Feedback-less Distributed Video Coding and its Application in Compressing Endoscopy Videos

Rami Cohen

8 July, 2012

M.Sc. Research
Supervised by Prof. David Malah

Outline

1. Distributed Video Coding (DVC)
    - Why DVC?
    - Theoretical Background
    - Standard Video Encoders
    - DVC Systems - Overview

Outline

1. Distributed Video Coding (DVC)
    - Why DVC?
    - Theoretical Background
    - Standard Video Encoders
    - DVC Systems - Overview
2. LORD: LOw-complexity, Rate-controlled, Distributed video coding system
    - Motivation
    - Encoder
    - Decoder
    - Adaptation to Endoscopy Video Compression

Outline

1. Distributed Video Coding (DVC)
    - Why DVC?
    - Theoretical Background
    - Standard Video Encoders
    - DVC Systems - Overview
2. LORD: LOw-complexity, Rate-controlled, Distributed video coding system
    - Motivation
    - Encoder
    - Decoder
    - Adaptation to Endoscopy Video Compression
3. Experimental Results

Outline

1. Distributed Video Coding (DVC)
    - Why DVC?
    - Theoretical Background
    - Standard Video Encoders
    - DVC Systems - Overview
2. LORD: LOw-complexity, Rate-controlled, Distributed video coding system
    - Motivation
    - Encoder
    - Decoder
    - Adaptation to Endoscopy Video Compression
3. Experimental Results
4. Conclusion

Why DVC?

- There are cases in which standard (complex) encoders are impractical
- DVC paradigm offers low complexity encoders with good performance

Limited-complexity video encoders: Examples

Theoretical Background

Coding of Correlated Sources

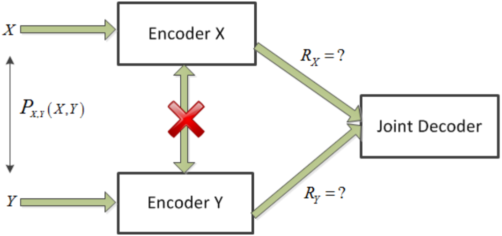
- X and Y are correlated sources
- (x_i, y_i) i.i.d., distributed according to $P_{X,Y}(X, Y)$, where $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$
- When X and Y are jointly encoded and jointly decoded ("conventional" compressing scheme), it is well known that:

$$R_X + R_Y \geq H(X, Y)$$

- $H(X, Y)$ is the *mutual entropy* of X and Y

Theoretical Background

- What if X and Y are *separately* encoded (*distributed coding*) and jointly decoded?



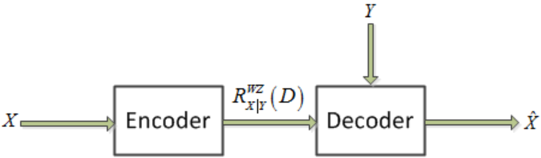
Slepian-Wolf Theorem (1973)

- Surprisingly, given that: $R_X \geq H(X|Y), R_Y \geq H(Y|X)$, Slepian & Wolf have shown that:

$$R_X + R_Y \geq H(X,Y)$$

- Y is referred to as *side information*

Theoretical Background



Wyner-Ziv Theorem (1976)

- When a distortion D is allowed, Wyner and Ziv have shown that:

$$R^{WZ}_{X|Y}(D) \geq R_{X|Y}(D)$$

- Special case in which equality holds:
  1. $X = Y + N$ where N is Gaussian and independent of Y
  2. MSE distortion metric
- The rate loss is bounded [Zamir 98]:

