

Symmetric Distributed Multiview Video Coding

Baochun Bai*, Yang Yang†, Cheng Lei*, Pierre Boulanger*, and Janelle Harms*

* Department of Computing Science, University of Alberta, Edmonton, Canada

Email: {baochun, clei, pierreb, harms}@cs.ualberta.ca

† Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA

Email: {yangyang}@ece.tamu.edu

Abstract—Symmetric distributed source coding has made great progress in the past several years. However, few attempts have been tried to use the symmetric distributed source coding schemes in real data compression scenarios. In this paper, with the inspiration of previous symmetric distributed source coding schemes, we first put forward a novel syndrome-based symmetric distributed coding scheme which can achieve the whole Slepian-Wolf rate region. Then we propose a general architecture for symmetric distributed multiview video schemes. Our experiments show very encouraging results. In the high-rate case, our scheme can achieve great rate saving from separate H.264 coding scheme at the same video quality.

Index Terms—symmetric distributed multiview video coding, distributed source coding, LDPC, nonuniform channel, H.264

I. INTRODUCTION

Multiview video applications such as 3D TV and wireless camera sensor network have become more and more popular and attract lots of research interest in the past decade. Multiview video systems normally use tens and hundreds of synchronized cameras to capture an enormous amount of video data for storage and transmission. Therefore, it is indispensable to design efficient and effective compression algorithms to reduce the size of video data. A popular approach to multiview video compression is to use the traditional block-based hybrid coding method to remove the intra-stream and inter-stream redundancies among multiple video streams. This is normally referred as joint multiview video coding (JMVC). For example, Joint Video Team (JVT) of MPEG and ITU-T is now developing a Joint Multiview Video Model (JMVM) which is based on the H.264 hybrid video coding standard [1]. However, JMVC demands communication between cameras to achieve compression.

Recently distributed multiview video coding (DMVC), which is rooted in the distributed source coding theory [2], has gained lots of research attention. Distributed source coding that exploits the statistics of source signals at the decoder to achieve the compression is essentially different from joint source coding such as methods standardized by MPEG and H.26x which achieves the compression by exploiting the source statistics at the encoder. Since then, distributed source coding has attracted lots of researchers' interest and many results have been produced [2]. Practical distributed video coding algorithms have also been developed and [3] summarizes recent advances on distributed video coding. In theory, DMVC holds the promise to achieve the same compression performance as JMVC while demanding no communication between cameras. Several distributed multiview video codecs [4], [5] are recently proposed. However, those approaches are based on asymmetric Slepian-Wolf codes which can not achieve the whole Slepian-Wolf rate region and thus limit the rate allocation options between the encoders.

Thirumalai et. al [6] and Taglisacchi et. al [7] use source splitting and asymmetric Slepian-Wolf codes to realize symmetric distributed multiview video coding. However, source splitting might incur performance loss. Recently several researchers have designed a capacity-approaching symmetric Slepian-Wolf code which can achieve the

whole Slepian-Wolf rate region [8], [9], [10]. Though these solutions are elegant, it is difficult to use them to design symmetric distributed multiview video codec since they implicitly assume that an encoder knows the exact bit correspondence between correlated sources. In the case of distributed multiview video coding, pixel correspondence between two correlated images is not known at the encoder and can only be inferred at the decoder since there is no communication channel between two encoders. Without the assumption of bit correspondence at the encoders, those approaches fail to decode the original source at the decoder. Though the approaches proposed in [11], [12] can handle the bit incorespondence problem, it is still elusive to design capacity-approaching codes.

In this paper, we first put forward a novel syndrome-based approach to realize the symmetric Slepian-Wolf coding which can achieve the whole Slepian-Wolf rate region. The basic idea is to use nonuniform channel [13] to design LDPC codes used for symmetric Slepian-Wolf coding. We then describe a symmetric distributed multiview video codec (SDMVC) that uses the proposed symmetric Slepian-Wolf code and is able to handle the pixel incorespondence problem at the encoder. In addition, the proposed symmetric multiview video codec can outperform separate H.264 coding of two stereo video sequences.

