Distributed Wireless Visual Communication With Power Distortion Optimization

Xiaopeng Fan, *Member, IEEE,* Feng Wu, *Fellow, IEEE,* Debin Zhao, and Oscar C. Au, *Fellow, IEEE*

***Abstract*—This paper proposes a novel framework called DCast for distributed video coding and transmission over wireless networks, which is different from existing distributed schemes in three aspects. First, coset quantized DCT coefficients and motion data are directly delivered to the channel coding layer without syndrome or entropy coding. Second, transmission power is directly allocated to coset data and motion data according to their distributions and magnitudes without forward error correction. Third, these data are transformed by Hadamard and then directly mapped using a dense constellation (64K-QAM) for transmission without Gray coding. One of the most important properties in this framework is that the coding and transmission rate is fixed and distortion is minimized by allocating the transmission power. Thus, we further propose a power distortion optimization algorithm to replace the traditional rate distortion optimization. This framework avoids the annoying cliff effect caused by the mismatch between transmission rate and channel condition. In multicast, each user can get approximately the best quality matching its channel condition. Our experiment results show that the proposed DCast outperforms the typical solution using H.264 over 802.11 up to 8 dB in video PSNR in video broadcast. Even in video unicast, the proposed DCast is still comparable to the typical solution.**

***Index Terms*—Distributed video coding (DVC), softcast, wireless visual communication.**

I. Introduction

DISTRIBUTED video coding (DVC) [1]–[4] is an attractive scheme for video compression that has emerged in the past decade. Different from conventional video coding schemes, it utilizes cross-frame correlation only at the decoder. This has several unique advantages. First, DVC can shift intensive computation from encoder to decoder, which is appealing for low complexity video encoding applications. Second, DVC framework is robust to transmission errors, which is desirable for wireless applications. Although it has been proven that the theoretical coding performance should be equivalent, no matter what source correlation is utilized at encoder or decoder for some typical sources [5], [6], the actual coding performance of DVC is still far inferior to that of the conventional H.264 standard [7].

In DVC, quantized transform coefficients are converted to bit planes and compressed to bits by syndrome or entropy coding [2], [4], [8]. The syndrome coding is implemented via channel codes (e.g., low-density parity-check codes). These channel codes are also typically applied for error protection in the physical (PHY) layer. Therefore, Xiong *et al.* [9], [10] propose a 46 joint source-channel coding (JSCC) framework for distributed 47 video transmission based on their previous work on JSCC 48 of binary source. Except for these JSCC works, the transmission of distributed coded video is still similar to that of conventional coded video.

Recently, a joint video coding and transmission scheme, named Softcast [11], [12], has been proposed for wireless video multicasting. The key idea in Softcast is that transform coefficients are not compressed by entropy coding. Instead, they are directly transmitted through a dense constellation after allocating a certain power, such that the received data can be decoded at any channel conditions. The decoded data is not error free and its signal-to-noise ratio (SNR) is dependent on channel condition for a given transmission power. Although the video coding layer of Softcast is simply done through 2-D or 3-D transformation, the overall performance of Softcast still outperforms the typical solution using H.264 over 802.11 in video multicast.

The current Softcast only adopts 3-D DCT to exploit the cross-frame correlation. Researches in scalable video coding has fully demonstrated that this is inefficient due to the lack of motion alignment among frames [13]–[15]. However, motion compensation (MC) in H.264 is difficult to adopt in Softcast because in Softcast the reconstructed frames are determined by channel noise and the encoder can hardly obtain the same reconstructed frames as the decoder. Thus this paper proposes a novel framework called DCast, which not only utilizes the cross-frame correlation by motion alignment but also retains the nice properties provided by Softcast.

In the proposed DCast, transformed coefficients are first coset quantized and then are transmitted as Softcast. Similar to other DVC frameworks, DCast utilizes the cross-frame correlation at the decoder. The proposed DCast has two different approaches to process motion vectors (MVs). Like most

Manuscript received June 4, 2012; revised October 18, 2012 and December 15, 2012; accepted January 26, 2013. Date of publication February 25, 2013; date of current version May 31, 2013. This work was supported in part by the Major State Basic Research Development Program of China (973 Program 2009CB320905), the Program for New Century Excellent Talents in University (NCET) of China (NCET-11-0797), the National Science Foundation of China under Grants 61100095, and the Fundamental Research Funds for the Central Universities under Grant HIT.BRETIII.201221. This paper was recommended by Associate Editor E. Magli.

X. Fan and D. Zhao are with the Department of Computer Science, Harbin Institute of Technology, Harbin 150001, China (e-mail: fxp@hit.edu.cn).

F. Wu is with Microsoft Research Asia, Beijing 100080, China.

O. C. Au is with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Kowloon, Hong Kong.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2013.2249019

Fig. 1. Compression of X when its side information S is available at the decoder.

traditional DVC schemes, in the first approach motion vectors are estimated at decoder. It does not need reference frames at encoder, and greatly reduces the encoding complexity. But the side information may not be accurate, thus leading to low coding efficiency. In the second approach, motion vectors are estimated at encoder and then transmitted to the decoder. Actually several other DVC schemes also propose to estimate motion vectors at encoder and transmit them to decoder for improving the quality of side information [16], [17]. The initial results of these two approaches have been reported in [18] and [19]. In this paper we will focus our study on the second approach but both of them will be evaluated.

The key technical contribution in this paper is the proposed power distortion optimization. In the proposed DCast, each pair of quantized DCT coefficients or transformed motion vectors are transmitted in one time slot and thus the transmission rate is fixed. The distortion is minimized by optimally allocating transmission power. This paper evaluates the impact of the channel noise on the distortion of the motion vectors and then the impact of this distortion on the distortion of reconstructed video via the power spectrum approach [20]. Furthermore, a joint power optimization among coefficients and motion data is derived. Our experimental results show that the proposed DCast can outperform Softcast up to 2 dB in video PSNR as it can better utilize the cross-frame correlation. Compared with the typical solution using H.264 over 802.11, the proposed DCast can gain up to 8 dB in video PSNR in multicast. Even in unicast, it is still comparable to the typical solution of H.264 over 802.11.

The rest of this paper is organized as follows. Section II briefly reviews the related work on distributed video coding and transmission. Section III introduces the proposed DCast including both encoder and decoder. Section IV discusses the proposed power distortion optimization. Section V presents our experimental results and compares them with Softcast and H.264 over 802.11. Finally, Section VI concludes this paper.

II. Related Works

A. Distributed Video Coding

To compress a source with its prediction that is only available at the decoder is a typical problem in distributed source coding (DSC). As shown in Fig. 1, X is the source to be compressed (possibly representing the source video), and S is its side information (possibly representing the predicted frame). The theoretical foundations of DSC, the Slepian-Wolf theorem [5], and the Wyner-Ziv theorem [6] show that the source X can be efficiently compressed with its predictor S available only at the decoder. In practice, efficient DSC can be achieved by coset coding, turbo coding and LDPC coding [21], [22].

Accompanied by advances of practical DSC solutions, DVC has emerged since a decade. Puri *et al.* [3], [4] propose a DVC framework called PRISM, which implements DVC by coset coding and supports motion estimation (ME) at decoder. The main attributes of PRISM include the increased robustness to channel losses and more flexible sharing of computational complexity between encoder and decoder. Another DVC work is the low complexity framework proposed by in [1] and [2]. In this framework, the DVC is implemented by turbo code, while the motion estimation at decoder is based on motion compensated interpolation (MCI) and motion compensated extrapolation (MCE).

Although DVC has shown unique advantages in visual communication, its compression efficiency is much lower than conventional framework [2], [23]. In recent years, much research has focused on improving the performance of DVC. Enabling transform coding [24], [25] and intra/inter mode selection [26]–[28] allows DVC to exploit not only inter but also intra frame redundancy. Hash based DVC lets the encoder send hash code to the decoder to improve the accuracy of ME and the side information quality [29]. Successive refinement schemes [30]–[33] perform ME and DVC decoding alternatively and recursively, such that the MVs and reconstruction frame are successively refined during decoding process. More accurate correlation estimation in DVC improves the utilization of the side information [34]–[37].

Different from these DVC schemes, the proposed DCast directly delivers coset quantized coefficients and motion vectors to the channel coding layer. Furthermore, when coefficients and motion vectors are transmitted from encoder to decoder, they are allowed to be corrupted by channel noise. It is clear from our results that DVC is robust to noise embedded in the received data.

B. Distributed Video Transmission Over Wireless Network

The transmission of distributed coded video is usually similar to the transmission of conventional coded video in the PHY layer of wireless network. Coded binary data is first protected by channel coding and then is mapped to a constellation for transmission. When syndrome coding is adopted, DVC coding and channel coding can be jointly optimized. Xu *et al.* [9] made the first attempt to study DVC from a JSCC. It is a layered coding scheme, where the enhancement layer uses Raptor code for both video compression and data protection. In another frame-based JSCC scheme [16], the functionality of both DVC and channel coding are implemented universally by one error correction code.

In these JSCC schemes, distributed video transmission are actually processed as data transmission. The transmission error are desired to be corrected in the JSCC decoder. Thus many bits are paid in channel coding to correct transmission errors. In the proposed DCast, quantized coefficients and transformed motion vectors are directly transmitted after allocating a certain power. Although the received data after decoding may still contain a certain channel noise, it is more efficient on power consumption because some received noise can be tolerated by DVC.

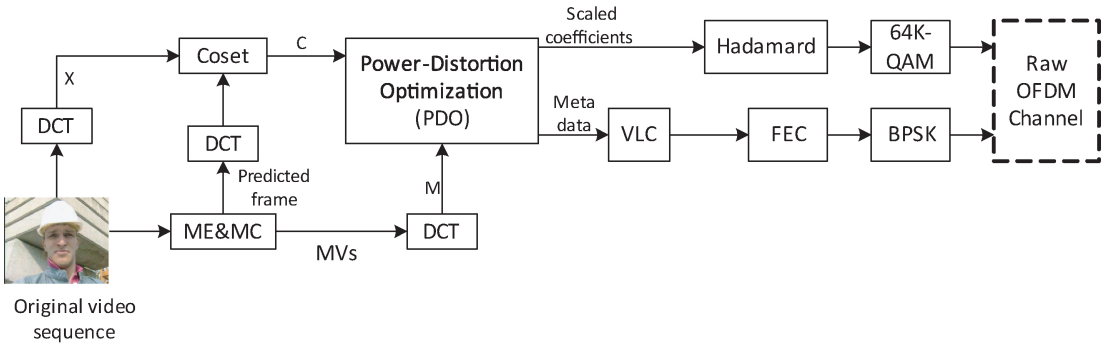


Fig. 2. DCast server (for inter frames).