$$R^{WZ}_{X|Y}(D) - R_{X|Y}(D) \leq 0.5\text{bits/sample}$$

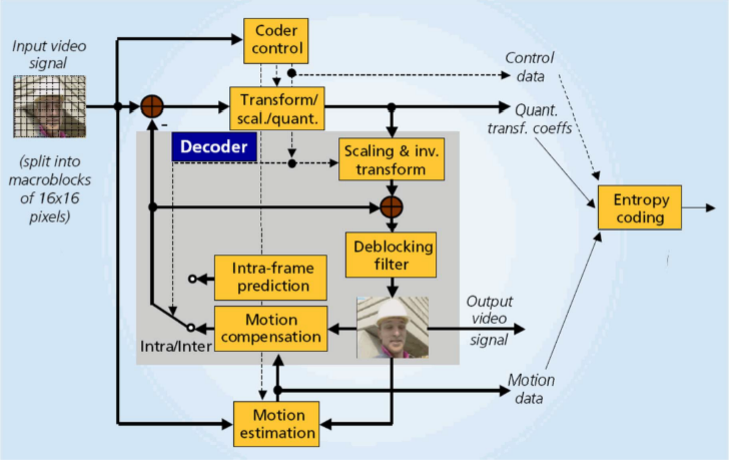
Standard Video Encoders

Hybrid video encoders

- The side information (SI) in modern (*hybrid*) video encoders such as MPEG-2 and H.264 is created by:
  1. Temporal prediction (constituting up to 70% of the encoder's complexity)
  2. Spatial prediction (H.264)
- These video encoders can be viewed as a source coding system with side information available both at the encoder and the decoder
- *Master-Slave*: Complex encoder, simple decoder
- Impractical in power or resources limited encoders

Standard Video Encoders

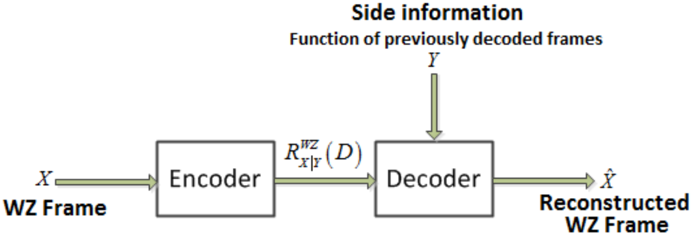
State-of-the-art: H.264 Scheme



DVC Systems - Overview

DVC System

- Together, the Slepian-Wolf and the Wyner-Ziv theorems suggest that it is possible to compress video in a distributed way
    - X denotes the current frame and Y denotes its prediction, which is created *at the decoder*
- Approaching (theoretically) the coding efficiency of conventional predictive coding schemes



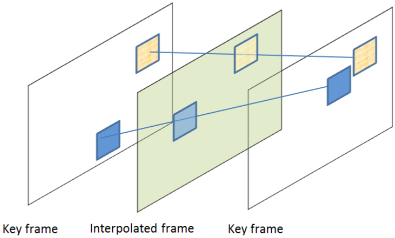
DVC Systems - Overview

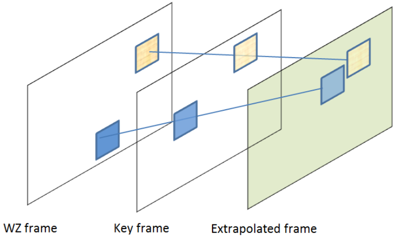
Main Parts

- Usually, the input is separated into *key* (intra-coded) frames and Wyner–Ziv (WZ) frames
- Side information (SI) creation: prediction (Y) of the frame to be encoded (X), is created *at the decoder*
    - Block matching
    - Motion interpolation/extrapolation
- Noise correlation model: estimating X from Y
    - Probability distribution models for $N = X - Y$, such as Laplace and Gamma distributions (usually in the frequency domain)
    - Offline/online estimation of parameters

DVC System - Overview

SI Creation Example: Interpolation/Extrapolation





Motivation for developing LORD

Adaptation to the video statistics

- on-line estimation of the parameters of the noise model

Rate control

- Not affected by the decoder
- Suitable for channels with constant rate constraint

