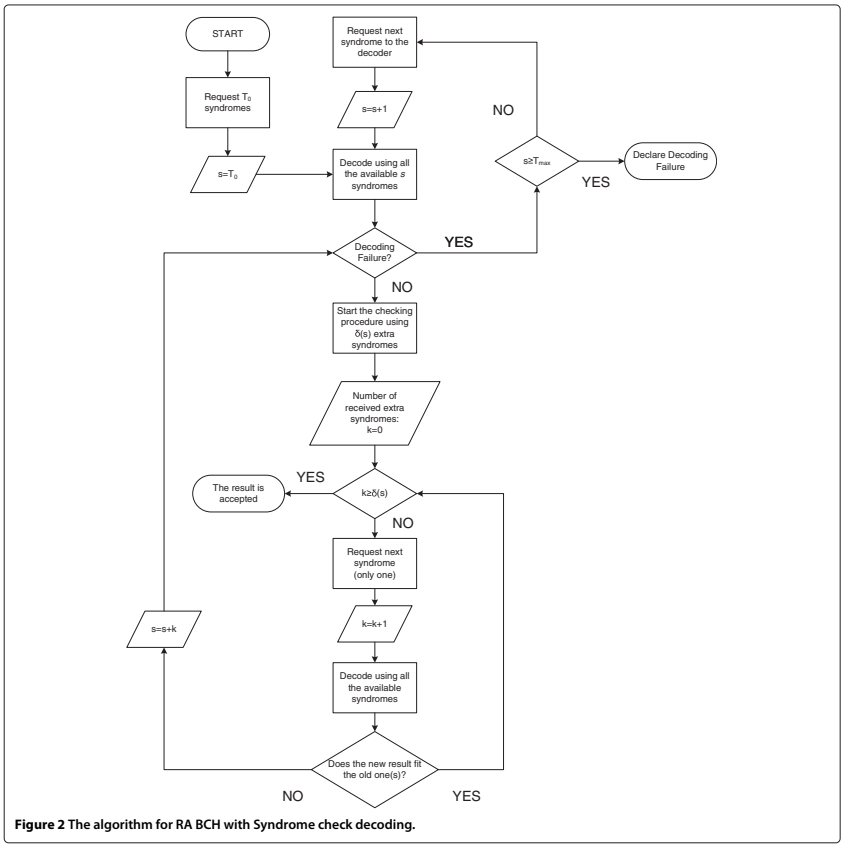


Figure 2 The algorithm for RA BCH with Syndrome check decoding.

and radius $t(s+\delta(s))$. We use the volume (weighted by the error probability) of the shell (from $t(s+i-1)$ to $t(s+\delta(s))$), we still have to explore, as numerator. This correction factor takes into account that the previous two terms overestimate the error probability as the system not only has to make consecutive errors but also identical errors.

We can now analyze the cases leading the system to attempt to decode in state s. The first is when, starting in state $s-1$, the decoding fails in $s-1$; this probability can be expressed as

$$p_0(s) = P_B(s-1)P_A(s-1). \tag{24}$$

It can also be that in state s an extra check fails. This check procedure could have been started in $s-i$, if $\delta(s-i) \geq i$. Let $P_S(s,i)$ denote the probability of revealing an error in state s, given that the latest normal syndrome request was in state $s-i$ and that we have requested i extra check syndromes:

$$P_S(s,i) = \left(\prod_{k=0}^{i-1} P_F(s-i,k)\right)(1-P_F(s-i,i)). \tag{25}$$

Salmistraro *et al. EURASIP Journal on Advances in Signal Processing* 2013, **2013**:166
http://asp.eurasipjournals.com/content/2013/1/166

Let $p_i(s)$ denote the estimate of the probability of arriving in state s due to failure in the extra check procedure using i extra syndromes:

$$p_i(s) = \begin{cases} \frac{P_B(s-i)}{D(s-i)} P_S(s,i) & \text{if } \delta(s-i) \geq i, \\ 0 & \text{otherwise.} \end{cases} \quad (26)$$

Finally, combining (24)-(26) gives

$$P_B(s) = \sum_{k=0}^{\delta_M} p_k(s), \quad s > T_0. \quad (27)$$

The initialization of $P_B(s)$ is $P_B(T_0) = 1$, and $P_B(s) = 0$ if $s < T_0$. Using (7) and (20)-(27), we can analyze the contributions of each state to the total rate and to the total BER. Let $P_T(s)$ denote the probability of failing all the extra checks:

$$P_T(s) = \begin{cases} \prod_{k=1}^{\delta(s)} P_F(s,k) & \text{if } \delta(s) > 0, \\ 1 & \text{otherwise.} \end{cases} \quad (28)$$

The probability of ending the decoding process in state s, taking into account the extra checking, is expressed by

$$P_Q(s) = D(s) - P_e(> t(s)) + P_E(s) P_T(s). \quad (29)$$

Let $F(s)$ denote the rate contributions coming from intermediate states, i.e. states traversed during a check procedure which reveals an error:

$$F(s) = \sum_{i=2}^{\delta(s)} \left(P_E(s)(1 - P_F(s,i)) \Psi(s) \prod_{k=1}^{i-1} P_F(s,k)) \right), \quad (30)$$

where $\Psi(s) = \sum_{k=1}^{i-1} m(s+k)$. The contribution of the rate for the state s is $r(s)$ and it is defined as

$$r(s) = \frac{P_B(s) \left(m_T(s) + \left(\sum_{k=1}^{\delta(s)} m(s+k) \right) P_Q(s) + F(s) \right)}{D(s)}. \quad (31)$$

Scrutinizing (31), we can see the various contributions to the rate at state s: $P_B(s) m_T(s)$ is the contribution coming from the fact that we are in the state s, and $\sum_{k=1}^{\delta(s)} m(s+k)$ is the contribution coming from the extra check which is not taken into account in the successive rate contributions, multiplied by the probability ($P_Q(s)$) of the two events which lead to the termination of the decoding process: the failure $P_E(s) P_T(s)$ and the probability of having a number of errors less than or equal to the correcting power of the code, but having more errors than the correcting power of the previous state $D(s) - P_e(> t(s))$. Finally, the contribution to the BER from the state s is $b(s)$, which is expressed by

$$b(s) = P_B(s) P_E(s) P_T(s) \frac{e_B(s + \delta(s))}{D(s)}. \quad (32)$$

Summing the contributions (8-9) of $r(s)$ (31) and $b(s)$ (32) gives the estimated performance of the RA BCH code.

4 Hierarchical check of rate-adaptive BCH codes

The reliability of the decoded result is a central issue in DSC. Increased reliability can be achieved at the expense of a higher rate. In order to decrease the BER without heavily effecting the rate, we proposed [15] the use of a hierarchy of checks, the first one being performed at block level, as described in Section 3.2, and the other(s) at a higher *macroblock* level where a macroblock is the union of f blocks. In this way, using an additional check of c bits on the macroblock, the cost sustained from each block is reduced to c/f bits at the expense of higher latency.