C. Softcast

Softcast is a simple but comprehensive design for wireless video multicast, covering the functionality of video compression, data protection and transmission in one scheme [11]. The Softcast encoder consists of the following steps: 1) DCT transform, power allocation, 2) Hadamard transform, and 3) direct dense modulation. Transform removes the spatial redundancy of a video frame. Power allocation minimizes the total distortion by optimally scaling the transform coefficients. Hadamard transform is in some sense a precoding to make packets with equal power and equal importance. After that, the data is directly mapped into the wireless symbols by a very dense QAM. The decoder uses linear least square estimator (LLSE) to reconstruct the signal. Almost all the steps in Softcast are linear operations and thus the channel noise is directly transformed into reconstruction noise of the video. Therefore, Softcast is asymptotically robust in the sense that each user can get the visual quality matching his channel condition.

However, Softcast exploits the intra-frame correlation only and thus is not very efficient in the aspects of video compression. Recently, Aditya *et al.* [38] proposed another video coding and transmission scheme called Flexcast. It removes entropy coding from conventional video coding and adopts rateless channel coding for channel variation. Thus, it has the better coding efficiency. However, Flexcast is a unicast approach and can hardly multicast or broadcast video to the users with different SNRs simultaneously because of motion compensation. In a recent improved version of Softcast, the utilization of 3-D-DCT partially enables inter frame compression [12]. However, without motion alignment the inter frame correlation is still not fully exploited yet.

The proposed DCast not only fully utilizes the cross-frame correlation but also retains the good properties of Softcast. DCast enables inter frame coding by DVC rather than conventional motion compensation. Instead of transmitting a video frame itself like Softcast, DCast transmits the coset codes of the video frame such that the frame can be reconstructed by utilizing the prediction frame as side information at the decoder. This saves the transmission power (or equivalently increases the SNR) because the coset data typically have much smaller magnitude than the original data. Recently, we also noticed that Kochman *et al.* [39] have studied the utilization of coset coding in the Wyner-Ziv Dirty-Paper problem and proved its optimality and asymptotical robustness in multicast applications. It can be considered in general as the theoretical foundation to support the proposed DCast.

III. Proposed DCast Framework

DCast divides input video sequences into groups of pictures (GOP). In each GOP, the first frame is an intra (coded) frame, while the following frames are inter frames. The compression and transmission of the intra frame in DCast is the same as in Softcast, which consists of DCT, power allocation and Hadamard transform. In the rest of this paper, we will focus on the compression and transmission of inter frames. For simplicity, we mainly discuss the case with motion vectors estimated at encoder.

Fig. 2 depicts the server side of DCast. DCast first transforms the current frame into DCT domain. Meanwhile, DCast performs ME and MC on the original video sequence to get the encoder predictions and MVs. Then DCast applies coset coding on the transform coefficients of the original image to get the coset data for each DCT coefficient. The quantization step size of the coset coding is determined at the encoder according to the estimated decoder prediction noise. The MVs of the current frame, in the form of a matrix, are also transformed by DCT. The coset data and the motion data are then scaled for power distortion optimization (PDO).

The scaling factors and other metadata are transmitted by using a conventional scheme consisting of variable length coding (VLC), forward error correction (FEC), and BPSK modulation. The scaled coefficients are transformed by Hadamard as precoding to make packets with equal power and equal importance. After that, the resulting coefficients are mapped to complex symbols directly by a very dense constellation (64K-QAM). Each coefficient is quantized into 8-bit integer number and every two integers compose one complex number of 64K possible values. At last, these complex numbers are passed into a raw OFDM module undergoing iFFT and D/A conversion for transmission.

The receiver side of DCast is depicted in Fig. 3. The raw OFDM module performs A/D conversion and FFT to reconstruct modulated data including both the scaled coefficients and the metadata. The metadata is demodulated and decoded first. Then the scaled coefficients are reconstructed by inverse 64K-QAM and inverse Hadamard transform. The inverse

64K-QAM here does nothing but splitting each complex value back into two real values. Each real value here is actually the 8-bit integer number plus channel noise.

After inverse Hadamard transform, linear minimum mean square error (LMMSE) estimation of the residue coefficients and the MV coefficients are performed. Then the MVs are transformed back to spatial domain by inverse DCT. After this, the MC module generates the predicted frame by the MVs and the reference frame. The predicted frame is transformed into frequency domain by DCT. Then with the coset residues and the predictors, the coset decoding module recovers the DCT coefficients of the current frame. At last, the signals are transformed back to spatial domain, and are linearly combined with the predicted signals by LMMSE to generate the final reconstruction.

A. Coset Coding

Coset coding is a typical technique used in DSC. It partitions the set of possible input source values into several cosets and transmits the coset index to the decoder. With the coset index and the predictor, the decoder can recover the source value by choosing the one in the coset closest to the predictor. Coset coding achieves compression because the coset index has typically lower entropy than the source value.

Let X be the DCT coefficients of the original video frame. DCast encodes X to get coset values C. DCast divides the coefficients into 64 subbands according to the frequency. Let X_i be the ith subband of X, and C_i be the ith subband of C. For each i, DCast quantizes the ith subband of X by a uniform scaler quantizer $\mathsf{Q_i}(\cdot)$ and gets the residue value [39] by

$$C_i = X_i - \mathsf{Q_i}(X_i) = X_i - \lfloor \frac{X_i}{q_i} + \frac{1}{2} \rfloor q_i \tag{1}$$

This coset coding is actually throwing away the main part of X. In some sense C represents the detail of X.

At the client side, with the side information S (i.e. the predicted DCT coefficients) and the received coset value $\hat{C}$, the receiver reconstructs the DCT coefficients by coset decoding. Let S_i be the ith subband of S, and $\hat{C}_i$ be the ith subband of $\hat{C}$. Since S_i is close to X_i, $S_i - \hat{C}_i$ is around $X_i - C_i$. Thus $S_i - \hat{C}_i$ is around $\mathsf{Q_i}(X_i)$ from (1). The quantizers are carefully designed such that applying quantization $\mathsf{Q_i}(\cdot)$ on $S_i - \hat{C}_i$ we could get $\mathsf{Q_i}(X_i)$, i.e.

$$\mathsf{Q_i}(X_i) = \mathsf{Q_i}(S_i - \hat{C}_i) \tag{2}$$

in high probability. Therefore, each subband of coefficients is decoded by

$$\hat{X}_i = \mathsf{Q_i}(S_i - \hat{C}_i) + \hat{C}_i \tag{3}$$

where $\hat{X}$ is the reconstruction of X, and each $\hat{X}_i$ is the ith subband of $\hat{X}$. When the coset decoding is successful, i.e. $\mathsf{Q_i}(X_i) = \mathsf{Q_i}(S_i - \hat{C}_i)$, the reconstruction noise is

$$\hat{X}_i - X_i = \hat{C}_i - C_i. \tag{4}$$

B. Estimation of Coset Quantization Step

The value of each coset step q_i is crucial to the coding performance of DCast. If q_i is too small, the coset decoding may suffer failure. On the other hand, if q_i is too large, the coset value C_i in (1) will be large and will consume a lot of transmission power to keep the distortion small. The value of each q_i is determined as follows. Injecting (1) into (2), we get

$$\begin{aligned} \mathsf{Q_i}(X_i) &= \mathsf{Q_i}(S_i - \hat{C}_i + C_i - X_i + \mathsf{Q_i}(X_i)), & (5)\\ &= \mathsf{Q_i}(X_i) + \mathsf{Q_i}(S_i - \hat{C}_i + C_i - X_i) & (6) \end{aligned}$$

To guarantee successful coset decoding, the last item should be 0. This means the quantization step q_i should satisfy

$$\frac{q_i}{2} \geq |S_i - X_i + C_i - \hat{C}_i| \tag{7}$$

In this equation, the $S_i - X_i$ is the prediction noise at the decoder and the $C_i - \hat{C}_i$ is the reconstruction noise of the coset value C_i due to transmission. In this paper, we assume they are independent Gaussian source. We let each q_i to be $2n$ times of the standard deviation of $S_i - X_i + C_i - \hat{C}_i$, i.e.

$$q_i^2 = 4n^2 \sigma^2_{S_i - X_i + C_i - \hat{C}_i} \tag{8}$$

and this guarantees that condition (7) is satisfied in probability

$$\mathsf{Pr} = \mathsf{erf}(n/\sqrt{2}) \tag{9}$$

Under the same assumption, the variance of $S_i - X_i + C_i - \hat{C}_i$ is the summation of the variance of $S_i - X_i$ and $C_i - \hat{C}_i$, i.e.

$$\sigma^2_{S_i - X_i + C_i - \hat{C}_i} = \sigma^2_{S_i - X_i} + \sigma^2_{C_i - \hat{C}_i} \tag{10}$$

and each q_i can be calculated by

$$q_i^2 = 4n^2(\sigma^2_{S_i - X_i} + \sigma^2_{C_i - \hat{C}_i}) \tag{11}$$

In our implementation, we let $n = 3$ such that the coset decoding is successful for more than 99.7% coefficients. In (11), $\sigma^2_{S_i - X_i}$ is the variance of the hypothetic residue between the source and the side information, and it is estimated by simulating at the encoder a receiver with target channel SNR. $\sigma^2_{C_i - \hat{C}_i}$ is the distortion of coset value C_i due to transmission. It is also the distortion of the source X_i according to (4). $\sigma^2_{C_i - \hat{C}_i}$ is related to both the residue $\sigma^2_{S_i - X_i}$ and the channel SNR. The explicit expression of $\sigma^2_{C_i - \hat{C}_i}$ is given in Section IV.

C. Power Allocation

DCast transmits both the coset values and the motion information. Thus, it has two levels of power allocation. The first allocation is between MV data and coset data. The second level is the allocation within MV coefficients or coset coefficients. The optimal power allocation between MV data and coset data is given in Section IV. The optimal power allocation within coset coefficients and the optimal power allocation within MV coefficients are as follows.