II. SYMMETRIC SLEPIAN-WOLF CODE

Since distributed source coding is dual to channel coding, the dependency between two correlated sources can be modeled as a virtual correlation channel. Thus powerful channel codes such as Turbo codes or low density parity check codes (LDPC) can be used to realize Slepian-Wolf codes. Distributed video coding is the application of Slepian-Wolf codes on video data. Previous distributed multiview video codecs [3] are based on the asymmetric Slepian-Wolf coding schemes. Symmetric distributed multiview video coding is expected to achieve better performance by using capacity-achieving symmetric Slepian-Wolf codes.

In this section, we propose a novel approach to realize symmetric Slepian-Wolf coding. We call the proposed approach the Syndrome-based Nonuniform Symmetric Slepian-Wolf Coding scheme (SNS-SWC). It is inspired by the work [10], [12]. The basic idea of the proposed approach is to design two nonuniform LDPC codes [13] and let each source transmit a complement set of source bits and syndrome bits. Its architecture is illustrated in Fig. 1.

Given two correlated n -bit sources, X and Y , Slepian-Wolf theorem dictates that compression rates $R_X \geq H(X|Y)$, $R_Y \geq H(Y|X)$ and $R_X + R_Y \geq H(X, Y)$. To facilitate exposition of basic idea, the virtual correlation channel between two sources is assumed to be a BSC (Binary Symmetric Channel). In this case, $R_X \geq H(p)$, $R_Y \geq H(p)$ and $R_X + R_Y \geq 1 + H(p)$, where p is the crossover probability $P(X \neq Y|X)$. We intend to design a symmetric Slepian-Wolf coding scheme to achieve the following compression rates: $R_X = \frac{n_X + (n - n_X)H(p)}{n}$ and $R_Y = \frac{n - n_X + n_X H(p)}{n}$, where n_X

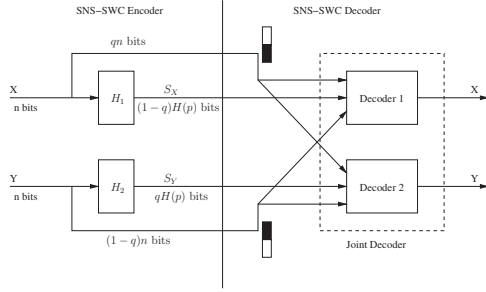


Fig. 1. The Architecture of the Syndrome-based Nonuniform Symmetric Slepian-Wolf Coding Scheme

is the number of source bits directly transmitted by source X. Let $q = \frac{n_X}{n}$ represent the fraction of transmitted source bits. Then we have $R_X = q + (1-q)H(p)$ and $R_Y = 1-q + qH(p)$.

Thus the rate of LDPC codes, r_X and r_Y , that realize the proposed symmetric Slepian-Wolf coding scheme should satisfy the following constraints: $r_X = \frac{k_X}{n} = \frac{n - (n - n_X)H(p)}{n} = 1 - (1-q)H(p)$ and $r_Y = \frac{k_Y}{n} = \frac{n - n_X H(p)}{n} = 1 - qH(p)$, where k_X and k_Y is the number of information bits of LDPC codes used to compress source X and source Y. In the case of $q = 1/2$, namely the encoder uses the equal rate to compress source X and Y, we have $R_X = R_Y = \frac{1+H(p)}{2}$ and $r_X = r_Y = 1 - \frac{1}{2}H(p)$.

A. LDPC Code Design

The LDPC code used in SNS-SWC can be deemed as an LDPC code for two parallel channels as shown in Fig. 2, where one part of the code bits are transmitted to the decoder losslessly through a perfect channel and while the other part of code bits are not transmitted, their information can be inferred from the perfectly available complement bits of the other source based on the virtual correlation channel between them. Let $Z_i, i = 1, 2$, denote the random variable that is equal to the log-likelihood ratio (LLR) of a received bit from the i th channel. The LLR distribution of a received bit can be represented by a single channel by using the two channels. Namely

$$P_{Z_X}(z) = qP_{Z_1}(z) + (1-q)P_{Z_2}(z) \quad (1)$$

$$P_{Z_Y}(z) = (1-q)P_{Z_1}(z) + qP_{Z_2}(z) \quad (2)$$

Where Z_1 is the perfect channel and Z_2 is the BSC.