Low delay

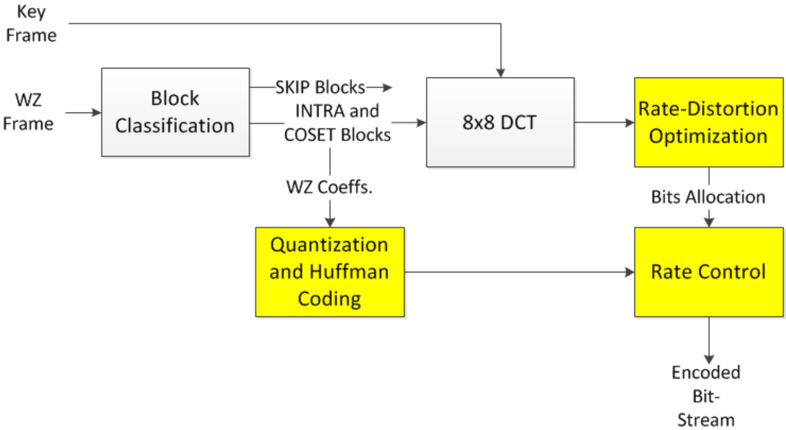
- No feedback-channel

Medical application

- Adaptation to the compression of endoscopy videos

LORD: Encoder

Scheme



Encoder

GOP

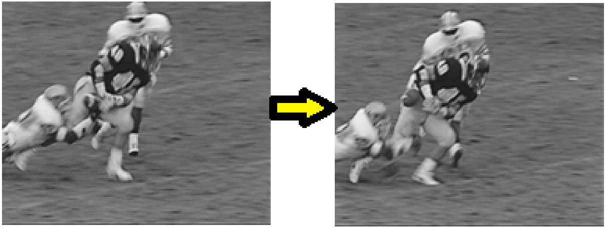
- Group of Pictures (GOP) of size 2 is used
- IW structure: the first one is intra-coded, the second is a WZ frame

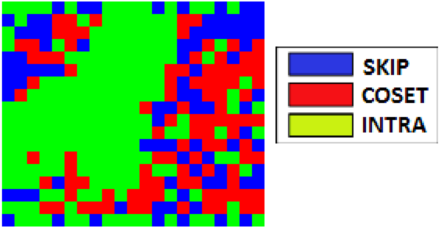
Blocks Classification

- The energy E_d of the differences between co-located blocks in the frames of the GOP is used
- SKIP mode: $E_d \leq SKIP_{TH}$
  - Not coded, the co-located block is copied
- INTRA mode: $E_d \geq INTRA_{TH}$
  - Coded using JPEG
- COSET mode: $SKIP_{TH} < E_d < INTRA_{TH}$
  1. First 15 AC coefficients are coded using DVC principles
  2. Remaining coefficients are coded using JPEG

Encoder

Blocks Classification: Example

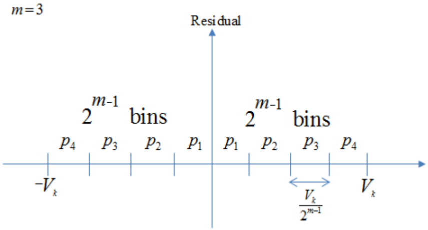




Encoder

COSET mode

- The maximal (over COSET blocks) absolute differences $\{V_k\}$ between co-located 15 AC coefficients (*WZ coefficients*) in the GOP are sent losslessly
- The differences between co-located the WZ coeffs. are quantized uniformly to symmetric 2^m levels, using $\{V_k\}$
- The quantization indices are sent using Huffman code
    - Huffman dictionary is built offline



Index	Codeword
1	01
2	00
3	101
4	100
5	1101
6	1100
7	1111
8	1110

Encoder

Rate-Distortion Optimization

- We have $P = 128$ *DCT bands* in the GOP
- They are modelled as P random variables $X_1, X_2, ..., X_P$ with zero mean and variances σ_i^2 $(i = 1, 2, ..., P)$
- The distortion (measured as MSE) incurred when uniformly quantizing X_i using b_i bits is modelled as:

$$D_i\left(b_i\right) = h_i \sigma_i^2 2^{-2b_i}$$

- h_i is determined according to the PDF of X_i
- The total distortion in the GOP is:

$$D = \underbrace{\sum_{Realizations} h_i \sigma_i^2 2^{-2b_i}}_{\text{distortion from key frame}} + \underbrace{\sum_{Realizations} h_i \sigma_i^2 2^{-2b_i}}_{\text{distortion from WZ frame}}$$

Encoder

Rate-Distortion Optimization

- Written more compactly, we get:

$$D = \sum_{i=1}^{P} m_i h_i \sigma_i^2 2^{-2b_i}$$

- m_i is the number of intra-coded coefficients in the i^{th} band
- σ_i^2 is calculated using the maximum-likelihood (ML) estimator
  1. Coefficients in the DC bands are assumed to be Gaussian-distributed
  2. Coefficients in the AC bands are assumed to be Laplace-distributed

$$\sigma_G^2 = \frac{1}{N_G} \sum_{j=1}^{N_G} x_j^2, \quad \sigma_L^2 = 2\left(\frac{1}{N_L} \sum_{j=1}^{N_L} |x_j|\right)^2$$