We implement and analyze only one level of the hierarchical structure. After decoding f blocks using the Syndrome check RA BCH-based system, a CRC spanning the macroblock is generated and it is used to check the decoded results. If the CRC check is not satisfied, an extra syndrome is requested for each of the individual blocks in turn, one at a time, until a block decodes to a different sequence than before. Thereafter, the c-bit CRC check is performed again. This continues until the c-bit CRC check is satisfied.

The model uses the contributions $b(s)$ (32) and $r(s)$ (31) calculated as described in Section 3.2. For the hierarchical CRC, let $b_H(s)$ and $r_H(s)$ denote the BER and rate estimated contributions, respectively. $b_H(s)$ may be derived from $b(s)$ using the same argument introduced in Section 2.3:

$$b_H(s) = 2^{-c} b(s). \quad (33)$$

For what concerns $r_H(s)$, we first estimate $P_\Omega(s)$ which is the probability of starting a hierarchical check procedure in state s:

$$P_\Omega(s) = \frac{P_B(s)(1 - 2^{-c})}{D(s)} P_E(s) P_T(s), \quad (34)$$

and then we can calculate $r_H(s)$ which is the rate contribution coming from the retransmissions required by the hierarchical check, plus the rate $r(s)$:

$$r_H(s) = r(s) + fMP_\Omega(s) \sum_{k=s+\delta(s)+1}^{T_{\max}} (1 - 2^{-c}) \frac{P_E(k)}{P_e(> t(k-1))}. \quad (35)$$

It has to be noted that $r_H(s)$ takes into account the full extra contributions on the whole macroblock coming

from errors of the current block. Since we are interested in average performance, we can add $r_H(s)$ to the rate estimation of the current block. In this way, the contributions to the current block coming from errors in other blocks in the total estimation of the rate are also included.

Finally, the rate and BER estimations (R_H and B_H, respectively) are

$$B_H = \sum_{s=T_0}^{T_{\max}} b_H(s), \tag{36}$$

and

$$R_H = \sum_{s=T_0}^{T_{\max}} r_H(s) + c/f. \tag{37}$$

5 Experimental results

In this section, we present our numerical calculations and simulations in order to validate our theoretical analysis. We compare our methods with LDPCA codes. All the simulations have been conducted using 10^7 blocks or 10^7 macroblocks in the case of hierarchical CRC check. We experimented with three rate-adaptive BCH codes with length $l = 255, 511, 1023$. We focused on a high correlation (low entropy) scenario; hence, we chose low values of p.

Among all the possible strategies, some can be inefficient, i.e. there are strategies having the same (or higher) rate but still achieving worse BER than another strategy. In order to identify the best strategies, a (linear BER) convex hull optimization is performed over the estimated performance of strategies, selecting the set of the strategies $\mathcal{T}$ as the points forming the convex hull.

We first discuss the results for the Syndrome check-based rate-adaptive BCH codes, which will be simply referred to as RA BCH codes in the first part of the section. Figures 3 and 4 depict how the codes behave when changing p. In general, one can expect that the longer the code, the higher the efficiency. In the high-correlation scenario, with short block length, for lower values of p, longer block lengths are more efficient, but for higher error probabilities, shorter block lengths are preferable. We used the model presented in Section 3.2, (31-32), and an actual LDCPA decoder [9] to determine the interval in which at least one of the addressed RA BCH codes, evaluated by the model, outperforms a LDCPA code having a length of 1584. For the RA BCH codes, the strategy having the lowest rate out of the set giving a BER not exceeding that of the LDPCA has been chosen. Figure 3 depicts these results when not using CRC check for the LDPCA code, while in Figure 4, an 8-bit CRC check was used for the LDPCA code. In Figure 3, the three RA BCH codes are reported, while in Figure 4, we only depict the optimal rate over the RA BCH codes class, for each value of the conditional entropy $H(X|Y)$ (per bit) we evaluate. As it can be seen for lower error probabilities (lower conditional entropy), the best-performing code is the longest one. As we have previously said, in order to correct the errors, plM bits are required on average (plus check bits). The minimum (and ideal) number of bits on average is $lH(X|Y)$. Comparing the two terms gives $M \approx H(X|Y)/p$, and it can be noticed from the graphs that the M of the optimal code is well approximated by $M = 1 + H(X|Y)/p$. The discrepancy can be due to the inability of the RA BCH code to reach the entropy coming from the overhead due to the check. The ratio $H(X|Y)/p$ in the analyzed scenario is a decreasing function, motivating the behaviour of the codes. It may be noted that an $H(X|Y)$ between 0.25 and 0.3 corresponds to (maximum) compression factors of $3 - -4$. For compression at these factors and above, corresponding

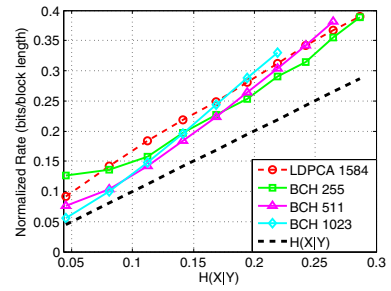


Figure 3 RA BCH code performance compared with an LDPCA code with no CRC check. BCH codes have the same or better BER than LDPCA at each $H(X|Y)$ point.

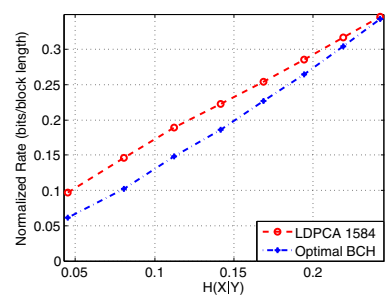


Figure 4 Optimal RA BCH performance compared with an LDPCA code with 8-bit CRC check.

Salmistraro *et al. EURASIP Journal on Advances in Signal Processing* 2013, **2013**:166
http://asp.eurasipjournals.com/content/2013/1/166

to reasonably compressible material, the well-selected RA BCH performs better than LPDCA with block length of 1584, even though the RA BCH block lengths are shorter. For the highest compressible point tested ($p = 0.005$), the LDPCA almost requires twice as many bits as the RA BCH of length 1023.