Let P_{coset} be the total power of coset data, and g_{C_i} be the gain (scaling factor) of C_i. The problem is how to minimize the reconstruction distortion of X, by optimally allocating power among C_i. Under the assumption that the coset decoding is successful in high probability, the reconstruction distortion of X will be equal to the reconstruction distortion of C according

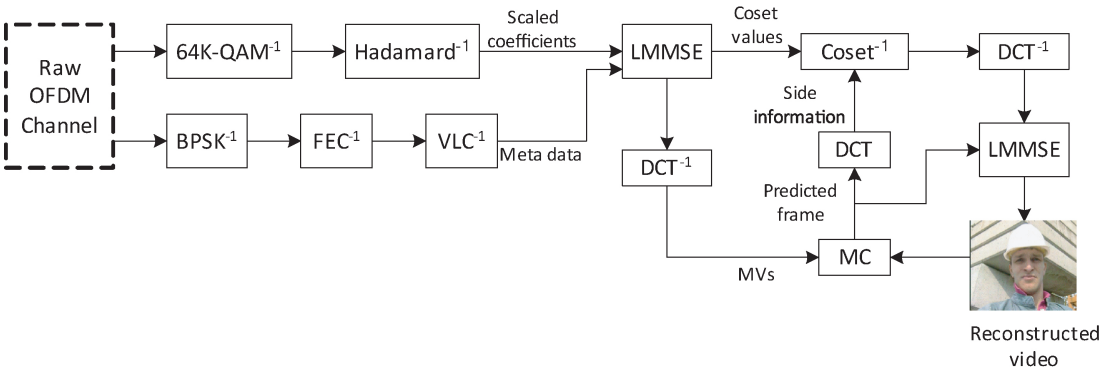


Fig. 3. DCast receiver (for inter frames).

to (4). This means that the problem now becomes how to minimize the reconstruction distortion of C, by optimally allocating power among C_i. Thus the solution has a similar form as the one in Softcast [12], i.e.

$$\tilde{C}_i = g_{C_i} C_i, \quad g_{C_i} = \left(\frac{P_{coset}}{\sigma_{C_i} \sum_j \sigma_{C_j}} \right)^{1/2} \tag{12}$$

where $\tilde{C}$ is the coset value after power allocation, $\tilde{C}_i$ is the ith subband of $\tilde{C}$, and σ_{C_i} is the standard deviation of C_i. This power allocation tends to scale down large coefficients to get better performance under the constrained total power. The encoder calculates the variance $\sigma^2_{C_i}$ for each subband and transmits it to the decoder. With $\sigma^2_{C_i}$, both the encoder and the decoder calculate the gain g_{C_i} for each C_i by (12).

On MV data, DCast also performs power allocation. To apply power allocation, the encoder performs 2-D DCT on the MVs (the whole MV field) and gets transform coefficients M. Note that each MV contains horizontal and vertical components, and the transform is actually applied to both components separately. Each coefficient M_i is then considered as a subband. The encoder applies a similar optimal power allocation over M, i.e.

$$\tilde{M}_i = g_{M_i} M_i, \quad g_{M_i} = \left(\frac{P_{mv}}{\sigma_{M_i} \sum_j \sigma_{M_j}} \right)^{1/2} \tag{13}$$

where $\tilde{M}$ is the MV data after power allocation, $\tilde{M}_i$ is the ith subband of $\tilde{M}$, σ_{M_i} is the standard deviation of M_i, and P_{mv} is the total power for motion data. Since each subband of M contains only one coefficient, it is not efficient to transmit the variance of each subband. In this light, DCast only transmits the average variance $\sigma^2_M = \frac{1}{n} \sum_i \sigma^2_{M_i}$ where n is the number of subbands. As shown in our previous work [19], the $\sigma^2_{M_i}$ and g_{M_i} are calculated by using σ^2_M. Under the assumption that the motion field is random Markov field where the correlation coefficient between two neighboring MVs is ρ, each $\sigma^2_{M_i}$ can be calculated by

$$\sigma^2_{M_i} = \sigma^2_M V_{M_i} \tag{14}$$

where V_{M_i} is the ith element of matrix V_M, and

$$V_M = \mathsf{diag}(2D\_DCT(R^{(h)}))diag(2D\_DCT(R^{(w)}))^T \tag{15}$$

is a constant matrix for given ρ. Here the function $\mathsf{diag}(\cdot)$ produces the diagonal elements of the input matrix in the form of a column vector. $\mathsf{2D\_DCT}(\cdot)$ means 2-D DCT transform. w and h are the width and height of the motion field respectively and

$$R^{(k)} = \begin{bmatrix} 1 & \rho & \cdots & \rho^{k-1} \\ \rho & 1 & \cdots & \rho^{k-2} \\ \vdots & \vdots & \ddots & \vdots \\ \rho^{k-1} & \rho^{k-2} & \cdots & 1 \end{bmatrix}. \tag{16}$$

The value of σ^2_M is calculated at the encoder and is transmitted to the decoder as mentioned in the previous section. Both the encoder and the decoder calculate the value of each $\sigma^2_{M_i}$ by (14)-(16). In our experiments we let $\rho = 0.7$ according to statistics over several different video sequences. With each $\sigma^2_{M_i}$, the optimal power allocation gain g_{M_i} for each subband is calculated at both encoder and decoder by (13). The decoder needs the value of g_{M_i} in (13) to reconstruct the signal.

D. Packaging and Transmission

Similar to Softcast [12], DCast transmits not only a small amount of binary symbols but mainly real-valued symbols. The organization of the symbol stream is as follows. The symbol stream consists of a header and a following data stream.

$$symbol\_stream = \{header\_bitstream, data\_stream\} \tag{17}$$

The header bitstream contains the meta data, including the information of coset variances $\sigma^2_{C_i}$, the quantization steps q_i, the average MV variance σ^2_M and other useful parameters.

$$\begin{aligned} header\_bitstream \leftarrow \quad & \{coset\ variances, \\ & quantization\ steps, \\ & average\ MV\ variance, \\ & parameters\} \end{aligned} \tag{18}$$

The header information is coded in a conventional way. The encoder applies 8-bits scalar quantization on σ_{C_i}, q_i and σ_M respectively. Then the quantization results are compressed by variable length coding (VLC). The VLC is the universal one used for coding motion vectors in H.264 [7]. The compressed header bitstream is transmitted by the standard 802.11 PHY

layer at the lowest speed, i.e., by using a 1/2 convolutional code and BPSK modulation. This is to make sure the header bits are decoded correctly when channel SNR is in typical working range (5–25 dB) of 802.11. Note that the size of the header is very small compared to the whole data of one frame. According to our experiments, the proportion of the bandwidth required by headers is less than 3%.

The data stream contains the information of the coset data $\tilde{C}$ and the MV data $\tilde{M}$. Similar to Softcast [12], DCast applies Hadamard transform on the coset data $\tilde{C}$ and the MV data $\tilde{M}$ to create packets with equal energy. Coset data and MV data are mixed together and then every 64 numbers are grouped for Hadamard transform. This forms the data stream

$$data\_stream \overset{H}{\leftarrow} \{coset\ data,\ MV\ data\}. \tag{19}$$

Note that the data stream consists of real values rather than binary values. In PHY layer, these real values are mapped to complex symbols directly by 64K-QAM constellation [12]. This constellation is a typical N-QAM constellation with N equal to 65536 (256 by 256). Each input real value is quantized into an 8-bit integer number by uniform scalar quantizer. The dynamic range of the quantizer is formed by the minimal and maximal input value. It is calculated for each frame at encoder and sent to decoder as a parameter in (18). After this quantization, every two integers compose one complex number as the output of the 64K-QAM constellation. An inverse FFT is computed on each packet of symbols, giving a set of complex time-domain samples. These samples are then quadrature-mixed to passband in the standard way. The real and imaginary components are first converted into the analogue domain using D/A converters. The analogue signals are then used to modulate cosine and sine waves at the carrier frequency, respectively. These signals are then summed to give the transmission signal.

In DCast, both MV data and coset data are transmitted by the aforementioned direct source channel mapping. This makes the system adaptive to the fluctuation of the channel SNR. Given a transmitter, high SNR users would receive accurate MVs and coset values and reconstruct high quality video. Meanwhile, low SNR users would receive noisy MVs and coset values, and derive noisy prediction frame based on the noisy MVs. However, the coset decoding in DCast has good tolerance to the noise of the prediction. Thus, the low SNR users would still reconstruct the video.

E. LMMSE at Decoder

The proposed approach contains two LMMSE estimators, operating in transform domain and spatial domain, respectively.

The purpose of the first LMMSE estimator is to reconstruct the coset data C and the MV data M in transform domain with minimum distortion. Let Y be the received signal after inverse Hadamard transform. Y contains the noisy version of the coset data and the MV data. Y can be written as:

$$Y = \begin{bmatrix} \dot{C} \\ \dot{M} \end{bmatrix} \tag{20}$$

where $\dot{C}$ is the noisy version of coset data, $\dot{M}$ is the noisy version of MV data. Let $W^{(C)}$ and $W^{(M)}$ be the channel noise in $\dot{C}$ and $\dot{M}$ respectively. Let $\dot{C}_i$, $\dot{M}_i$, $W_i^{(C)}$ and $W_i^{(M)}$ be the ith subband of $\dot{C}$, $\dot{M}$, $W^{(C)}$ and $W^{(M)}$, respectively. We model each element in $W^{(C)}$ and $W^{(M)}$ as i.i.d Gaussian source with variance N_0. Each subband of $\dot{C}$ and $\dot{M}$ can be expressed as

$$\dot{C}_i = g_{C_i} C_i + W_i^{(C)}, \qquad \dot{M}_i = g_{M_i} M_i + W_i^{(M)}. \tag{21}$$

Therefore, the LMMSE reconstruction of the original signals is

$$\hat{C}_i = \frac{\sigma_{C_i}^2}{\sigma_{C_i}^2 g_{C_i}^2 + N_0} \dot{C}_i, \qquad \hat{M}_i = \frac{\sigma_{M_i}^2}{\sigma_{M_i}^2 g_{M_i}^2 + N_0} \dot{M}_i. \tag{22}$$