Encoder

Rate-Distortion Optimization

- Assuming that the available number of bits for encoding the GOP is B, the resulting optimization problem is:

$$\min_{\mathrm{b}_i} D = \sum_{i=1}^{P} m_i h_i \sigma_i^2 2^{-2b_i}, \quad \text{s.t.} \sum_{i=1}^{P} b_i \leq B$$

- The solution (obtained using Lagrange multipliers):

$$b_i = \bar{b} + \frac{1}{2}\log_2 \frac{\sigma_i^2}{\rho^2} + \frac{1}{2}\log_2 \frac{h_i}{H} + \frac{1}{2}\log_2 \frac{m_i}{M}$$

$$\bar{b} = \frac{B}{P}, \rho^2 = \left(\prod_{i=1}^{P} \sigma_i^2\right)^{1/P}, H = \left(\prod_{i=1}^{P} h_i\right)^{1/P}, M = \left(\prod_{i=1}^{P} m_i\right)^{1/P}$$

Encoder

Rate Control

- Once the bit distribution among the GOP is determined, it is enforced using a rate control (RC) algorithm
- We employ the linear relationship between the coding bit rate R and the fraction ρ of zeros among the quantized intra-coded coefficients [He & Mitra, 2002]:
$$R\left(\rho\right) = \theta\left(1 - \rho\right)$$
- θ is a constant related to the image content
- The number of zeros is controlled by the parameter q used in JPEG, which determines the quantization step $step_i\left(q\right)$ $\left(i = 1, 2, ..., 64\right)$

Encoder

Rate Control - Implementation

- ρ is determined using q:

$$\rho\left(q\right)=\frac{1}{N}\sum_{i}\sum_{j:\left|x_{i,j}\right|\leq step_{i}(q)}1$$

- The relation between R and ρ is maintained through an adaptive estimation of θ
- Denote:
  1. M the number of the blocks in the current frame
  2. N_m the number of already coded blocks
  3. B_m the number of bits used for encoding these N_m blocks
  4. S the number of INTRA coefficients in each block

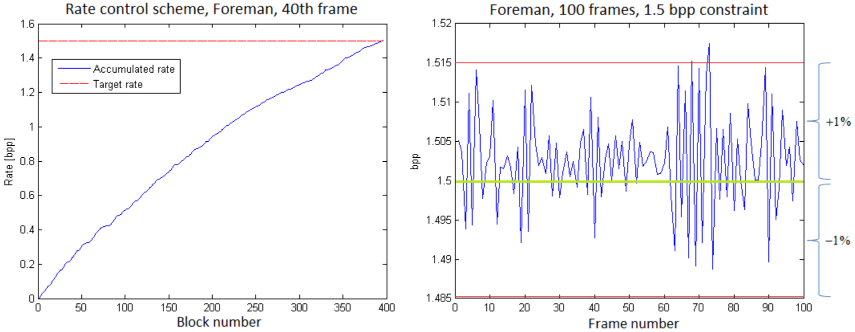
Encoder

RC - Practical Implementation

1. Set $N_m = \eta_m = B_m = 0$, $\theta = 6.5$
2. The number of zeros to be produced by quantizing the remaining blocks is:
$$\eta = S \cdot (M - N_m) - \frac{B - B_m}{\theta}$$
using η, calculate $q(\eta)$
3. Let η_0 and B_0 denote the number of zeros and the number of bits produced by the current block, respectively. set:
$$\eta_m := \eta_m + \eta_0, B_m := B_m + B_0, N_m := N_m + 1$$
and update θ according to: $\theta = \frac{B_m}{S \cdot N_m - \eta_m}$
4. Repeat stages 2,3 until all the blocks are encoded