In Figure 5, we present a complete analysis for $p = 0.01$, reporting models and actual performance of the proposed method, and our RA BCH codes are compared against well-known rate-adaptive and fixed-rate alternatives. The performance for LDPCA codes with lengths of 1584 and 396 without CRC check and with an 8-bit CRC check, as well as the performance for fixed-rate BCH and LDPC, codes are reported. We present the performance of the three chosen RA BCH codes and the corresponding estimated performance based on the presented model. The rate-adaptive BCH with length 1023 performs the best. Also, RA BCH 511 and RA BCH with hierarchical check have good performance. They all perform significantly better than LDPCA 396 and 1584, being significantly closer to $H(X|Y)$ than these. Furthermore, it may be seen that our model is able to predict the simulated performance of the decoder with high accuracy. We also report the performance of the hierarchical check adding a higher level check to the strategies of the 511 BCH code. The increase of the reliability is high compared with the increase in the rate, making the hierarchical check an interesting solution if higher latency can be accepted. For the hierarchical check, we used the same strategies as for the normal RA BCH codes, and we performed (linear BER) convex hull optimization over the parameter set $f \in \{2, 4, 6, 8\}$. The strength of the CRC check used is 8 bits. The model presented in Section 4 is able to predict the behaviour of the code. The hierarchical check increases the latency; hence, in a sense, it is like using a longer block length. Therefore, we report the performance for a LDPCA code having a block length of 4800 (and data increment of 75 bits), which is close to the longest analyzed macroblock length $511 \times 8 = 4088$ (the code having block length 4800 has been produced using the same approach of [9,23]). We also present, as a term of comparison, a theoretical BER bound for a fixed-rate error-correcting code of length 1023: as we can see, the rate-adaptive BCH codes are able to outperform this bound. The bound is based on Theorem 33 in [19], which allows the calculation of an upper bound for achievable block error rate. We adapted the bound assuming that non-decodable codewords have a BER which is twice as high as the input BER; this will tend to overestimate the BER since decoding errors do not always double the number of errors. Comparing with the fixed-rate (BCH and LDPC) codes, the rate-adaptive codes perform, as expected, significantly better, especially having a much faster decrease in BER. For these results, the

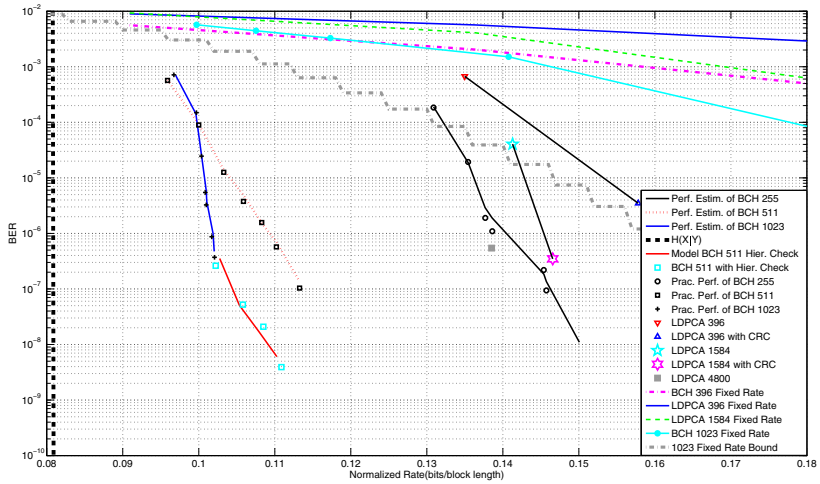


Figure 5 Analysis and comparison of the code performance for $p = 0.01$.

achievable theoretical bound for a fixed rate seems to coincide with the performance of LDPCA. Again, we can note that the best RA BCH codes have significantly better performance.

In Figure 6, the performance of a Fixed CRC-based rate-adaptive BCH code for $p = 0.04$ is presented. CRC strengths $C = M$ and $C = 2M$ are used. In this case, the best Syndrome check-based RA BCH code is able to outperform the best available Fixed CRC check-based RA BCH code. In particular for $p \geq 0.015$, the Fixed CRC version of a code is unable to compete with its variable check syndrome version. Secondly, we have also validated our model developed in Section 3.1, which is able to predict the behaviour of the code. We also present the performance comparison between the codes having a length of 255 when using Variable CRC check and Syndrome check. We can see that Syndrome check outperforms the Variable CRC check system. It can be seen that the Syndrome check is also able to provide more flexibility. This is due to the fact that when using Syndrome check we can be more aggressive: if the check rejects the decoding considered, we can reuse the bits requested for checking to decode, but in the same situation, when using the Variable CRC check we cannot go back, and we will use a very strong CRC check in future decoding attempts. Obviously, this leads to less robust strategies (the strategies leading to higher BER) for the Syndrome check approach, but when examining strategies having comparable BER, Syndrome check is superior for what concerns the rate.

Since the best-performing solution among the ones presented is the Syndrome check-based RA BCH code, the reliability of the model for this code is summarized in Table 1. The performance of the model for $p \leq 0.04$ was analyzed since the BCH codes outperform the LDPCA code for such probabilities (Figures 3 and 4) for the range of code lengths considered.

As can be seen from Figure 5, the model is less precise with respect to the prediction of the BER, while the rate is usually well predicted; in the studied scenarios, the maximum difference between simulated and estimated rates was less than 0.04%. In order to summarize all the numerical results, a measure of the reliability of the prediction of the models is proposed: the mean absolute BER difference for a given code and a given error probability denoted as $\Gamma(l, p)$.

$$\Gamma(l,p) = \frac{1}{|\mathcal{T}|} \sum_{\mathbf{T} \in \mathcal{T}} \frac{|B_S(l,p,\mathbf{T}) - B(l,p,\mathbf{T})|}{B_S(l,p,\mathbf{T})}, \tag{38}$$

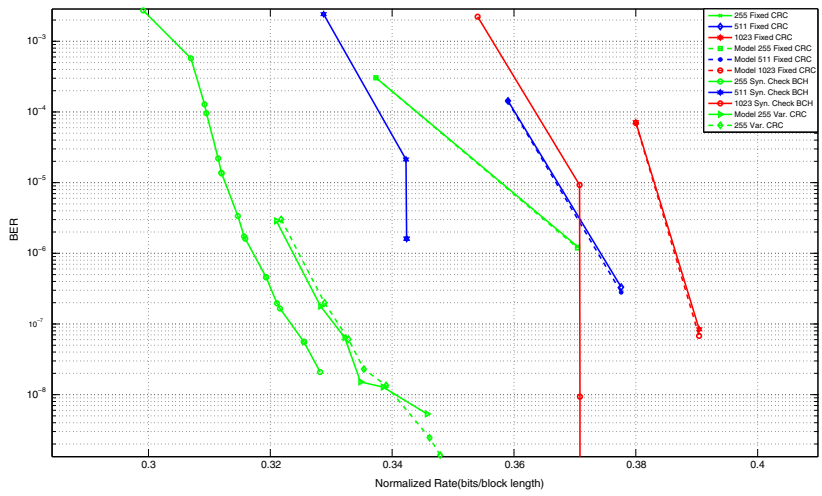


Figure 6 Performance for RA BCH codes with fixed and Variable CRC for $p = 0.04$. Fixed and Variable CRC check-based rate-adaptive BCH code for $p = 0.04$, the model, and actual decoder behaviour are depicted. The performance of the Syndrome check BCH code is also depicted for comparison.

Table 1 Evaluation of BER model accuracy $\Gamma(p, l)$

l	Syndrome check-based model $\Gamma(p, l)$
$p = 0.01$	
255	0.2395
511	0.0920
1023	0.0951
$p = 0.015$	
255	0.2064
511	0.0759
1023	0.1609
$p = 0.02$	
255	0.0989
511	0.1092
1023	0.4271
$p = 0.025$	
255	0.1076
511	0.1239
1023	0.1057
$p = 0.03$	
255	0.0305
511	0.0545
1023	0.0229
$p = 0.035$	
255	0.0772
511	0.0798
1023	0.0623
$p = 0.04$	
255	0.0586
511	0.3445
1023	0.0145

where $B_S(l, p, \mathbf{T})$ is the BER estimated by simulations and $B(l, p, \mathbf{T})$ is the BER calculated using the model proposed in Section 3.2.