And the reconstruction distortion of each subband is

$$\mathbb{E}\{(\hat{C}_i - C_i)^2\} = \frac{\sigma_{C_i}^2 N_0}{\sigma_{C_i}^2 g_{C_i}^2 + N_0}, \tag{23}$$

$$\mathbb{E}\{(\hat{M}_i - M_i)^2\} = \frac{\sigma_{M_i}^2 N_0}{\sigma_{M_i}^2 g_{M_i}^2 + N_0}. \tag{24}$$

The purpose of the second LMMSE estimator is to reconstruct each pixel x in spatial domain with minimum distortion. DCast decoder applies inverse DCT transform on coset reconstruction $\hat{X}$ and gets a pixel-domain preliminary reconstruction $\hat{x}$. $\hat{x}$ is considered as the first noisy version of x. DCast also has the predicted pixel s as the second noisy version of x. With $\hat{x}$ and s, the optimal LMMSE estimation x^* is given by:

$$x^* = \theta s + (1 - \theta)\hat{x} \tag{25}$$

where

$$\theta = \frac{\sigma_{\hat{x}-x}^2}{\sigma_{s-x}^2 + \sigma_{\hat{x}-x}^2}. \tag{26}$$

$\sigma_{\hat{x}-x}^2$ is the variance of $\hat{x} - x$, and σ_{s-x}^2 is the variance of $s - x$. In DCast, the prediction noise variance σ_{s-x}^2 is estimated at block level. Since $\hat{x}$ is close to x, σ_{s-x}^2 is estimated by calculating $\mathbb{E}\{(s - \hat{x})^2\}$. The variance $\sigma_{\hat{x}-x}^2$ is calculated as follows. According to the Parseval's theorem and (4), we have

$$\sigma_{\hat{x}-x}^2 = \mathbb{E}\{(\hat{x} - x)^2\} = \mathbb{E}\{(\hat{X} - X)^2\} = \mathbb{E}\{(\hat{C} - C)^2\} \tag{27}$$

where $\mathbb{E}\{(\hat{C} - C)^2\}$ is directly calculated by summation on (23).

IV. Power-Distortion Optimization

In DCast, both the MVs and the coset values require power to transmit. Thus it is necessary to investigate the optimal power allocation between MVs and the coset values. Let D be the reconstruction distortion, and P be the transmission power. P_{coset} and P_{mv} be the transmission power for the coset values and the MVs, respectively. The optimal power allocation is the one minimizing the reconstruction distortion D for given power P, i.e., the optimization problem is

$$\begin{aligned} \min \quad & D, \\ s.t. \quad & P_{mv} + P_{coset} \le P. \end{aligned} \tag{28}$$

A. Relationship Between Variables

The distortion D is directly related to both the decoder prediction noise variance σ^2_{S-X}, and the coset transmission power P_{coset}. Intuitively, using larger transmission power P_{coset} decreases the variance of the coset error $\hat{C}-C$ at decoder. This means smaller D since the reconstruction error $\hat{X}-X$ equals to the coset error $\hat{C}-C$ according to (4). Meanwhile, larger σ^2_{S-X} means lower quality of side information (SI), and lower quality SI leads to larger reconstruction distortion. Therefore, the distortion D should be a decreasing function of the coset power P_{coset} and an increasing function of the prediction noise variance σ^2_{S-X}.

Furthermore, the prediction noise variance σ^2_{S-X} is related to the MV transmission power P_{mv}. We use a two dimensional random vector $\mathbf{\Delta} \sim \mathcal{N}(0, \sigma^2_{\mathbf{\Delta}}\mathbf{I}_{2\times 2})$ to model MV error, while $\sigma^2_{\mathbf{\Delta}} = \frac{1}{2}\mathbb{E}\{\mathbf{\Delta}^T\mathbf{\Delta}\}$ is the distortion of MV. Using larger transmission power P_{mv} decreases the MV distortion $\sigma^2_{\mathbf{\Delta}}$ and this means more accurate MVs. More accurate MVs produces higher quality of decoder SI S, and hence a smaller prediction noise variance σ^2_{S-X}. Thus the prediction noise variance σ^2_{S-X} decreases in the MV transmission power P_{mv}. However, due to the power constraint, giving more power to coset (i.e. using larger P_{coset}) means less power to MV (i.e., using smaller P_{mv}), and vice versa. This is why we need power distortion optimization. In the following part of this section, before solving (20) we will derive the relationship between:

1) MV transmission power P_{mv} and MV distortion $\sigma^2_{\mathbf{\Delta}}$;
2) MV distortion $\sigma^2_{\mathbf{\Delta}}$ and prediction noise variance σ^2_{S-X};
3) Distortion D, coset power P_{coset} and prediction noise variance σ^2_{S-X}.

B. MV transmission Power P_{mv} and MV Distortion $\sigma^2_{\mathbf{\Delta}}$

This subsection focuses on the relationship between MV transmission power P_{mv} and MV distortion $\sigma^2_{\mathbf{\Delta}}$. According to Parseval's theorem, the MV distortion $\sigma^2_{\mathbf{\Delta}}$ in spatial domain equals to the MV distortion in DCT domain, i.e.

$$\sigma^2_{\mathbf{\Delta}} = \frac{1}{n_{mv}}\sum_i \mathbb{E}\{(\hat{M}_i - M_i)^2\} \tag{29}$$

where n_{mv} is the number of MV coefficients. From (23), we get

$$\sigma^2_{\mathbf{\Delta}} = \frac{1}{n_{mv}}\sum_i \frac{\sigma^2_{M_i}N_0}{\sigma^2_{M_i}g^2_{M_i} + N_0} \approx \frac{1}{n_{mv}}\sum_i \frac{N_0}{g^2_{M_i}} \tag{30}$$

where the approximation is accurate when $P_{mv} \gg N_0$. Substituting (13) into (30), we get

$$\sigma^2_{\mathbf{\Delta}} \approx \frac{N_0(\sum_i \sigma_{M_i})^2}{n_{mv}P_{mv}}. \tag{31}$$

Then using (14) we get

$$\sigma^2_{\mathbf{\Delta}} \approx \frac{N_0\sigma^2_M(\sum_i V^{\frac{1}{2}}_{M_i})^2}{n_{mv}P_{mv}}. \tag{32}$$

By defining

$$\alpha_{mv} = (\frac{1}{n_{mv}}\sum_i V^{\frac{1}{2}}_{M_i})^2 \tag{33}$$

we can rewrite (32) as

$$\sigma^2_{\mathbf{\Delta}} \approx \frac{n_{mv}N_0\sigma^2_M\alpha_{mv}}{P_{mv}} = \alpha_{mv}\sigma^2_M\left(\frac{P_{mv}}{n_{mv}N_0}\right)^{-1}. \tag{34}$$

In this equation, σ^2_M is the variance of the MV signal to transmit, $\frac{P_{mv}}{n_{mv}N_0}$ is the SNR for MV signal. Thus α_{mv} can be considered as the extra gain owning to the power allocation in (13). From this equation, the MV distortion $\sigma^2_{\mathbf{\Delta}}$ is proportional to the inverse of the MV transmission power P_{mv}.

C. MV Distortion $\sigma^2_{\mathbf{\Delta}}$ and Prediction Noise Variance σ^2_{S-X}

This subsection focuses on the relationship between MV distortion $\sigma^2_{\mathbf{\Delta}}$ and prediction noise variance σ^2_{S-X}. Let $\dot{S}$ be the original decoder prediction when the MVs are perfectly received. The practical decoder prediction noise $S-X$ consists of two components: the original prediction noise $\dot{S}-X$, and the additional prediction noise $S-\dot{S}$ caused by erroneous MVs. In this paper, we assume they are independent of each other, and therefore

$$\sigma^2_{S-X} = \sigma^2_{\dot{S}-X} + \sigma^2_{S-\dot{S}}. \tag{35}$$

Given that the $\dot{S}$ is a phase-shift version of S, $\sigma^2_{S-\dot{S}}$ can be analyzed by using power density. Similar to the derivation in [20], we have

$$\sigma^2_{S-\dot{S}} = \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} 2\Phi_{ss}(\boldsymbol{\omega})(1-\mathbb{E}\{cos(\boldsymbol{\omega}^T\mathbf{\Delta})\})d\boldsymbol{\omega} \tag{36}$$

where $\Phi_{ss}(\cdot)$ is the power density function of side information, $\boldsymbol{\omega}$ is two-dimensional frequency (in radians), and $\mathbf{\Delta} \sim \mathcal{N}(0, \sigma^2_{\mathbf{\Delta}}\mathbf{I}_{2\times 2})$ is the MV error. For small $\sigma^2_{\mathbf{\Delta}}$, we have

$$1-\mathbb{E}\{cos(\boldsymbol{\omega}^T\mathbf{\Delta})\} \approx \frac{1}{2}\mathbb{E}(\boldsymbol{\omega}^T\mathbf{\Delta})^2 = \frac{1}{2}\sigma^2_{\mathbf{\Delta}}\boldsymbol{\omega}^T\boldsymbol{\omega}, \tag{37}$$

and thus

$$\sigma^2_{S-\dot{S}} \approx \frac{1}{4\pi^2}\sigma^2_{\mathbf{\Delta}}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\Phi_{ss}(\boldsymbol{\omega})\boldsymbol{\omega}^T\boldsymbol{\omega}d\boldsymbol{\omega}. \tag{38}$$

We define

$$\gamma = \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\Phi_{ss}(\boldsymbol{\omega})\boldsymbol{\omega}^T\boldsymbol{\omega}d\boldsymbol{\omega} \tag{39}$$

and γ is a constant for a given video frame. Then we get

$$\sigma^2_{S-\dot{S}} \approx \gamma\sigma^2_{\mathbf{\Delta}}. \tag{40}$$

Substituting (40) into (35), we get

$$\sigma^2_{S-X} = \sigma^2_{\dot{S}-X} + \gamma\sigma^2_{\mathbf{\Delta}}. \tag{41}$$

Therefore, the prediction noise variance σ^2_{S-X} is linear to the MV distortion $\sigma^2_{\mathbf{\Delta}}$.

D. Distortion D as a Function of P_{coset} and σ^2_{S-X}

The derivation of the distortion D is as follows. Firstly, from (4) we have $\hat{X}-X = \hat{C}-C$ in high probability. Thus the distortion D approximately equals to the distortion of the coset value, that is

$$D = \sigma^2_{\hat{X}-X} \approx \sigma^2_{\hat{C}-C}. \tag{42}$$

Similar to section IV-B, we can derive and express the coset distortion as

$$\sigma^2_{\hat{C}-C} \approx \alpha_{coset}\sigma^2_C \left(\frac{P_{coset}}{n_{coset}N_0}\right)^{-1} \tag{43}$$

where α_{coset} is the coding gain of power allocation, σ^2_C is the variance of C, and n_{coset} is number of coset subbands.