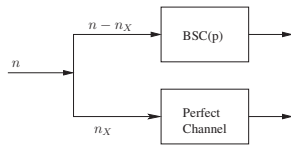


Fig. 2. The Illustration of Two Parallel Channels

The proposed LDPC code has similar format to the symmetric Slepian-Wolf code in [10]. They both are LDPC codes for a nonuniform channel [13] which are composed of two parallel channels. Therefore, the proposed LDPC code profile can be designed using either the classical single-channel density evolution algorithm [14] or the density evolution algorithm of nonuniform channels in [13]. The difference between the two approaches is that our approach is a syndrome-based approach and treats the n source bits as the code bits and uses the parity check matrix to encode the sources while the

approach in [10] is a parity-based approach and treats the n source bits as the information bits and uses the generator matrix to encode the sources.

B. Encoding

The left part of Fig. 1 illustrates the encoder structure of the proposed symmetric Slepian-Wolf code. Each encoder transmits a complement set of source bits and the corresponding syndrome bits. Different rates of two sources are achieved by adjusting the proportion of source bits to be perfectly transmitted.

C. Decoding

The decoder structure is shown in the right part of Fig. 1. The decoder includes a separate LDPC decoder for each source X and Y. The architecture makes it possible to avoid dependent decoding. Hence, it prevents error propagation that will happen in [9], [8], [11], [12]. The decoding algorithm in each LDPC decoder is the standard message passing algorithm. The only difference between the decoding algorithm in SNS-SWC and the original LDPC decoding algorithm is the initialization of LLR values. The initial LLR values of all bits used in the message passing algorithm are set based on the type of channel to which the bit belongs. The bits passing through the perfect channel have their LLR values set as ∞ or $-\infty$. The LLR values of the bits passing through the BSC(p) channel are equal to either $\log \frac{1-p}{p}$ or $\log \frac{p}{1-p}$.

III. SYMMETRIC DISTRIBUTED MULTIVIEW VIDEO CODING

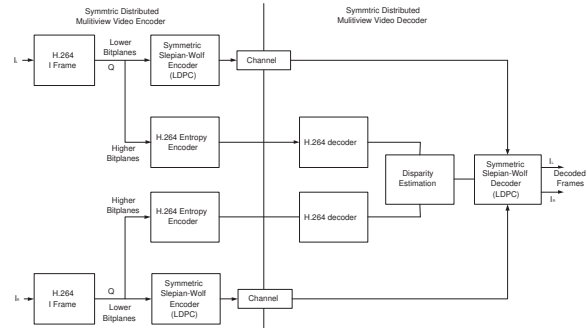


Fig. 3. The Architecture of The Symmetric Distributed Multiview Video Coding Scheme

The architecture of the proposed symmetric distributed multiview video coding (SDMVC) scheme is illustrated in Fig. 3. The proposed SDMVC uses the SNS-SWC discussed in Section II as the underlying distributed source coding scheme. We call our symmetric multiview video coding scheme SDMVC-SNS. Let $I_L = \{I_{L_1}, I_{L_2}, \dots, I_{L_n}\}$ and $I_R = \{I_{R_1}, I_{R_2}, \dots, I_{R_n}\}$ be the left and right stereo video sequences, respectively. The frames of both left and right video sequences are first quantized and intra-coded as the H.264 I-frames. Then the higher bit planes of residual DCT coefficients are coded by the H.264 entropy encoder. The lower bit planes of the residual DCT coefficients are compressed by the SNS-SWC introduced in Section II. At the decoder, the higher bit planes of DCT coefficients are first used to construct the low quality frames \hat{I}_{L_i} and $\hat{I}_{R_i}, i = 1, 2, \dots, n$. Then a rough disparity map is estimated from the reconstructed low quality frames. The pixel correspondence can be inferred from the disparity map and then the side information $I_{L_i}^s$ and $I_{R_i}^s$ for I_{L_i} and I_{R_i} ($i = 1, 2, \dots, n$) can be estimated. Finally the lower bit planes can be decoded by using the side information

and the higher bit planes. We will elaborate on the details of the algorithms below.