Encoder

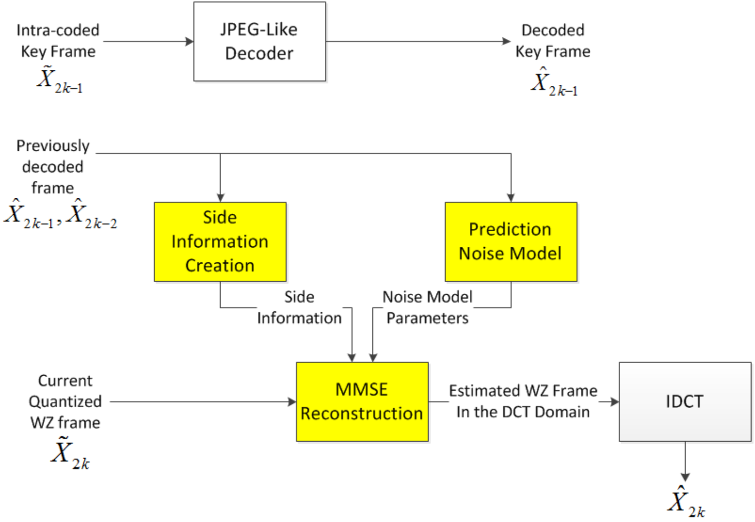
RC algorithm: Example



1. One pass, low-complexity algorithm

LORD: Decoder

Scheme



Intra-coded Key Frame $\tilde{X}_{2k-1}$

JPEG-Like Decoder

Decoded Key Frame $\hat{X}_{2k-1}$

Previously decoded frame $\hat{X}_{2k-1}, \hat{X}_{2k-2}$

Side Information Creation

Prediction Noise Model

Side Information

Noise Model Parameters

Current Quantized WZ frame $\tilde{X}_{2k}$

MMSE Reconstruction

Estimated WZ Frame In the DCT Domain

IDCT

$\hat{X}_{2k}$

Decoder

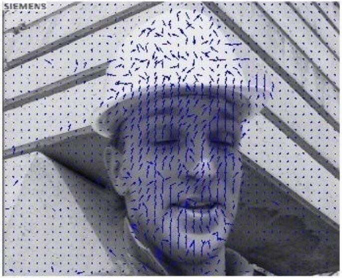
SI Creation: 1. Motion Estimation

- Qpel full search motion estimation is performed between two already decoded frames, $\hat{X}_{2k-2}$ and $\hat{X}_{2k-1}$
- Qpel precision is obtained in $\hat{X}_{2k-2}$ using H.264 interpolation filter (over $\hat{X}_{2k-2}$):

$$h = \left[\begin{array}{ccccccccccc} 1 & 0 & -5 & 0 & 20 & 32 & 20 & 0 & -5 & 0 & 1 \end{array}\right] /32$$

$\hat{X}_{2k-2}$

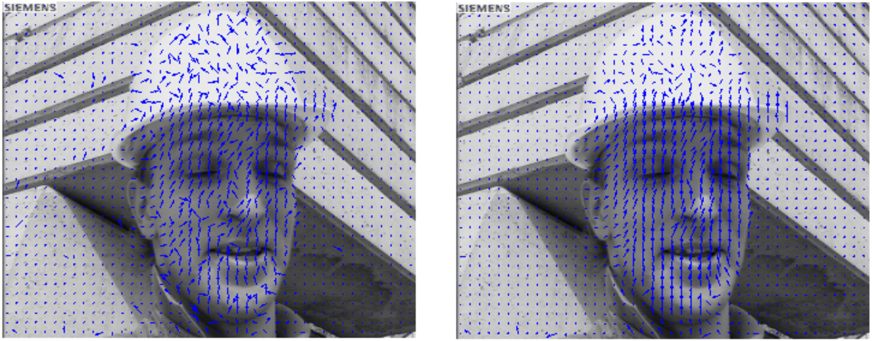
$\hat{X}_{k-1}$



Decoder

SI Creation: 1. Motion Estimation (Cont.)

- The motion field is smoothed by replacing each MV is by the median of its 4 closest MVs

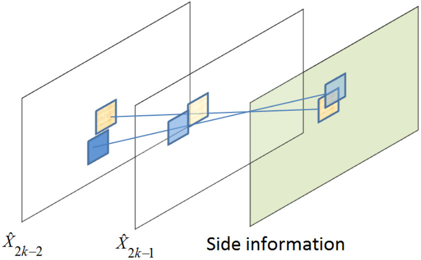


Before smoothing **After smoothing**

Decoder

SI Creation: 2. Motion Extrapolation

1. Assuming linear motion, the pixels from $\hat{X}_{2k-1}$ are projected to the next (extrapolated) frame, which is used as side information
2. The previous stages are repeated with different offsets of $\hat{X}_{2k-1}$ from the upper-left corner, denoted by (o_x, o_y)
   - If there are multiple predictions, their average is used
   - If there are pixels with no predictor, spatial interpolation is used