In this work, we focused on a high-correlation scenario $p \leq 0.04$ ($H(X|Y) \leq 0.24$), but we also assessed the performance of our code for $p = 0.1$ ($H(X|Y) = 0.47$) in order to assess the robustness of the proposed code in terms of the ability of the proposed codes (including checking strategy) to perform reasonably well outside the error interval they have been developed for. We define, for this purpose, a rate loss metric:

$$\Delta_g = 100 \times \frac{R_{\mathrm{BCH}} - R_{\mathrm{LDPCA}}}{R_{\mathrm{BCH}}}, \tag{39}$$

where R_{LDPCA} is the normalized rate of the LDPCA code and R_{BCH} is the normalized rate of the BCH code, both obtained by simulations and not the model. For the BCH codes, we chose $l = 255$, and for the LDPCA codes, we chose $l = 396, 1584$ with an 8-bit CRC check. In this case, the LDPCA codes outperform, in terms of the normalized rate, the BCH code by $\Delta_g = 7.6\%$ and $\Delta_g = 10\%$, respectively, for similar BERs. Using the same metric for $p = 0.005$ and comparing the RA BCH having $l = 1023$ with the LDPCA code having $l = 1584$ and an 8-bit CRC check, we obtain $\Delta_g = -58\%$. It has to be noted that for low-correlation scenarios our system is not able to outperform the LDPCA codes, but based on the relatively small loss at $p = 0.1$, we note that in case of, e.g. varying values of p, the RA BCH codes do provide robustness outside the interval for which it performs better than LDPCA.

Our new model is also able to provide more accurate performance estimates than the model presented in [15] for a given strategy in almost all the cases. The improved accuracy is high: for example, one of the best cases is for $p = 0.01$, $l = 255$: the estimated BER by the proposed model for a given strategy is 1.85×10^{-7}, while the simulated performance of the real decoder is BER = 2.20×10^{-7}, and the model of [15] predicted BER = 8.89×10^{-7}. Among the tested scenarios, the model of [15] is able to obtain better accuracy than the proposed one only in a few cases, but even in these cases, the results of the proposed model are still sufficiently accurate. The worst of these cases is $l = 1023$, $p = 0.035$: for the strategy having the highest difference, the BER is 9.36×10^{-6}, the estimated BER by [15] is 9.16×10^{-6}, and the predicted BER using our proposed model is 8.21×10^{-6}.

The adaptive BCH and the LDPCA approach may also be compared with respect to complexity. Two aspects are interesting: the encoder and the decoder complexity.

The BCH encoder produces syndromes of (mostly) M bits and each of them may be produced with l division steps with an M degree polynomial. The number of syndromes needed is variable on average around pl, so the number of operations is growing as $plMl$. For the LDPCA encoder, approximately the same number of syndrome bits should be produced, and if it is implemented as an ordinary matrix multiplication, the number of operations becomes the same. Actually, the number of operations could be reduced using the sparse nature of the parity check matrix since it depends on the number of edges in the bipartite graph for the code which grows with l, but overall, we estimate the encoder complexity to be similar for the two approaches.

The BCH decoder uses a few operations to perform the next step in the Berlekamp-Massey algorithm for each received syndrome, but then a search for roots in the error locator has to be done. The complexity is proportional to l and to the current number of syndromes, and it is done each time a new syndrome is requested. The LDPCA decoder uses the approach of [9] and performs

100 iterations for each new set of syndrome bits. The complexity is difficult to estimate, but in our implementation which was not optimized in any way, the execution time of the BCH decoder was around 16 times less than that of the LDPCA decoder for a typical case: $p = 0.04$, $l = 1023$, LDPCA 1584 as benchmark. In order to account for the different lengths, in this comparison, we normalized the decoding time of both codes by their respective lengths.

6 Conclusions

In this work, we propose and analyze the concept and use of rate-adaptive BCH codes for DSC. We demonstrated that these codes can outperform the rate-adaptive LDPCA codes when employed in a high-correlation scenario using short block lengths. Checking strategies are applied in order to increase the reliability of the decoded results. We presented and analyzed an adaptive strategy together with the RA BCH and, for comparison, both a fixed and an adaptive CRC. Finally, we devised and tested models which were able to correctly predict the performance of the codes. These models are employed to find the optimal code and check strategy knowing only the probability p. Furthermore, the reliability of our scheme was increased using a hierarchical CRC, which consists of a CRC spanning more blocks in order to divide the cost of a check between the them, obtaining a good trade-off between the reliability and increase of the rate at the expense of increased latency.

Appendix

Rate-adaptive BCH codes using Variable CRC check

When dealing with the Variable CRC check approach, we can use an approach similar to the one we have seen in the Fixed CRC check, but now the number of CRC check bits used for each state s is variable. In fact, $c(s)$ bits are required only if no CRC bits have been requested in the past states; hence, we can now define $C_{\text{avg}}(s)$ as the average number of bits used to check an acceptable decoded solution when in state s and $\Lambda(s)$ as the average reliability improvement due to the CRC check in state s.

In this case, formulas (13), (16), and (17) in Section 3.1 can be adapted, keeping $P_B(s)$ (15) unchanged:

$$P_A(s) = \frac{P_e(> t(s)) - P_E(s)\Lambda(s)}{D(s)} \tag{40}$$

$$b(s) = P_B(s)\frac{P_E(s)\Lambda(s)e_B(s)}{D(s)}, \tag{41}$$

$$r(s) = P_B(s) \times \left(m_T(s) + \frac{D(s) - P_e(> t(s)) + P_E(s)\Lambda(s)}{D(s)} C_{\text{avg}}(s)\right). \tag{42}$$

Now the main problem is to find estimates for $\Lambda(s)$ and $C_{\text{avg}}(s)$. We can start by defining $P_R(k|s)$, which is the probability of requesting a CRC check in state k, $T_0 \le k \le s$ given that the result in state s is acceptable, and hence, it should be checked by a CRC. If $c(s) = 0$ but a CRC has already been requested in a past state k, the check is carried out as well. First of all, we need to estimate the probability of a decoding error in state k given that we are checking a decoding in state s, $P_{Ec}(k|s)$:

$$P_{Ec}(k|s) = \begin{cases} L(s) & \text{if } T_0 \le k < s, \\ L(s) + (1 - P_e(> t(s))) & \text{otherwise,} \end{cases} \tag{43}$$

where

$$L(s) = \frac{\sum_{e=0}^{t(s)} \binom{l}{e}}{2^{N(s)}}. \tag{44}$$

Finally, we have

$$P_R(k|s) = \prod_{i=T_0}^{k} (1 - P_{Ec}(i|s)), \tag{45}$$

with $P_R(k|s)$, we can calculate $\Lambda(s)$:

$$\Lambda(s) = \sum_{k=T_0}^{s} 2^{-c(k)} \frac{P_R(k|s)}{\sum_{i=T_0}^{s} P_R(i|s)} \tag{46}$$

and $C_{\text{avg}}(s)$:

$$C_{\text{avg}}(s) = \sum_{k=T_0}^{s} c(k) \frac{P_R(k|s)}{\sum_{i=T_0}^{s} P_R(i|s)}. \tag{47}$$

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
The authors express their gratitude to Dr. David Varodayan for the valuable support in generating the LDPCA code having a block length of 4800.