In general, our DCast transmits the coset values of the source X over Gaussian channel, with the side information S at the receiver side. Therefore, for each subband, it forms a typical Wyner–Ziv dirty-paper problem, in which transmitting the coset values has been proven to be as efficient as transmitting the residue $S - X$ over the same channel (in case that $S - X$ is available to the encoder) [39]. Actually, according to the theorem in [39] (the existence of good lattice), the coset value C of each subband has the same variance with the prediction residue $S - X$ of each subband, that is

$$\sigma^2_{C_i} = \mathbb{E}\{C_i^2\} = \mathbb{E}\{(S_i - X_i)^2\}. \tag{44}$$

Thus the coset value and the prediction residue have the same variance in frame level, that is

$$\sigma^2_C = \mathbb{E}\{(S - X)^2\} = \sigma^2_{S-X}. \tag{45}$$

Therefore, (42),(43) and (45) implies

$$D = \sigma^2_{\hat{C}-C} \approx \alpha_{coset}\sigma^2_{S-X}\left(\frac{P_{coset}}{n_{coset}N_0}\right)^{-1}. \tag{46}$$

This means D is proportional to the prediction noise variance σ^2_{S-X} and the inverse of coset power P_{coset}

E. Solution

Substituting (34) and (41) into (46), we get

$$D=(\sigma^2_{\dot{S}-X}+\gamma\alpha_{mv}\sigma^2_M n_{mv}N_0 P_{mv}^{-1})\alpha_{coset}n_{coset}N_0 P_{coset}^{-1}. \tag{47}$$

Then taking (47) into the problem (28), and solving the problem, we get

$$\begin{aligned} P_{mv} &= [(A^2 + A)^{1/2} - A]P, \\ A &= \frac{\gamma\alpha_{mv}\sigma^2_M n_{mv}N_0 P^{-1}}{\sigma^2_{\dot{S}-X}}. \end{aligned} \tag{48}$$

Although it seems that A contains so many variables, there is actually a quite straightforward way to estimate A. In A, σ^2_M is the variance of the MV signal to transmit, $\frac{P}{n_{mv}N_0}$ is the SNR when all power is allocated to MV, and α_{mv} is the coding gain of the power allocation. This means that, if all power is allocated to MV, the MV distortion $\sigma^2_{\mathbf{\Delta}}$ will be $\alpha_{mv}\sigma^2_M n_{mv}N_0 P^{-1}$ according to (34). Furthermore, (34) together with (40), implies that $\gamma\alpha_{mv}\sigma^2_M n_{mv}N_0 P^{-1}$ is the variance of the additional prediction noise caused by erroneous MVs when all transmission power is allocated to MV. Therefore, the parameter A is estimated as follows. DCast simulates the transmission and decoding process to get for each frame a hypothetic side information S^*, which is the side information when all transmission power is allocated to MV data. DCast also calculates for each frame another hypothetical side information $\dot{S}$, which is the side information assuming the transmission of MVs are lossless. Since $S^* - \dot{S}$ is the additional prediction noise caused by erroneous MVs, we have

$$\sigma^2_{S^*-\dot{S}} = \gamma\alpha_{mv}\sigma^2_M n_{mv}N_0 P^{-1}. \tag{49}$$

With (49), the solution (48) is rewritten as

$$\begin{aligned} P_{mv} &= [(A^2 + A)^{1/2} - A]P \\ A &= \frac{\sigma^2_{S^*-\dot{S}}}{\sigma^2_{\dot{S}-X}}. \end{aligned} \tag{50}$$

Therefore, for optimal power distortion optimization, the encoder first estimates $\sigma^2_{S^*-\dot{S}}$ and $\sigma^2_{\dot{S}-X}$, and then calculates optimal MV transmission power P_{mv} by (50).

V. Experiments

In our experiments, we evaluate the performance of the proposed DCast in video streaming applications including both unicast and multicast. We compare DCast with Softcast [11], [12] and conventional frameworks. We have implemented two versions of Softcast based on 2-D-DCT and 3-D-DCT respectively, i.e. Softcast2-D [11] and Softcast3-D [12].

We also implement two conventional frameworks. One uses H.264 as video encoder and the other uses a DVC codec named Witsenhausen-Wyner Video Codec (WWVC) [17]. Both of the two frameworks use standard 802.11 PHY layer with FEC and QAM modulations. We use JM14.2 software as H.264 codec. For error resilience, the intra MB refresh rate is set to be 10%. Each video slice is packed into one RTP packet. We set the maximal slice size to be 1192 bytes such that the length of RTP packet is no greater than 1200 bytes. The WWVC coded bitstream is also packed into RTP packet of maximal length 1200bytes. We append to each RTP packet a 32-bits CRC, and then encode each packet separately. Similar to the experiments in [12], for error protection we apply on each packet an outer Reed-Solomon code with the same parameters (188/204) used for digital TV [40]. Each packet is individually interleaved between the outer Reed-Solomon code and the inner FEC in accordance with the same recommendation. For inner FEC, we generate the 1/2 convolutional code with polynomials {133, 171} and puncture it to get 2/3 and 3/4 convolutional codes. The FEC coded bits are mapped to the complex symbols by BPSK, QPSK, 16QAM or 64QAM. The complex symbols are then transmitted over OFDM. We assume the channel noise is Gaussian and the channel bandwidth is 1.15 MHz. The FEC decoding is done by soft Viterbi algorithm. After the FEC decoding and RS decoding, the decoder performs CRC check for each RTP packet, and forward those error-free packets to video decoders. The WWVC decoder performs Wyner-Ziv decoding and is able to reconstruct the video frames when the reference frames have some error. The H.264 decoder can also tolerate a small percentage of RTP packet loss, by utilizing the error concealment. In our test, we have configured the H.264 decoder to use the most complex error concealment method in JM14.2, the motion copy one, to get best reconstruction quality. The test video sequences are standard CIF sequences (352×288, 30 Hz), including *Akiyo, Bus, Coastguard, Crew, Flower, Football, Foreman, Harbour, Husky, Ice, News, Soccer,*

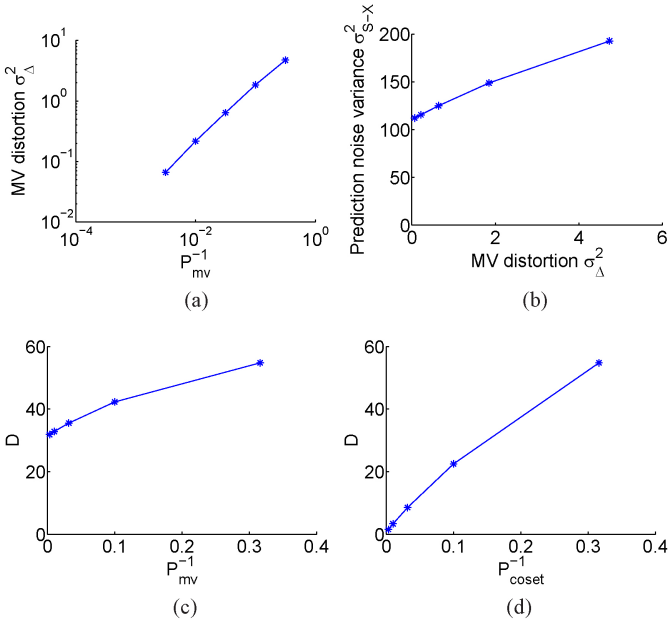


Fig. 4. Verification of the models of power distortion optimization in Section IV. P_{coset} and P_{MV} are transmission power of coset data and MV data respectively. D is reconstruction distortion.

Stefan, Tempete, Tennis, and *Waterfall*. To evaluate average performance of each framework, we also create a monochrome 512-frame test video sequence, called all_seq, by combining the first 32 frames of the above 16 test sequences.

For DCast, H.264 and WWVC, the GOP structure is 'IPPP' and the GOP length is 32. In the following tests, all the PSNR results are for all the frames including both intra and inter frames. The number of reference frame for inter frame is 1. In DCast, the intra frame coding is exactly the same as Softcast2-D and the inter frame coding is by proposed framework. The transmission power allocated to an intra frame is set to be 4 times of the power of an inter frame. According to our experiments, this approximately makes intra and inter frames have similar video PSNR. The search range of ME is 32×32 and the MV precision is $1/4$ pixel. In ME, DCast uses only 8×8 block size, while H.264 and WWVC use all the 7 block size from 4×4 to 16×16. Table I gives a summary of the techniques and configurations of these frameworks.

A. PDO Model Verification

This test is to verify the models of power distortion optimization (PDO) in Section IV. We use all_seq as the test sequence. In the first test, we fix the coset transmission power P_{coset} and let the MV transmission power P_{MV} change. The channel noise power N_0 is set to 1. The results are given in Fig. 4. Fig. 4(a) shows the relation between the MV transmission power P_{MV} and the MV distortion σ^2_Δ. According to the result, the inverse of P_{MV} is proportional to the MV distortion. This confirms the equation (34). Fig. 4(b) shows the linear relation between the MV distortion σ^2_Δ and the prediction noise variance σ^2_{S-X}. This verifies the model of equation (41). Fig. 4(c) shows the relation between the MV transmission power P_{MV} and the reconstruction distortion D. They are approximately in linear relation as shown in the equation (47).

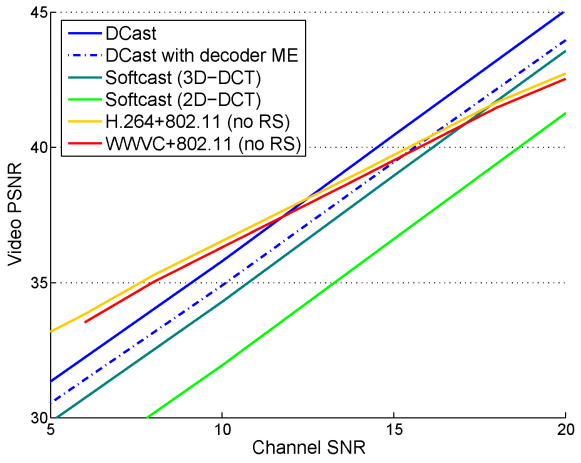


Fig. 5. Unicast performance comparison. Both the encoder and the decoder are assumed to know the channel SNR.