Decoder

Motion Extrapolation: Example

ipel precision, PSNR: 20.2dB

qpel precision, PSNR: 24.9dB

- Better performance than integer pixel (ipel) based motion extrapolation algorithms, by 3-4dB on average

Decoder

Prediction Noise Model

- The noise (N) between $\hat{X}_{2k-2}$ and $\hat{X}_{2k-1}$ serves as an estimate of the noise between the SI (Y) and the WZ frame (X)
- N is assumed to be Laplace-distributed:

$$f_{X|y}(x) = f_N(x-y) = \frac{\alpha}{2} e^{-\alpha|x-y|}$$

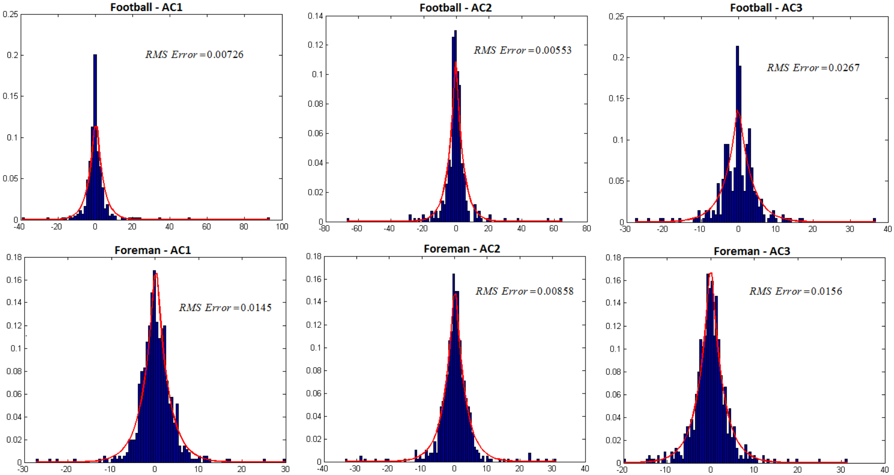
- α is calculated for each band, using the ML estimator:

$$\alpha_i = \left(\frac{1}{K_i} \sum_{j=1}^{K_i} |x_j| \right)^{-1}$$

  - K_i is the number of the samples in the i^{th} band, and x_j ($j = 1, 2, ..., K_i$) are the samples

Decoder

Prediction Noise Model (Cont.)

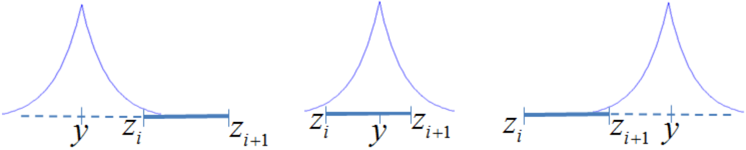


Decoder

MMSE Reconstruction

- We get an MMSE estimate of the source X, using the quantization interval $[z_i, z_{i+1})$ and the side information Y (for $x \in X$ and $y \in Y$):

$$\hat{x} = \mathbb{E}\left[x \middle| \, x \in [z_i, z_{i+1}), y\right] = \frac{\int\limits_{z_i}^{z_{i+1}} x f_{X|y}(x)\, dx}{\int\limits_{z_i}^{z_{i+1}} f_{X|y}(x)\, dx}$$



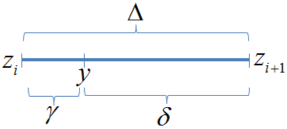
Decoder

MMSE Reconstruction (Cont.)

- The last integrals can be carried out analytically, resulting in:

$$\hat{x} = \begin{cases} z_i + \frac{1}{\alpha} + \frac{\Delta}{1 - e^{\alpha\Delta}} & \text{if } y < z_i \\ y + \frac{\left(\gamma + \frac{1}{\alpha}\right) e^{-\alpha\gamma} - \left(\delta + \frac{1}{\alpha}\right) e^{-\alpha\delta}}{2 - \left(e^{-\alpha\gamma} + e^{-\alpha\delta}\right)} & \text{if } y \in [z_i, z_{i+1}) \\ z_{i+1} - \frac{1}{\alpha} - \frac{\Delta}{1 - e^{\alpha\Delta}} & \text{if } y \geq z_{i+1} \end{cases}$$

- Δ, γ, δ are defined according to:



Decoder

MMSE Reconstruction (Cont.)