Received: 15 April 2013 Accepted: 11 October 2013
Published: 5 November 2013

References

1. D Slepian, J Wolf, Noiseless coding of correlated information sources. IEEE Trans. Inf. Theory **19**(4), 471–480 (1973)
2. A Wyner, J Ziv, The rate-distortion function for source coding with side information at the decoder. IEEE Trans. Inf. Theory **22**, 1–10 (1976)
3. R Puri, A Majumdar, K Ramchandran, PRISM: a video coding paradigm with motion estimation at the decoder. IEEE Trans. Image Process. **16**(10), 2436–2448 (2007)
4. C Yeo, K Ramchandran, Robust distributed multiview video compression for wireless camera networks. IEEE Trans. Image Process. **19**(4), 995–1008 (2010)
5. B Girod, A Aaron, S Rane, D Rebollo-Monedero, Distributed video coding. Proc. IEEE **93**, 71–83 (2005)
6. X Artigas, J Ascenso, M Dalai, S Klomp, D Kubasov, M Ouaret, The DISCOVER codec: architecture, techniques and evaluation, in *Proceedings of the 2007 Picture Coding Symposium*, (Lisbon, Portugal, November 2007)

7. A Wyner, A Recent results in the Shannon theory. IEEE Trans. Inf. Theory **20**, 2–10 (1974)
8. V Stankovic, A Liveris, Z Xiong, C Georghiades, On code design for the Slepian-Wolf problem and lossless multiterminal networks. IEEE Trans. Inf. Theory **52**(4), 1495–1507 (2006)
9. D Varodayan, A Aaron, B Girod, Rate-adaptive distributed source coding using low-density parity-check codes. EURASIP Signal Process. J. Spec Section Distributed Source Coding. **86**, 3123–3130 (2006)
10. M Grangetto, E Magli, G Olmo, Distributed arithmetic coding for the Slepian-Wolf problem. IEEE Trans. Signal Process. **57**(6), 2245–2257 (2009)
11. M Ali, M Kuijper, Source coding with side information using list decoding, in *Proceedings of the IEEE ISIT 2010*, (Austin, TX, USA, June 2010), pp. 91–95
12. S Malinowski, X Artigas, C Guillemot, L Torres, Distributed coding using punctured quasi-arithmetic codes for memory and memoryless sources. IEEE Trans. Signal Process. **57**(10), 4154–4158 (2009)
13. M Vaezi, F Labeau, Wyner-Ziv coding in the real field based on BCH-DFT codes. arXiv:1301.0297 (2013)
14. P Treeviriyanupab, P Sangwongngam, K Sripimanwat, O Sangaroon, BCH-based Slepian-Wolf coding with feedback syndrome decoding for quantum key reconciliation, in *Proceedings of ECTI-CON 2012*, (Thailand, May 2012), pp. 1–4
15. S Forchhammer, M Salmistraro, X Larsen, KJ Huang, HV Luong, Rate-adaptive BCH coding for Slepian-Wolf coding of highly correlated sources, in *Proceedings of Data Compression Conference (DCC), 2012*, (Snowbird, UT, USA, April 2012), pp. 237–246
16. H Luong, L Rakêt, X Huang, S Forchhammer, Side information and noise learning for distributed video coding using optical flow and clustering. IEEE Trans. Image Process. **21**(12), 4782–4796 (2012)
17. J Slowack, S Mys, J Škorupa, N Deligiannis, P Lambert, A Munteanu, RV de Walle, Rate-distortion driven decoder-side bitplane mode decision for distributed video coding. Signal Process. Image Commun. **25**(9), 660–673 (2010)
18. RE Blahut, *Algebraic Codes for Data Transmission,* 1st edn. (Cambridge University Press, Cambridge, 2003)
19. Y Polyanskiy, H Poor, S Verdu, Channel coding rate in the finite blocklength regime. IEEE Trans. Inf. Theory **56**(5), 2307–2359 (2010)
20. A Shiozaki, Adaptive type-II hybrid ARQ system using BCH codes. Trans. IEICE **E75-A**, 1071–1075 (1992)
21. R McEliece, L Swanson, On the decoder error probability for Reed - Solomon codes (Corresp.) IEEE Trans. Inf. Theory **32**(5), 701–703 (1986)
22. T Kløve, V Korzhik, *Error Detecting Codes* (Kluwer Academic, Boston, 1995)
23. D Varodayan, YC Lin, B Girod, Adaptive distributed source coding. IEEE Trans. Image Process. **21**(5), 2630–2640 (2012)

doi:10.1186/1687-6180-2013-166
Cite this article as: Salmistraro *et al.*: **Rate-adaptive BCH codes for distributed source coding.** *EURASIP Journal on Advances in Signal Processing* 2013 **2013**:166.

Appendix B

Test Material

The first frame of each test sequence used for the experiments is shown below. The original resolution is reported, for the lower resolution used in some of the works, please refer to the corresponding paper in Appendix A. The sequence *Ballet* is not shown here, but it is provided in the Introduction, in Fig. 2.12.

B.1 Multiview Video Sequences

Texture Depth

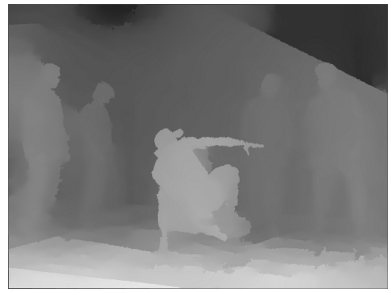
Figure B.1: *Balloons*, view 3, 1024×768, 30 fps.

Texture

Depth

Figure B.2: *BookArrival*, view 8, 1024 × 768, 16.67 fps.

Texture



Depth

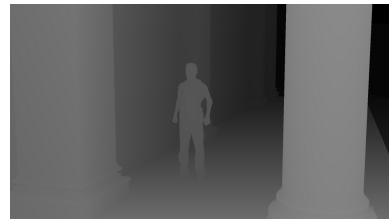
Figure B.3: *Breakdancers*, view 4, 1024 × 768, 15 fps.

Texture

Depth

Figure B.4: *Cafe*, view 3, 1920 × 1080, 30 fps.

Texture



Depth

Figure B.5: *Dancer*, view 5, 1920 × 1088, 25 fps.

Texture

Depth

Figure B.6: *Kendo*, view 3, 1024 × 768, 30 fps.