In the second test, we fix the MV transmission power P_{MV} and let the coset transmission power P_{coset} change. The channel noise power N_0 is set to 1. The result is given in Fig. 4(d). The reconstruction distortion D is proportional to the inverse of the coset transmission power P_{coset}. This verifies the model in equation (46) and (47).

B. Unicast Performance

This test is to compare unicast performance among all the above frameworks. In this test the input video is all_seq and the channel SNR is $5 - 20$ dB. Both the encoder and the decoder is assumed to know the channel SNR. For each channel SNR, the parameters of DCast are optimally tuned. The total transmission power is optimally allocated to coset data and motion data as explained in Section IV. The conventional framework is assumed to be able to choose the best combinations of the FEC and the QAM methods recommended by 802.11 according to the channel SNR, to get maximal bitrate for source coding layer. The RS coding is skipped in this unicast test. The source coding layer, i.e. the H.264 codec or WWVC codec, performs rate control to utilize the bitrate as much as possible.

The experimental result is given in Fig. 5. This figure compares the reconstruction quality of six frameworks at different channel SNR. The reconstruction quality is measured by video PSNR. DCast is uniformly 4 dB better in video PSNR than Softcast2D at all channel SNR, mainly due to enabling inter frame prediction. DCast gains about 1.5 dB in video PSNR over Softcast3D, which mainly comes from motion alignment. Compared with H.264 based framework, DCast is about 0.8 dB worse in video PSNR at low channel SNR but is about 2.9 dB better in video PSNR at high channel SNR. WWVC based framework performs slightly worse than H.264 based framework. In this test, we also implement another version of DCast in which the ME is performed at the decoder by motion compensated extrapolation [2]. Compared with conventional framework, the DCast with decoder ME is about

TABLE I
SUMMARY OF THE FOUR FRAMEWORKS

Frameworks	Softcast2D	Softcast3D	DCast	H.264/WWVC+802.11
GOP	IIII...	–	IPPP...	IPPP...
Reference frames	0	–	1	1
ME	N	N	Y	Y
ME block size	–	–	fixed	variable
ME search range	–	–	32×32	32×32
MV precision	–	–	1/4	1/4
DCT	2-D	3-D	2-D	2-D
Coding delay	1 frame	4 frames	1 frame	1 frame
Modulation	OFDM	OFDM	OFDM	OFDM
Constellation	64K-QAM, BPSK	64K-QAM, BPSK	64K-QAM, BPSK	BPSK, QPSK, 16-QAM, 64-QAM
FEC rate	1/2 (BPSK only)	1/2 (BPSK only)	1/2 (BPSK only)	1/2, 2/3, 3/4
RS rate	–	–	–	188/204

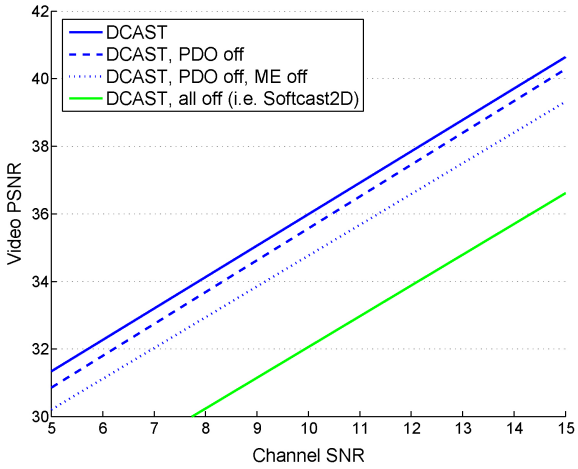


Fig. 6. Evaluation of each module. The contribution of coset coding, ME and PDO are about 2.7 dB, 0.8 dB and 0.5 dB in video PSNR respectively.

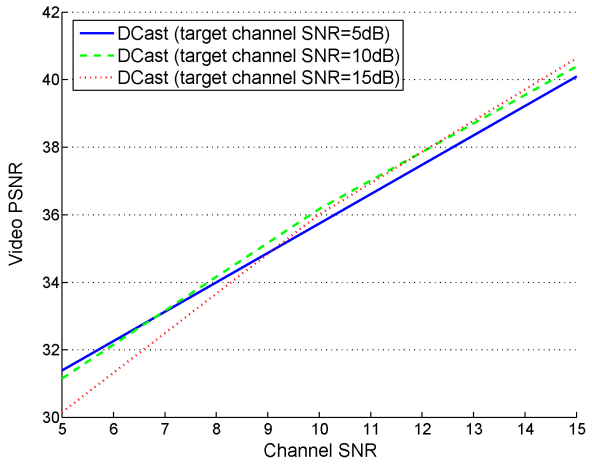


Fig. 7. Robustness test. DCast is configured to optimized for target channel SNR of 5 dB, 10 dB and 15 dB respectively, and then tested under different channel SNR.

1.6 dB worse in video PSNR at low channel SNR but is 1.7 dB better in video PSNR at high channel SNR.

Note that the result in Fig. 5 does not mean DCast can outperform H.264 in compression efficiency. H.264 is a video coding standard while DCast is a wireless video transmission framework. H.264 has very high compression efficiency but the bitstream is not very robust to error. This is why H.264 bitstream needs additional FEC bits to protect. DCast may not be as efficient as H.264 in video compression, but is robust to channel noise. Thus, it can skip FEC and can use a very dense 64K-QAM modulation, and achieves high transmission efficiency.

C. Evaluation of Each Module

DCast has several modules such as coset coding, motion estimation (ME) and power distortion optimization (PDO). In the following test, we incrementally turn off these modules in DCast to evaluate their contribution. In this test the input video is all_seq, and the channel SNR is $5 - 15$ dB. The test results are given in Fig. 6. In this figure, "PDO off" means there are no PDO and the encoder utilizes an adhoc power allocation where the total transmission power is equally allocated between motion data and coset data, i.e. $\frac{P_{mv}}{N_{mv}} = \frac{P_{coset}}{N_{coset}}$. "ME off" means there are no ME and the decoder uses previous reconstructed frame directly as side information. Note that there are dependencies between the three modules (coset, ME, and PDO). When ME is disabled, the PDO must be off because there is no MV to transmit. When coset coding is disabled, the ME should be disabled also because the decoder no longer needs side information. Furthermore, when all the three modules (coset, ME and PDO) are off, the DCast becomes the same as Softcast2-D. According to the result in Fig. 6, the contribution of coset coding, ME and PDO are about 2.7 dB, 0.8 dB and 0.5 dB respectively in video PSNR.

D. Robustness Test

In practical wireless applications, the channel SNR may not be perfectly known to the encoder. In the following tests, we will evaluate the performance of DCast in this situation. The input video is all_seq and the channel SNR is $5 - 15$ dB. We let DCast to optimize for target channel SNR of 5 dB, 10 dB and 15 dB respectively. The video PSNR are compared in Fig. 7. According to the result, each of the three encoders performs best when the practical channel

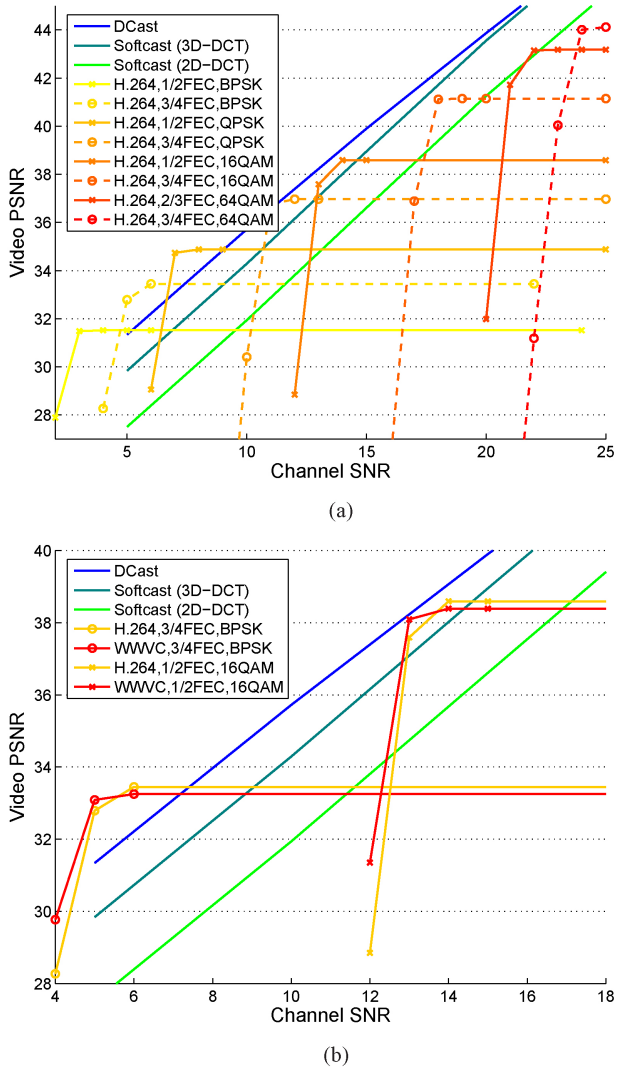


Fig. 8. Robustness comparison between DCast and (a) H.264 and (b) another DVC framework: WWVC. Channel SNR is unknown to all the encoders. DCast encoder is optimized for channel SNR of 5 dB.

SNR matches its optimization target, but performs slightly worse than the best one when the practical channel SNR does not match the target. The one optimized for 15 dB channel performs 1.2 dB lower in video PSNR than the optimal one when the practical channel SNR is 5 dB. This indicates that DCast should optimize for a lower channel SNR for more robustness in multicast.