A. Handling Pixel Inconspendence at The Encoder

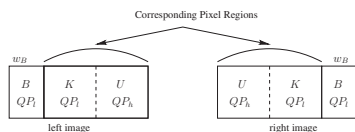


Fig. 4. Handling Pixel Inconspendence at The Encoder

As we know from the architecture of SNS-SWC elaborated in Section II, the decoding process of SNS-SWC is essentially two separate asymmetric Slepian-Wolf decoding procedures. This property makes it possible for our scheme to handle the bit inconspendence at the encoder while other approaches such as [9], [8] will fail. The output of the SNS-SWC composes of two parts, syndrome bits and a subset of the source bits. The aim of the SNS-SWC decoder is to recover the unknown part of source bits. The unknown source bits can be recovered correctly if we can successfully estimate the side information and then calculate the correct initial LLR values for those unknown source bits. We use Fig. 4 to illustrate how we carefully design our SDMVC-SNS encoder to handle the pixel inconspendence problem at the encoder. For easy exposition, we assume that there is only horizontal disparity between the left and right image and the maximal disparity is d_{max} . Pixels in the left or right frame are partitioned into 3 regions: boundary pixel region (B), known pixel region (K) and unknown pixel region (U). Lower bit planes of pixels in the region K and U are coded by SNS-SWC. Pixels in the region B are not used in Slepian-Wolf coding since they can be occluded with a very high probability and thus can not find corresponding pixels in the other image. The width of the region B , w_B , should be greater than d_{max} . In addition, since a macroblock is a basic coding unit in H.264, w_B should be divided by the macroblock width (16 pixels). Since the corresponding pixels for pixels in the region U of an image are located in the region B and K of the other image which are known at the decoder, the side information for pixels in the region U can be estimated. Thus the bit planes for pixels in region U can be recovered even though the SNS-SWC encoder is oblivious to the pixel correspondence information between the left and right image.

B. Multi-Level Bit Plane Slepian-Wolf Coding

For each 4x4 block in H.264, its residual pixel values are transformed by DCT and quantized. The number of bit planes for quantized DCT coefficients in H.264 is essentially controlled by the quantization parameter (QP). Suppose that k lower bit planes are coded by the SNS-SWC. Quantization step size in H.264 doubles for every increment of 6 in QP [15]. Given two quantization parameters, QP_l and QP_h , let $QP_l = QP_h - 6k$, we can use them to quantize the DCT coefficients and get the lower k bit planes that are encoded by SNS-SWC.

As shown in Fig. 4, QP_h is used to quantize pixels in the region B and K and QP_l is used to quantize pixels in the region U . For a 4x4 block, let c denote its original pixel values and R_p be its intra-predicted pixel values. The unquantized residual coefficients, R , are equal to $c - R_p$. Thus the quantized residual coefficients, R_q , are $Q(DCT(R), QP)$, where $Q()$ is a quantization function and $DCT()$ is the discrete cosine transform function. At the decoder the reconstructed residual coefficients, \hat{R}_c , are $\hat{R}_c = IDCT(IQ(R_q, QP))$,

where $IQ()$ is an inverse quantization function and $IDCT()$ is an inverse discrete cosine transform function. The reconstructed pixel values, \hat{c} , are $\hat{c} = R_p + \hat{R}_c$. R_p is predicted from the reconstructed neighbor block, \hat{c} , at the encoder. To reduce the overhead bits as much as possible and improve the compression performance, we use the following method to encoded the lower bit planes. First, for each 4x4 block, the residual coefficients, R , are quantized by QP_l to get R_q^l . The lower bit planes of R_q^l are extracted to calculate the syndrome bits which are transmitted to the decoder. For pixels in the region K , the lower bit planes of R_q^l need not be separately entropy coded since the R_q^l for blocks in the region K can be entropy coded by using the standard H.264 entropy coding method. In this way, we avoid using more overhead bits to encode the lower bit planes of pixels in the region K . For pixels in the region U , the lower bit planes of R_q^l need not be transmitted and are simply discarded. One critical point worthy of note is that the method to reconstruct the approximation of the original pixel value, \hat{c} , is changed to $\hat{c} = R_p + IDCT(IQ(bitshift(R_q^l, -k), QP_h))$ instead of $\hat{c} = R_p + IDCT(IQ(R_q^l, QP_l))$ used in H.264 I frames, where $bitshift()$ is a bit shift function. The reason is that the encoder has to pretend to not know the lower bit planes of R_q^l in the region U and can use only the higher bit planes, which is known at both encoder and the decoder, to guarantee that the decoder is able to recover the same \hat{c} as that in the encoder.