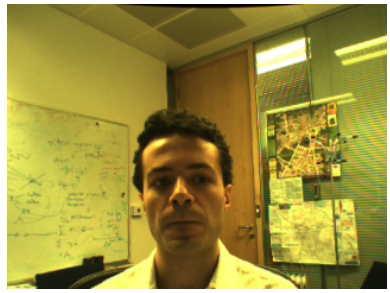
Texture

Depth

Figure B.7: *Lovebird1*, 1024 × 768, 30 fps.

Figure B.8: *Outdoor*, view 8, 1024×768, 16.67 fps.

Left



Right

Figure B.9: *AC*, 320×240, 15 fps.

Left

Right

Figure B.10: *IU*, 320×240, 15 fps.

Left

Right

Figure B.11: *VK*, 320 × 240, 15 fps.

Left

Right

Figure B.12: *IUJW*, 320 × 240, 15 fps.

B.2 Monoview Video Sequences

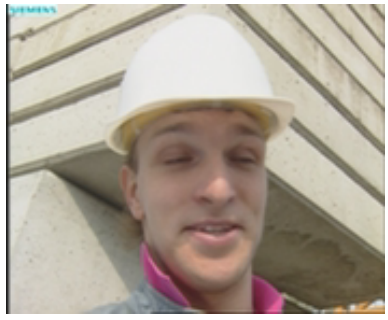


Figure B.13: *Foreman*, 176 × 144, 15 fps.

Figure B.14: *Soccer*, 176 × 144, 15 fps.

Figure B.15: *Hall*, 176 × 144, 15 fps.

Figure B.16: *Coast*, 176 × 144, 15 fps.

B.3 Multiview Images plus Depth

Texture Depth

Figure B.17: *Teddy*, view 2, 1800 × 1500.

List of Acronyms

ARPS	Adaptive Rood Pattern Search
BCH	Bose-Chaudhuri-Hocquenghem
BER	Bit Error Rate
CRC	Cyclic Redundancy Check
DCVP	Disparity Compensated View Prediction
DIBR	Depth-Image-Based-Rendering
DISCOVER	DIStributed COding for Video sERvices
DP	Difference Projection
DSC	Distributed Source Coding
DVC	Distributed Video Coding
FOV	Field Of View
FVV	Free Viewpoint Video
GOP	Group Of Pictures
KF	Key Frame
LDPC	Low Density Parity Check
LDPCA	Low Density Parity Check Accumulate
MB	MacroBlock

MCI	Motion Compensated Interpolation
M-DVC	Multiview Distributed Video Coding
ME	Motion Estimation
MH	Multi-Hypothesis
MSE	Mean Squared Error
MV	Motion Vector
MVC	Multiview Video Coding
MVD	Multiview Video-plus-Depth
MVME	MultiView Motion Estimation
MVSim	Motion Vector Similarity
OBDC	Overlapped Block Disparity Compensation
OBMC	Overlapped Block Motion Compensation
OF	Optical Flow
PRISM	Power-efficient, Robust, hIgh-compression, Syndrome-based Multimedia coding
PSNR	Peak Signal-to-Noise Ratio
RA	Rate Adaptive
RCPT	Rate Compatible Punctured Turbo
RD	Rate-Distortion
SAD	Sum of Absolute Differences
SI	Side Information
SLEP	Systematic Lossy Error Protection
TDOF	Time Disparity Optical Flow

VSN	Video Sensor Network
WSN	Wireless Sensor Network
WZ	Wyner-Ziv

Bibliography

[1] Z. Xiong, A. Liveris, and S. Cheng, "Distributed Source Coding for Sensor Networks," *IEEE Signal Processing Magazine*, vol. 21, no. 5, pp. 80–94, Sept 2004.

[2] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed Video Coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 71–83, Jan 2005.

[3] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560 –576, July 2003.

[4] G. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[5] J. Ahmad, H. Khan, and S. Khayam, "Energy Efficient Video Compression for Wireless Sensor Networks," in *43rd Annual Conference on Information Sciences and Systems, 2009 (CISS 2009)*, March 2009, pp. 629–634.

[6] F. Pereira, "Distributed Video Coding: Basics, Main Solutions and Trends," in *IEEE International Conference on Multimedia and Expo, 2009 (ICME 2009)*, Jul. 2009, pp. 1592–1595.

[7] A. Vetro, T. Wiegand, and G. Sullivan, "Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 626 –642, April 2011.

[8] A. Vetro and D. Tian, "Analysis of 3D and Multiview Extensions of the Emerging HEVC Standard," in *SPIE Applications of Digital Image Processing XXXV, 2012*, 2012, pp. 84 990Y/1–7.

[9] C. Guillemot, F. Pereira, L. Torres, T. Ebrahimi, R. Leonardi, and J. Ostermann, "Distributed Monoview and Multiview Video Coding," *IEEE Signal Processing Magazine,*, vol. 24, no. 5, pp. 67–76, Sep. 2007.

[10] D. Slepian and J. Wolf, "Noiseless Coding of Correlated Information Sources," *IEEE Transactions on Information Theory*, vol. 19, no. 4, pp. 471 – 480, July 1973.

[11] A. Wyner and J. Ziv, "The Rate-Distortion Function for Source Coding with Side Information at the Decoder," *IEEE Transactions on Information Theory*, vol. 22, no. 1, pp. 1–10, Jan. 1976.

[12] A. Wyner, "Recent Results in the Shannon Theory," *IEEE Transactions on Information Theory*, vol. 20, no. 1, pp. 2–10, Jan 1974.

[13] R. Puri and K. Ramchandran, "PRISM: A Video Coding Architecture Based on Distributed Compression Principles," in *Allerton Conference on Communication, Control and Computing*, 2002.

[14] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv Coding of Motion Video," in *Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, 2002*, vol. 1, Nov 2002, pp. 240–244 vol.1.

[15] R. Puri, A. Majumdar, and K. Ramchandran, "PRISM: A Video Coding Paradigm With Motion Estimation at the Decoder," *IEEE Transactions on Image Processing*, vol. 16, no. 10, pp. 2436–2448, Oct 2007.

[16] D. Varodayan, A. Aaron, and B. Girod, "Rate-Adaptive Codes for Distributed Source Coding," *EURASIP Signal Processing Journal, Special Section on Distributed Source Coding*, vol. 86, no. 11, pp. 3123 – 3130, 2006.

[17] F. Pereira, C. Brites, J. Ascenso, and M. Tagliasacchi, "Wyner-Ziv Video Coding: a Review of the Early Architectures and Further Developments," in *IEEE International Conference on Multimedia and Expo, 2008 (ICME 2008)*, June 2008, pp. 625–628.

[18] D. Rowitch and L. Milstein, "On the Performance of Hybrid FEC/ARQ Systems Using Rate Compatible Punctured Turbo (RCPT) Codes," *IEEE Transactions on Communications*, vol. 48, no. 6, pp. 948–959, Jun 2000.