- If the noise conveys no information ($\alpha \to 0 \Rightarrow \sigma^2 \to \infty$):

$$\lim_{\alpha \to 0} \hat{x} = \begin{cases} z_i + \dfrac{\Delta}{2} & \text{if } y < z_i \\ y - \dfrac{\gamma - \delta}{2} & \text{if } y \in [z_i, z_{i+1}) \\ z_{i+1} - \dfrac{\Delta}{2} & \text{if } y \geq z_{i+1} \end{cases}$$

- If the noise is highly localized ($\alpha \to \infty \Rightarrow \sigma^2 \to 0$):

$$\lim_{\alpha \to \infty} \hat{x} = \begin{cases} z_i & \text{if } y < z_i \\ y & \text{if } y \in [z_i, z_{i+1}) \\ z_{i+1} & \text{if } y \geq z_{i+1} \end{cases}$$

Endoscopy videos

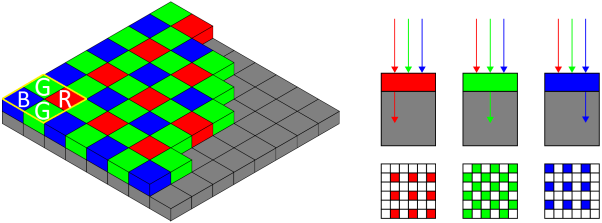
Endoscopy

- Endoscopy refers to looking inside the body for medical reasons using an endoscope
- An endoscope is consisted of a long, thin, flexible tube that has a light source and an attached camera
- Recently, a shift towards transmission of endoscopy videos over a wireless channel - limited power resources

Endoscopy Videos

Bayer Filter

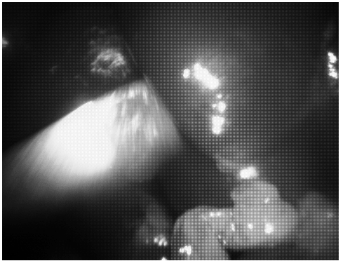
- Bayer color filter array (CFA) is composed of filter blocks of size 2x2, which are 50% green, 25% red and 25% blue
- Conforms with the strong sensitivity of the human vision system (HVS) to green light
- Each physical pixel has an optical filter placed over it, allowing penetration of only particular color of light (red, green or blue)
- Almost universal on consumer digital cameras, used in endoscopes



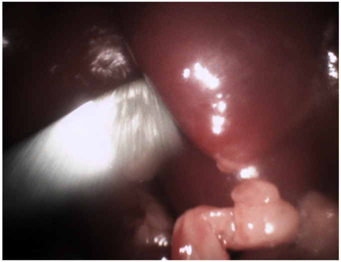
Endoscopy Videos

Bayer Filter

- Bayer *demosaicing* is the process of translating a Bayer image into a full color (RGB) image



Raw image



Demosaiced image

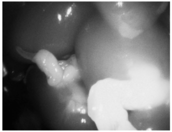
From video provided by Gyrus ACMI, Inc.

Endoscopy Videos

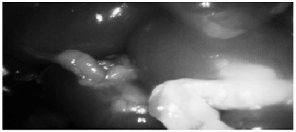
RGB Separation

- Each color component is compressed separately
- Exploiting the correlation between pixels of the same color

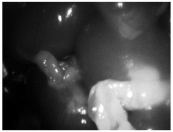
Raw Bayer



R component



G component

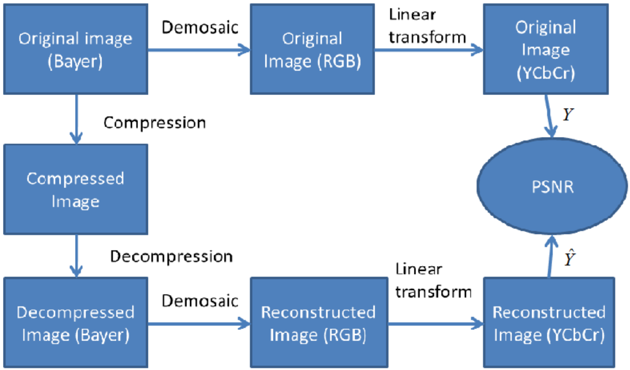


B component

Endoscopy Videos

Rate-Distortion Optimization

- The PSNR calculation for Bayer format is taken into account for RDO



Endoscopy Videos

Rate-Distortion Optimization

- The relation between Y and RGB components is (ITU-R BT.601):