C. Disparity Estimation and Side Information Generation

The region-tree based stereo matching algorithm [16] is used in the decoder to estimate the disparity map between two stereo video sequences. Once we have the disparity map, the side information can be generated by warping the other frame, for example, $I_{L_i}^s = warp(I_{R_i}, D_i)$, where D_i is the disparity map between two frames, I_{L_i} and I_{R_i} . However, not every pixel has a matched pixel due to occlusion. For occluded pixels, we use their neighbor pixel values to estimate their real pixel values. Denote the intra-predicted I-frame as $I_{L_i}^p$ and $I_{R_i}^p$. The side information of the DCT coefficients in the region U can be calculated by transforming and quantizing the residual frame $I_{L_i}^s - I_{L_i}^p$ for the left stream or $I_{R_i}^s - I_{R_i}^p$ for the right stream.

D. LDPC Code Design

The capacity-approaching LDPC codes need to be designed for the virtual correlation channel of bit planes between two correlated coefficients, X and Y . Let b_i^j ($i = 1, 2, \dots, n$, $j = X, Y$) denote the i th bit plane of X or Y . The virtual correlation channel between the i th bit plane can be modeled by the conditional probability mass function $P(b_i^X | b_{i-1}^X, \dots, b_1^X, b_0^X, Y)$. We use a Gaussian channel to approximate the virtual correlation channel between two correlated bit planes. Thus the LDPC code profile for the nonuniform channel that is comprised of a perfect channel and a Gaussian channel is designed by using the Differential Evolution method [14].

E. SNS-SWC Decoding

To successfully decode the symmetric Slepian-Wolf code and recover the lower bit planes of pixels in region U , we need to estimate the initial LLR value of each bit plane of unknown pixels. The most significant bit planes are decoded first and they are used to help decode the least significant bit planes. Suppose that $b_0^X, b_1^X, \dots, b_{i-1}^X$ are known $i - 1$ higher bit planes. To decode an i th bit plane, we need to calculate the value of $\log \frac{P(b_i^X=0 | b_{i-1}^X, \dots, b_1^X, b_0^X, Y)}{P(b_i^X=1 | b_{i-1}^X, \dots, b_1^X, b_0^X, Y)}$, which can be estimated based on the joint statistics of the previous decoded frames.

IV. EXPERIMENTAL RESULTS

We use the Y-component of the 1024x768 Microsoft "Breakdance" multiview video sequence [17] to evaluate the performance of our symmetric distributed multiview video codec. Only video streams from two central views, camera 4 and camera 5, are used. The width of the region B , w_B , is set to 32. We use 10 frames in our experiments. The number of bit planes coded by SNS-SWC is 2. In the experiments, we only consider the symmetric rate allocation case where each video stream has equal rate since the symmetric case is practically more interesting.

We vary the quantization parameters in H.264 to evaluate the performance of the proposed SDVMC. The results are also compared with separate H.264 coding (H.264) and asymmetric Slepian-Wolf coding (ASWC) that uses LDPC codes optimized for a Gaussian Channel. Table I shows the sum bit rates of different coding methods. All three coding schemes have the same peak signal noise ratio (PSNR) when the same quantization parameter is used.

TABLE I
SUM BIT RATE FOR DIFFERENT CODING SCHEMES (KBPS)

Coding Schemes	H264 I Frames	SDVMC-SNS	ASWC
$QP_l = 16$	62003.7	66900.9	66894.8
$QP_l = 10$	115889.9	115206.2	114632.8
$QP_l = 4$	171124.8	168354.4	167514.3

As shown in Table I, in the low-rate case ($QP_l = 16$), SDVMC-SNS cannot save any bits from H.264 I frames. This is because there is not much correlation after residual coefficients are quantized using a high quantization parameter. The asymmetric distributed multiview video coding scheme in [5] produces the similar results. In the high-rate case ($QP_l = 4, 10$), SDVMC-SNS can save 1.62% and 0.59% of total bit rate, respectively. The higher the bit rate, the higher the saving of the total bit rates. This is expected since there is a higher correlation between the residual coefficients of two stereo video streams in the high-rate case. The performance of SDVMC-SNS is consistently inferior to the ASWC. This indicates that there is still significant room to improve the LDPC code design.

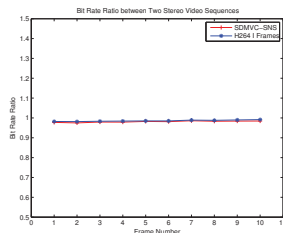


Fig. 5. Bit Ratio between Two Stereo Video Sequences, $QP = 10$.