We then compare DCast with the conventional frameworks based on H.264 and WWVC. Still we assume that only the decoder knows the channel SNR. DCast is optimized for a target channel SNR of 5 dB in the following tests. For conventional framework, we implement all the eight recommended combination of channel coding and modulation of 802.11. We calculate the corresponding bitrates respectively according to the bandwidth, and set the bitrates constraint to the H.264 encoder and WWVC encoder for rate control. Both the video bitrate and the channel bitrate (the bitrate after RS coding and FEC) under the eight transmission approaches are given in Table II (Note that WWVC and H.264 have same bitrate constraints.). For DCast, there are no bitrate but only channel

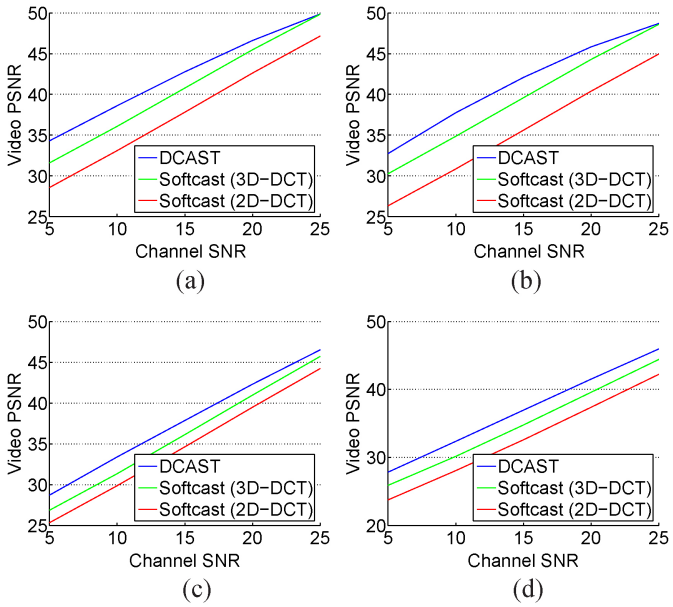


Fig. 9. Multicast performance on different video sequences.

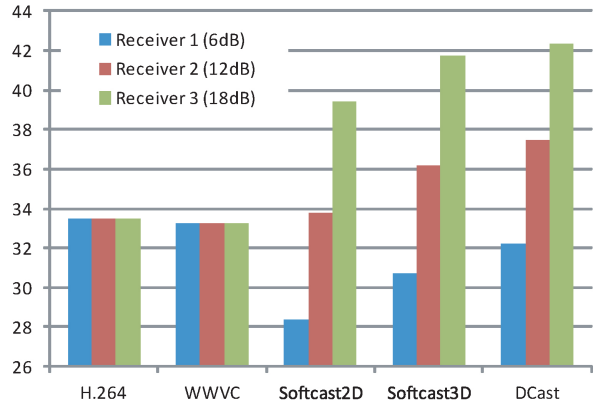


Fig. 10. Multicast to three receivers.

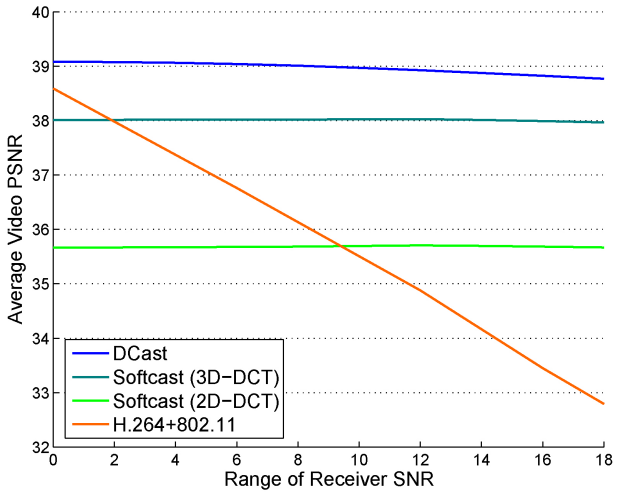


Fig. 11. Serving a group of receivers with diverse channel SNR. The average channel SNR of each group is 14 dB.

Fig. 12. Visual quality comparison, channel SNR is 5 dB. (a) Original frame. (b) Softcast2D. (c) Softcast3D. d) DCast.

symbol rate. Note that all the frameworks consume the same bandwidth and transmission power.

The video PSNR of each framework under different channel SNR is given in Fig. 8. In Fig. 8(a), all eight conventional transmission approaches suffer a very serious cliff effect. For example, the approach 'H.264,1/2FEC,16QAM' performs well when channel SNR is between 13 dB to 14 dB, but is not good when channel SNR is out of this range. When the channel SNR becomes more than 14 dB, the reconstruction quality does not increase. When the channel SNR becomes 12 dB, the reconstruction quality drops very quickly. When the channel SNR becomes even lower, the video decoder cannot work since almost all received RTP packets have bit error. Note that the cliff effect can be partially mitigated in a layered approach [41] combining the scalable video extension of H.264 and a hierarchical modulation PHY layer. However, as shown in [12], the layered approach needs a higher channel SNR than the single layer approach to achieve the same PSNR. Fig. 8(b) shows the performance of WWVC based framework. In erroneous situation, WWVC can benefit from Wyner-Ziv decoding and gains 1-2 dB in video PSNR over H.264. This complies with the results in [17]. However, it still suffers a very serious cliff effect.

In contrast, the three all-in-one frameworks do not suffer the cliff effect. When the channel SNR increases, the reconstruction PSNR increases accordingly, and vice versa. DCast is still the best one among the three all-in-one frameworks. At low channel SNR, DCast is still 1.5 dB and 4 dB better in video PSNR than Softcast3D and Softcast2D respectively. However, when the channel SNR increases, the gain of DCast decreases. When channel SNR is 25 dB, DCast performs similar to Softcast3D and gains only about 2.5 dB in video PSNR over Softcast2D. Compared with the unicast result in Fig. 5, the performance of DCast becomes 1.5 dB worse in video PSNR at high channel SNR. This is mainly due to the fact that the optimization of DCast (including both the PDO and the coset quantization step) is for 5 dB channel SNR in this test. Fig. 9 gives the performance comparison on different video sequence.

E. Multicast Performance

Next, we let all the frameworks serve a group of three receivers with diverse channel SNR. The channel SNR for each receiver is 6 dB, 12 dB, and 18 dB, respectively. The test result is given in Fig. 10. In conventional frameworks based on H.264 and WWVC, the server transmits the video stream by using 3/4 FEC and BPSK. It cannot use higher transmission rate because in that case the 6 dB user will not be able to decode the video. Due to this, although the other two receivers have better channel conditions, they will also receive low speed 802.11 signal, and reconstruct low quality video. In Softcast and DCast, the server can accommodate all the receivers simultaneously. Using DCast, the 6 dB user can get slightly lower reconstruction quality than using H.264 or WWVC based conventional frameworks. However, the 12 dB and 18 dB users get 4 dB and 8 dB better reconstruction quality respectively by using DCast other than conventional frameworks.

TABLE II
COMPARISON OF COMPLEXITY AND BITRATE

	Encode time	Decode time	Video bit rate	Channel bit rate	Channel symbol rate
H.264+1/2FEC+BPSK	387 ms	7 ms	530 Kb/s	1.15 Mb/s	1.15 M/s
H.264+3/4FEC+BPSK	387 ms	8 ms	795 Kb/s		
H.264+1/2FEC+QPSK	406 ms	9 ms	1060 Kb/s	2.3 Mb/s	
H.264+3/4FEC+QPSK	389 ms	10 ms	1590 Kb/s		
H.264+1/2FEC+16QAM	381 ms	11 ms	2120 Kb/s	4.6 Mb/s	
H.264+3/4FEC+16QAM	385 ms	14 ms	3180 Kb/s		
H.264+2/3FEC+64QAM	371 ms	15 ms	4240 Kb/s	6.9 Mb/s	
H.264+3/4FEC+64QAM	427 ms	16 ms	4770 Kb/s		
DCast	304 ms	10 ms	–	–	

Fig. 11 compares the multicast performance of four frameworks, with respect to the range of receiver SNR. The range of receiver SNR is defined as the difference of the maximal and minimal channel SNR of the users in the group. The average channel SNR of the users in group is 14 dB. When the channel SNR range is 0 dB, i.e., the channel SNR of all the users are equally 14 dB, DCast, Softcast3D and H.264 framework performs similar. However, when the users' channel SNR becomes diverse, the performance of H.264 framework drops quickly.

The visual quality comparison is given in Fig. 12. The channel SNR is set to be 5 dB. DCast has clearly better visual quality than both Softcast2-D and Softcast3-D.

In all the tests, including unicast and multicast, DCast performs better than both Softcast2-D and Softcast3-D. Moreover, DCast does not introduce frame delays as Softcast3-D does, and is applicable for realtime video multicast like Softcast2-D.

F. Complexity and Bitrate

The proposed DCast allows the ME to be performed at encoder. Therefore the encoder would be in high complexity but the decoder would be in low complexity. Table II shows the average encoding time and decoding time per frame in millisecond. The test machine has a Pentium (R) Dual-Core CPU E5300 @ 2.60 GHz, 2G internal memory and Microsoft Windows XP Professional 5.1.2600, with Service Pack 3. The input video is all_seq of 'CIF' size at 30 frames per second. DCast has less encoding time than H.264 codec (JM14.2) possibly because that DCast has no mode decision and no entropy coding. As to the decoding time, DCast is comparable to the H.264 codec.

Table II also shows the video bitrate and channel bitrate of H.264 solutions. For example, when the modulation is BPSK, the channel bitrate is equal to the channel symbol rate, i.e. 1.15 M/s. If the FEC is 1/2 convolutional code and the RS code is 188/204, then the video bitrate is $1.15\,M \times \frac{1}{2} \times \frac{188}{204} =$ 530 Kb/s. When the modulation is QPSK and the FEC is 3/4 convolutional code, then the channel bitrate is 2.3 Mb/s and the video bitrate is 1590 Kb/s. The decoding time of H.264 codec depends on the video bitrate. Basically, the decoding time becomes longer when the bitrate increases. The DCast framework has no bitrate but a universal channel symbol rate. Its decoding time is fixed and is similar to the decoding time of H.264 decoder at bitrate 1590 Kb/s.

VI. CONCLUSION

In this paper, we proposed a novel framework called DCast for distributed video coding, and transmission over wireless networks. DCast first presented a new design on how to efficiently transmit distributed coded video data over Gaussian channel. Furthermore, we also proposed a new power distortion optimization for the proposed DCast.

DCast avoided the annoying cliff effect of conventional frameworks caused by the mismatch between transmission rate and channel condition. A single DCast server accommodated multiple users with diverse channel SNRs simultaneously in multicast without sacrificing any user's coding performance approximately. As shown in the experiments, DCast performed competitively with H.264 framework in unicast but gained up to 8 dB in video PSNR in multicast.