$$Y = w_R R + w_G G + w_B B$$

where: $w_R = 0.299, w_G = 0.587, w_B = 0.114$

- Assuming that the reconstruction error is mainly due to the error between color components of the same type, we get:

$$\mathbb{E}\left[\left(Y - \hat{Y}\right)^2\right] = \mathbb{E}\left[\left(w_R\left(R - \hat{R}\right) + w_G\left(G - \hat{G}\right) + w_B\left(B - \hat{B}\right)\right)^2\right]$$

$$\approx w_R^2 \cdot \mathbb{E}\left[\left(R - \hat{R}\right)^2\right] + w_G^2 \cdot \mathbb{E}\left[\left(G - \hat{G}\right)^2\right] + w_B^2 \cdot \mathbb{E}\left[\left(B - \hat{B}\right)^2\right]$$

Endoscopy Videos

Rate-Distortion Optimization

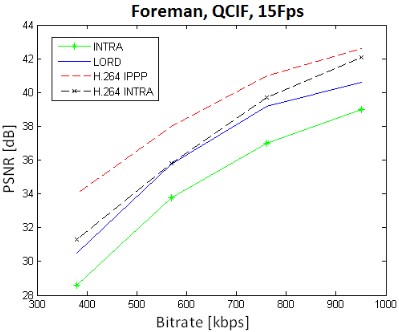
- The distortion is calculated separately for each color components
- Each distortion is weighted according to w_C^2 $(C = R, G, B)$:

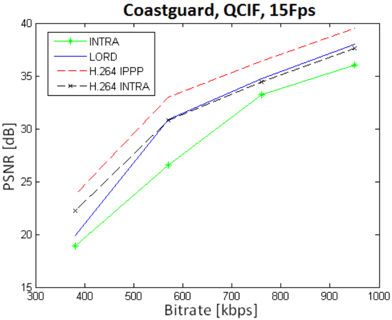
$$D = \underbrace{w_R^2 \cdot \sum_{\substack{DCT \\ bands}} m_{R_i} h_i \sigma_i^2 2^{-2b_i}}_{\text{distortion from R component}} + \underbrace{w_G^2 \cdot \sum_{\substack{DCT \\ bands}} m_{G_i} h_i \sigma_i^2 2^{-2b_i}}_{\text{distortion from G component}} + \underbrace{w_B^2 \cdot \sum_{\substack{DCT \\ bands}} m_{B_i} h_i \sigma_i^2 2^{-2b_i}}_{\text{distortion from B component}}$$

- Considering the available bits B, we get an optimization problem
- The solution is a simple extension of the previous one

Standard Videos

PSNR Results

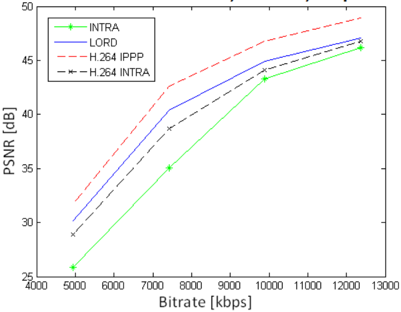




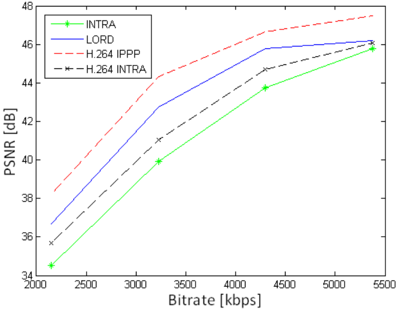
Endoscopy Videos

PSNR Results

Simulation - chicken, 368x480, 28Fps



Simulation - gastrointestinal tract, 240x320, 28Fps



Summary

- New DVC codec was developed
- On-line estimation of the parameters of the noise model
- Rate-distortion model and rate control algorithm are used, at the encoder
- No feedback channel is used
- Adaptation to endoscopy videos (Bayer format)
- Improvement over standard intra coding, for both standard videos and endoscopy videos

Future Work

- Improved localization of the noise model
- Side information creation for videos with non-linear motion
- Dynamic decision on coding modes
- De-correlation of the RGB components in a Bayer frame and using an appropriate distribution model for the de-correlated components


Thank You