Fig. 5 shows the rate ratio, $\frac{r_{L_i}}{r_{R_i}}$, of the left video stream and the right video stream in both H.264 and SDVMC-SNS when $QP = 10$. Similar results are attained when the other quantization parameter is used. The ratio measures how close the bit rate of two video streams are in both H.264 and SDVMC-SNS coding schemes. It is evident from the figure that the two streams have almost equal rate. Thus the SDVMC-SNS can achieve the design objective of symmetric rate allocation in two streams.

V. CONCLUSIONS

In this paper, we first propose a syndrome-based symmetric Slepian-Wolf coding scheme that can achieve the whole Slepian-Wolf

region. A generic framework for the symmetric distributed multiview video coding is then put forward. To the best of our knowledge, this is the first work to address the symmetric distributed multiview video coding problem with the help of a symmetric distributed source code. Our preliminary results show that the proposed SDVMC demonstrates a very promising result and moderate rate saving from H.264 I frames can be achieved in the high-rate case.

VI. ACKNOWLEDGMENT

We are grateful to the grant support from Killam Trusts and NSERC (Natural Science and Engineering Research Council of Canada).

REFERENCES

- [1] A. Vetro, Y. Su, H. Kimata, and A. Smolic, "Joint multiview video model jmvvm 2.0," in *ITU-T and ISO/IEC Joint Video Team, Document JVT-U207*, HangZhou, China, 2006.
- [2] Z. Xiong, A. Liveris, and S. Cheng, "Distributed source coding for sensor networks," *IEEE Signal Processing Magazine*, vol. 21, pp. 80–94, 2004.
- [3] 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, September 2007.
- [4] X. Guo, Y. Lu, F. Wu, and Gao W, "Distributed multiview video coding," in *Proc. of Visual Communications and Image Processing (VCIP 2006)*, San Jose, CA, 2006.
- [5] Y. Yang, V. Stankovic, W. Zhao, and Z. Xiong, "Multiterminal video coding," in *Proc. of IEEE International Conference on Image Processing (ICIP 2007)*, San Antonio, TX, USA, 2007.
- [6] V. Thirumalai, I. Tosic, and P. Frossard, "Symmetric distributed coding of stereo omnidirectional image," *Signal Processing: Image Communication*, vol. 23, no. 5, pp. 379–390, June 2008.
- [7] M. Tagliasacchi, G. Prandi, and S. Tubaro, "Symmetric distributed coding of stereo video sequences," in *Proc. of IEEE International Conference on Image Processing (ICIP 2007)*, San Antonio, TX, USA, 2007.
- [8] V. Stankovic, A. D. Liveris, Z. Xiong, and C. N. Georghiadis, "On code design for the slepian-wolf problem and lossless multiterminal networks," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1495–1507, 2006.
- [9] P. Tan and J. Li, "A general and optimal framework to achieve the entire rate region for slepian-wolf coding," *EURASIP Signal Processing Journal, Special Issue on Distributed Source Coding*, vol. 86, no. 11, 2006.
- [10] M. Sartipi and F. Fekri, "Distributed source coding in wireless sensor networks using ldpc coding: The entire slepian-wolf rate region," in *Proc. of the Wireless Communication Networking Conference (WCNC 2005)*, New Orleans, USA, 2005.
- [11] D. Schonberg, *Practical Distributed Source Coding and Its Application to The Compression of Encrypted Data*, Ph.D. thesis, University of California, Berkeley, 2007.
- [12] B. Bai, Y. Yang, P. Boulanger, and J. Harms, "Symmetric distributed source coding using ldpc," in *Proc. of the IEEE International Conference on Communication (ICC 2008)*, Beijing, China, 2008.
- [13] H. Pishro-Nik, N. Rahnavard, and F. Fakri, "Nonuniform error correction using irregular low density parity check codes," *IEEE Transaction on Information Theory*, , no. 7, pp. 2702–2714, 2005.
- [14] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low density parity check codes," *IEEE Transaction on Information Theory*, , no. 2, pp. 619–637, Feb. 2001.
- [15] T. Wiegand, G. Sullivan, G. Bjintegaard, 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.
- [16] C. Lei, J. Selzer, and Y. H. Yang, "Region-tree based stereo using dynamic programming optimization," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2006)*, New York, NY, USA, 2005, pp. 499–406.
- [17] C. L. Zitnick, S. B. King, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," in *Proc. of ACM SIGGRAPH, 2004*, 2004.