[19] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER Codec: Architecture, Techniques and Evaluation," in *Picture Coding Symposium, 2007 (PCS 2007)*, 2007.

[20] S. L. P. Yasakethu, W. A. R. J. Weerakkody, W. A. C. Fernando, F. Pereira, and A. Kondoz, "An Improved Decoding Algorithm for DVC Over Multipath Error Prone Wireless Channels," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 10, pp. 1543–1548, Oct 2009.

[21] A. Heidarzadeh and F. Lahouti, "On Robust Syndrome-Based Distributed Source Coding over Noisy Channels using LDPC Codes," in *IEEE International Conference on Signal Processing and Communications, 2007 (ICSPC 2007)*, Nov 2007, pp. 400–403.

[22] [Online]. Available: http://www.discoverdvc.org/

[23] D. Kubasov, K. Lajnef, and C. Guillemot, "A Hybrid Encoder/Decoder Rate Control for Wyner-Ziv Video Coding with a Feedback Channel," in *IEEE International Workshop on Multimedia Signal Processing, 2007 (MMSP 2007)*, Oct 2007, pp. 251–254.

[24] C. Brites and F. Pereira, "Encoder Rate Control for Transform Domain Wyner-Ziv Video Coding," in *IEEE International Conference on Image Processing, 2007 (ICIP 2007)*, vol. 2, Sept 2007, pp. II – 5–II – 8.

[25] D. Kubasov, J. Nayak, and C. Guillemot, "Optimal Reconstruction in Wyner-Ziv Video Coding with Multiple Side Information," in *IEEE International Workshop on Multimedia Signal Processing, 2007 (MMSP 2007)*, Oct 2007, pp. 183–186.

[26] J. Ascenso and F. Pereira, "Adaptive Hash-Based Side Information Exploitation for Efficient Wyner-Ziv Video Coding," in *IEEE International Conference on Image Processing, 2007 (ICIP 2007)*, vol. 3, Sept 2007, pp. III – 29–III – 32.

[27] C. Brites, J. Ascenso, and F. Pereira, "Studying Temporal Correlation Noise Modeling for Pixel Based Wyner-Ziv Video Coding," in *IEEE International Conference on Image Processing, 2006 (ICIP 2006)*, Oct 2006, pp. 273–276.

[28] [Online]. Available: http://www.discoverdvc.org/cont_Publications.html

[29] X. Huang and S. Forchhammer, "Cross-band Noise Model Refinement for Transform Domain Wyner-Ziv Video Coding," *Signal Processing: Image Communication*, vol. 27, no. 1, pp. 16–30, 2012.

[30] ——, "Improved Side Information Generation for Distributed Video Coding," in *IEEE International Workshop on Multimedia Signal Processing, 2008 (MMSP 2008)*, Oct 2008, pp. 223–228.

[31] A. Abou-Elailah, F. Dufaux, J. Farah, M. Cagnazzo, and B. Pesquet-Popescu, "Fusion of Global and Local Motion Estimation for Distributed Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 1, pp. 158–172, Jan 2013.

[32] X. Huang, L. Raket, H. Van Luong, M. Nielsen, F. Lauze, and S. Forchhammer, "Multi-hypothesis Transform Domain Wyner-Ziv video Coding Including Optical Flow," in *IEEE International Workshop on Multimedia Signal Processing, 2011 (MMSP 2011)*, 2011, pp. 1–6.

[33] L. L. Rakêt, L. Roholm, A. Bruhn, and J. Weickert, "Motion Compensated Frame Interpolation with a Symmetric Optical Flow Constraint," in *Advances in Visual Computing*. Springer, 2012, vol. 7431 of *Lecture notes in computer science*, pp. 447–457.

[34] I. Daribo, W. Miled, and B. Pesquet-Popescu, "Joint Depth-motion Dense Estimation for Multiview Video Coding," *Journal of Visual Communication and Image Representation*, vol. 21, no. 5 - 6, pp. 487 – 497, 2010.

[35] A. Tomé and F. Pereira, "Low Delay Distributed Video Coding with Refined Side Information," *Signal Processing: Image Communication*, vol. 26, no. 4-5, pp. 220 – 235, 2011.

[36] C. Brites and F. Pereira, "An Efficient Encoder Rate Control Solution for Transform Domain Wyner-Ziv Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 9, pp. 1278 – 1292, Sept 2011.

[37] J. Slowack, J. Skorupa, N. Deligiannis, P. Lambert, A. Munteanu, and R. Van De Walle, "Distributed Video Coding With Feedback Channel Constraints," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 7, pp. 1014–1026, July 2012.

[38] J. Slowack, S. Mys, J. Skorupa, N. Deligiannis, P. Lambert, A. Munteanu, and R. V. de Walle, "Rate-Distortion Driven Decoder-Side Bitplane Mode Decision for Distributed Video Coding," *Signal Processing: Image Communication*, vol. 25, no. 9, pp. 660 – 673, 2010.

[39] C. Brites, J. Ascenso, and F. Pereira, "Learning Based Decoding Approach for Improved Wyner-Ziv Video Coding," in *Picture Coding Symposium, 2012 (PCS 2012)*, May 2012, pp. 165–168.

[40] X. HoangVan, J. Ascenso, and F. Pereira, "Improved B-slices DIRECT Mode Coding Using Motion Side Information," in *International Workshop on Image Analysis for Multimedia Interactive Services, 2012 (WIAMIS 2012)*, May 2012, pp. 1–4.

[41] C. Yeo and K. Ramchandran, "Robust Distributed Multiview Video Compression for Wireless Camera Networks," *IEEE Transactions on Image Processing*, vol. 19, no. 4, pp. 995–1008, April 2010.

[42] S. E. Chen and L. Williams, "View Interpolation for Image Synthesis," in *20th Annual Conference on Computer Graphics and Interactive Techniques, 1993 (SIGGRAPH 1993)*, 1993, pp. 279–288.

[43] D. Varodayan, A. Mavlankar, M. Flierl, and B. Girod, "Distributed Grayscale Stereo Image Coding with Unsupervised Learning of Disparity," in *Data Compression Conference, 2007 (DCC 2007)*, March 2007, pp. 143–152.

[44] D. Chen, D. Varodayan, M. Flierl, and B. Girod, "Wyner-Ziv Coding of Multiview Images with Unsupervised Learning of Two Disparities," in *IEEE International Conference on Multimedia and Expo, 2008 (ICME 2008)*, June 2008, pp. 629–632.

[45] ——, "Wyner-Ziv Coding of Multiview Images with Unsupervised Learning of Disparity and Gray Code," in *IEEE International Conference on Image Processing, 2008 (ICIP 2008)*, Oct 2008, pp. 1112–1115.

[46] M. Ouaret, F. Dufaux, and T. Ebrahimi, "Multiview Distributed Video Coding with Encoder Driven Fusion," in *European Conference on Signal Processing, 2007 (EUSIPCO 2007)*, 2007.

[47] X. Artigas, F. Tarrés, and L. Torres, "Comparison of Different Side Information Generation Methods for Multiview Distributed Video Coding," in *International Conference on Signal Processing and Multimedia Applications, 2007 (SIGMAP 2007)*, 2007.