DCast, as a unique DVC framework, did not utilize some sophisticated video coding tools such as variable block ME, intra mode, or mode decision. How to enable these tools to further improve the performance of DCast is one possible future work. Furthermore, the DCast in this paper was mainly designed and optimized for Gaussian channel. Another opportunity for future work is to extend the proposed DCast to fading channel which may require more complicated channel estimation and power distortion optimization.

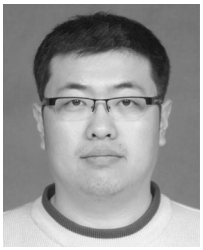
ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their valuable comments that greatly improved this paper.

REFERENCES

[1] A. Aaron, R. Zhang, and B. Girod, "Wyner–Ziv coding of motion video," in *Proc. 36th Asilomar Conf. Signals Syst. Comput.*, 2002.
[2] 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.
[3] R. Puri and K. Ramchandran, "PRISM: A new robust video coding architecture based on distributed compression principles," in *Proc. Annu. Allerton Conf. Commun. Control Comput.*, 2002.
[4] 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.
[5] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, vol. 19, no. 4, pp. 471–480, Jul. 1973.
[6] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, vol. 22, no. 1, pp. 1–10, Jan. 1976.
[7] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H. 264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[8] Q. Xu and Z. Xiong, "Layered Wyner–Ziv video coding," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3791–3803, Dec. 2006.
[9] Q. Xu, V. Stankovic, and Z. Xiong, "Distributed joint source-channel coding of video using Raptor codes," *IEEE J. Select. Areas Commun.*, vol. 25, no. 4, pp. 851–861, May 2007.
[10] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Joint source-channel coding of binary sources with side information at the decoder using ira codes," in *Proc. IEEE Workshop Multimedia Signal Process.*, 2002, pp. 53–56.
[11] S. Jakubczak and D. Katabi, "SoftCast: One-size-fits-all wireless video," in *Proc. ACM SIGCOMM Comput. Commun. Rev.*, 2010.
[12] S. Jakubczak and D. Katabi, "A cross-layer design for scalable mobile video," in *Proc. 17th Annu. Int. Conf. Mobile Comput. Networking*, 2011.
[13] S. J. Choi and J. W. Woods, "Motion-compensated 3-D subband coding of video," *IEEE Trans. Image Process.*, vol. 8, no. 2, pp. 155–167, Feb. 1999.
[14] A. Secker and D. Taubman, "Lifting-based invertible motion adaptive transform (LIMAT) framework for highly scalable video compression," *IEEE Trans. Image Process.*, vol. 12, no. 12, pp. 1530–1542, Dec. 2003.
[15] R. Xiong, J. Xu, F. Wu, and S. Li, "Barbell-lifting based 3-D wavelet coding scheme," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1256–1269, Sep. 2007.
[16] Y. Zhang, C. Zhu, and K. H. Yap, "A joint source-channel video coding scheme based on distributed source coding," *IEEE Trans. Multimedia*, vol. 10, no. 8, pp. 1648–1656, Dec. 2008.
[17] M. Guo, Z. Xiong, F. Wu, D. Zhao, X. Ji, and W. Gao, "Witsenhausen-Wyner video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 8, pp. 1049–1060, Aug. 2011.
[18] X. Fan, F. Wu, and D. Zhao, "D-cast: DSC based soft mobile video broadcast," in *Proc. 10th Int. Conf. Mobile Ubiquitous Multimedia*, 2011.
[19] X. Fan, F. Wu, D. Zhao, O. C. Au, and W. Gao, "Distributed soft video broadcast (DCAST) with explicit motion," in *Proc. Data Compression Conf.*, 2012.
[20] A. Secker and D. Taubman, "Highly scalable video compression with scalable motion coding," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1029–1041, Aug. 2004.
[21] J. Garcia-Frias, "Compression of correlated binary sources using turbo codes," *IEEE Commun. Lett.*, vol. 5, no. 10, pp. 417–419, Oct. 2001.
[22] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Commun. Lett.*, vol. 6, no. 10, pp. 440–442, Oct. 2002.
[23] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: architecture, techniques and evaluation," in *Proc. Picture Coding Symp.*, vol. 6. 2007, pp. 14496–10.
[24] A. Aaron, S. Rane, E. Setton, B. Girod, *et al*., "Transform-domain Wyner-Ziv codec for video," in *Proc. SPIE Visual Commun. Image Process.*, 2004.
[25] X. Guo, Y. Lu, F. Wu, D. Zhao, and W. Gao, "Wyner-Ziv-based multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 6, pp. 713–724, Jun. 2008.
[26] M. Tagliasacchi, A. Trapanese, S. Tubaro, J. Ascenso, C. Brites, and F. Pereira, "Intra mode decision based on spatio-temporal cues in pixel domain Wyner–Ziv video coding," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2006.
[27] J. Slowack, S. Mys, J. Skorupa, N. Deligiannis, P. Lambert, A. Munteanu, and R. Van de Walle, "Rate-distortion driven decoder-side bitplane mode decision for distributed video coding," *Signal Processing: Image Commun.*, vol. 25, no. 9, pp. 660–673, 2010.
[28] S. Benierbah and M. Khamadja, "Generalized hybrid intra and Wyner–Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 12, pp. 1929–1934, 2011.
[29] A. Aaron, S. Rane, and B. Girod, "Wyner–Ziv video coding with hash-based motion compensation at the receiver," in *Proc. Int. Conf. Image Process.*, 2004.
[30] R. Martins, C. Brites, J. Ascenso, and F. Pereira, "Refining side information for improved transform domain Wyner–Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 9, pp. 1327–1341, Sep. 2009.
[31] B. Macchiavello, D. Mukherjee, and R. L. De Queiroz, "Iterative side-information generation in a mixed resolution wyner-ziv framework," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 10, pp. 1409–1423, Oct. 2009.
[32] X. Fan, O. C. Au, N. M. Cheung, Y. Chen, and J. Zhou, "Successive refinement based Wyner-Ziv video compression," *Signal Process. Image Commun.*, vol. 25, no. 1, pp. 47–63, 2010.
[33] W. Liu, L. Dong, and W. Zeng, "Motion refinement based progressive side-information estimation for Wyner-Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1863–1875, Dec. 2010.
[34] C. Brites and F. Pereira, "Correlation noise modeling for efficient pixel and transform domain Wyner-Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 9, pp. 1177–1190, Sep. 2008.
[35] X. Fan, O. C. Au, and N. M. Cheung, "Transform-domain adaptive correlation estimation (TRACE) for Wyner–Ziv video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1423–1436, Nov. 2010.
[36] N. Deligiannis, J. Barbarien, M. Jacobs, A. Munteanu, A. Skodras, and P. Schelkens, "Side-information dependent correlation channel estimation in hash-based distributed video coding," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1934–1949, Apr. 2012.
[37] G. R. Esmaili and P. C. Cosman, "Wyner-Ziv video coding with classified correlation noise estimation and key frame coding mode selection," *IEEE Trans. Image Process.*, vol. 20, no. 9, pp. 2463–2474, Sep. 2011.
[38] S. Aditya and S. Katti, "FlexCast: Graceful wireless video streaming," in *Proc. 17th Annu. Int. Conf. Mobile Comput. Networking*, 2011.
[39] Y. Kochman and R. Zamir, "Joint Wyner–Ziv/dirty-paper coding by modulo-lattice modulation," *IEEE Trans. Inform. Theory*, vol. 55, no. 11, pp. 4878–4889, Nov. 2009.
[40] ETSI. (2009). *Digital Video Broadcasting (DVB)* [Online]. Available: http://www.etsi.org/deliver/etsi_en/300700_300799/300744/01.06.01_60/en_300744v010601p.pdf
[41] T. Kratochvíl, "Hierarchical modulation in DVB-T/H mobile TV transmission," in *Multi-Carrier Systems and Solutions*. The Netherlands: Springer, 2009, pp. 333–341.



Xiaopeng Fan (S'07–M'09) received the B.S. and M.S. degrees from the Harbin Institute of Technology (HIT), Harbin, China, in 2001 and 2003, respectively, and the Ph.D degree from the Hong Kong University of Science and Technology (HKUST), Kowloon, Hong Kong, in 2009.

In 2009, he joined the Department of Computer Science, HIT, where he is currently an Associate Professor. From 2003 to 2005, he was with the Intel China Software Laboratory as a Software Engineer. He has authored or co-authored over 50 technical journal and conference papers. His current research interests include image/video coding and processing, video streaming and wireless communication.

Feng Wu (F'12) received the B.S. degree in electrical engineering from XIDIAN University in 1992, and the M.S. and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1996 and 1999, respectively.

In 1999, he joined Microsoft Research Asia, Beijing, China, where he is currently a Senior Researcher/Research Manager. He has authored or co-authored over 200 publications, including 50 journal papers. He has had 13 of his techniques adopted into international video coding standards. His current research interests include image and video compression, media communication, and media analysis and synthesis.

Dr. Wu serves as an Associate Editor for various publications, such as the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY AND IEEE TRANSACTIONS ON MULTIMEDIA. He served as the Technical Program Committee (TPC) Chair in MMSP 2011, VCIP 2010 and PCM 2009, the TPC Track Chair in ICME 2013, ICIP 2012, ICME 2012, ICME 2011 and ICME 2009, and the Special Sessions Chair in ISCAS 2013 and ICME 2010. He was the recipient of the Best Paper Award in IEEE T-CSVT 2009, PCM 2008 and SPIE VCIP 2007.

Debin Zhao (M'11) received the B.S., M.S., and Ph.D. degrees in computer science from Harbin Institute of Technology (HIT), Harbin, China, in 1985, 1988, and 1998, respectively.

He is now a Professor at the Department of Computer Science, HIT. He has published over 200 journal and conference papers.

Oscar C. Au (F'11) received the B.A.Sc. from the University of Toronto, Toronto, Canada, in 1986, the M.A. and Ph.D. degrees from Princeton University, Princeton, NJ, USA, in 1988 and 1991, respectively.

He is a Professor at the Department of Electronic and Computer Engineering, HKUST, Hong Kong, China. He has published 300+ papers and 70+ contributions to international standards.