[48] J. Areia, J. Ascenso, C. Brites, and F. Pereira, "Wyner-Ziv Stereo Video Coding using a Side Information Fusion Approach," in *IEEE International Workshop on Multimedia Signal Processing, 2007 (MMSP 2007)*, October 2007, pp. 453–456.

[49] T. Maugey, W. Miled, M. Cagnazzo, and B. Pesquet-Popescu, "Fusion Schemes for Multiview Distributed Video Coding," in *European Conference on Signal Processing, 2009 (EUSIPCO 2009)*, vol. 1, Glasgow, Scotland, 2009, pp. 559–563.

[50] F. Dufaux, "Support Vector Machine Based Fusion for Multi-view Distributed Video Coding," in *Conference on Digital Signal Processing, 2011 (DSP 2011)*, July 2011, pp. 1–7.

[51] M. Ouaret, F. Dufaux, and T. Ebrahimi, "Iterative Multiview Side Information for Enhanced Reconstruction in Distributed Video Coding," *EURASIP Journal on Image and Video Processing - Special issue on distributed video coding*, vol. 2009, pp. 3:1–3:17, January 2009.

[52] V. Thirumalai and P. Frossard, "Joint Reconstruction of Multiview Compressed Images," *IEEE Transactions on Image Processing*, vol. 22, no. 5, pp. 1969–1981, May 2013.

[53] L. C. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality Video View Interpolation Using a Layered Representation," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 600–608, Aug. 2004.

[54] P. Kauff, N. Atzpadin, C. Fehn, M. Muller, O. Schreer, A. Smolic, and R. Tanger, "Depth Map Creation and Image-based Rendering for Advanced 3DTV Services Providing Interoperability and Scalability," *Signal Processing: Image Communication - Special issue on three-dimensional video and television*, vol. 22, no. 2, pp. 217 – 234, 2007.

[55] K. Muller, P. Merkle, and T. Wiegand, "3-D Video Representation Using Depth Maps," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 643–656, April 2011.

[56] Y. Morvan, D. Farin, and P. de With, "Depth-Image Compression Based on an R-D Optimized Quadtree Decomposition for the Transmission of Multiview Images," in *IEEE International Conference on Image Processing, 2007 (ICIP 2007)*, vol. 5, Sept 2007, pp. V – 105–V – 108.

[57] G. Shen, W.-S. Kim, S. Narang, A. Ortega, J. Lee, and H. Wey, "Edge-adaptive Transforms for Efficient Depth Map Coding," in *Picture Coding Symposium, 2010 (PCS 2010)*, Dec 2010, pp. 566–569.

[58] L. Lucas, N. Rodrigues, C. Pagliari, E. da Silva, and S. de Faria, "Efficient Depth Map Coding Using Linear Residue Approximation and a Flexible Prediction Framework," in *IEEE International Conference on Image Processing, 2012 (ICIP 2012)*, Sept 2012, pp. 1305–1308.

[59] I. Daribo, C. Tillier, and B. Pesquet-Popescu, "Motion Vector Sharing and Bitrate Allocation for 3D Video-plus-depth Coding," *EURASIP Journal on Applied Signal Processing - 3DTV: Capture, Transmission, and Display of 3D Video*, vol. 2009, pp. 1–13, Jan. 2008.

[60] G. Petrazzuoli, M. Cagnazzo, F. Dufaux, and B. Pesquet-Popescu, "Wyner-Ziv Coding for Depth Maps in Multiview Video-plus-depth," in *IEEE International Conference on Image Processing, 2011 (ICIP 2011)*, 2011, pp. 1817–1820.

[61] Y. Li, H. Liu, X. Liu, S. Ma, D. Zhao, and W. Gao, "Multi-hypothesis Based Multi-view Distributed Video Coding," in *Picture Coding Symposium, 2009 (PCS 2009)*, May 2009, pp. 1–4.

[62] G. Petrazzuoli, M. Cagnazzo, and B. Pesquet-Popescu, "Novel Solutions for Side Information Generation and Fusion in Multiview DVC," *EURASIP Journal on Advances in Signal Processing*, vol. 2013, no. 1, p. 154, 2013.

[63] J. Skorupa, J. Slowack, S. Mys, N. Deligiannis, J. De Cock, P. Lambert, C. Grecos, A. Munteanu, and R. Van De Walle, "Efficient Low-Delay Distributed Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 4, pp. 530–544, April 2012.

[64] G. Bjøntegaard, "Calculation of Average PSNR Differences between RD-Curves," Apr. VCEG-M33, Apr. 2001.

[65] X. Huang, C. Brites, J. Ascenso, F. Pereira, and S. Forchhammer, "Distributed Video Coding with Multiple Side Information," in *Picture Coding Symposium, 2009 (PCS 2009)*, 2009, pp. 385–388.

[66] W.-S. Kim, S. Narang, and A. Ortega, "Graph based Transforms for Depth Video Coding," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2012 (ICASSP 2012)*, Mar. 2012, pp. 813–816.

[67] G. Cheung, A. Kubota, and A. Ortega, "Sparse Representation of Depth Maps for Efficient Transform Coding," in *Picture Coding Symposium, 2010 (PCS 2010)*, Dec. 2010, pp. 298–301.

[68] S. Shimizu, H. Kimata, S. Sugimoto, and N. Matsuura, "Block-adaptive Palette-based Prediction for Depth Map Coding," in *IEEE International Conference on Image Processing, 2011 (ICIP 2011)*, Sep. 2011, pp. 117–120.

[69] S. Milani, P. Zanuttigh, M. Zamarin, and S. Forchhammer, "Efficient Depth Map Compression Exploiting Segmented Color Data," in *IEEE International Conference on Multimedia and Expo, 2011 (ICME 2011)*, July 2011, pp. 1–6.

[70] Y. Nie and K.-K. Ma, "Adaptive Rood Pattern Search for Fast Block-matching Motion Estimation," *IEEE Transactions on Image Processing*, vol. 11, no. 12, pp. 1442–1449, 2002.

[71] A. Wedel, T. Pock, J. Braun, U. Franke, and D. Cremers, "Duality TV-L^1 Flow with Fundamental Matrix Prior," in *International Conference Image and Vision Computing New Zealand, 2008 (IVCNZ 2008)*, Auckland, New Zealand, November 2008, pp. 1–6.

[72] *Extension of existing 3DV test set toward synthetic 3D video content*, Std., ISO/IEC JTC1/SC29/WG11, Doc. M19221, Daegu, Korea, January 2011.

[73] M. Ali and M. Kuijper, "Source Coding with Side Information Using List Decoding," in *IEEE International Symposium on Information Theory, 2010 (ISIT 2010)*, 2010, pp. 91–95.

[74] V. Stankovic, A. Liveris, Z. Xiong, and C. Georghiades, "On Code Design for the Slepian-Wolf Problem and Lossless Multiterminal Networks," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1495–1507, April 2006.