UC Berkeley
EECS

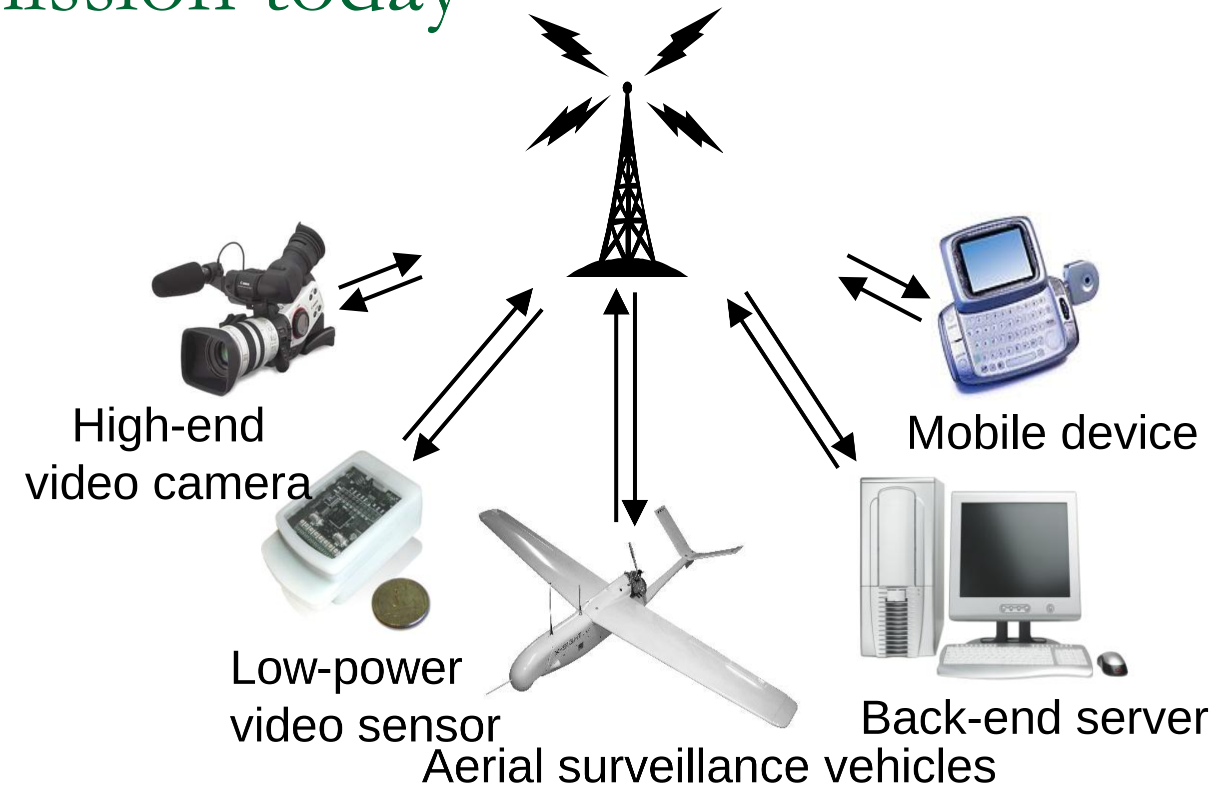
The BASiCS Group
Berkeley Audio-visual Signal Processing and Communication Systems

Perspectives in distributed source coding

Kannan Ramchandran

UC Berkeley

Media transmission today



Challenges

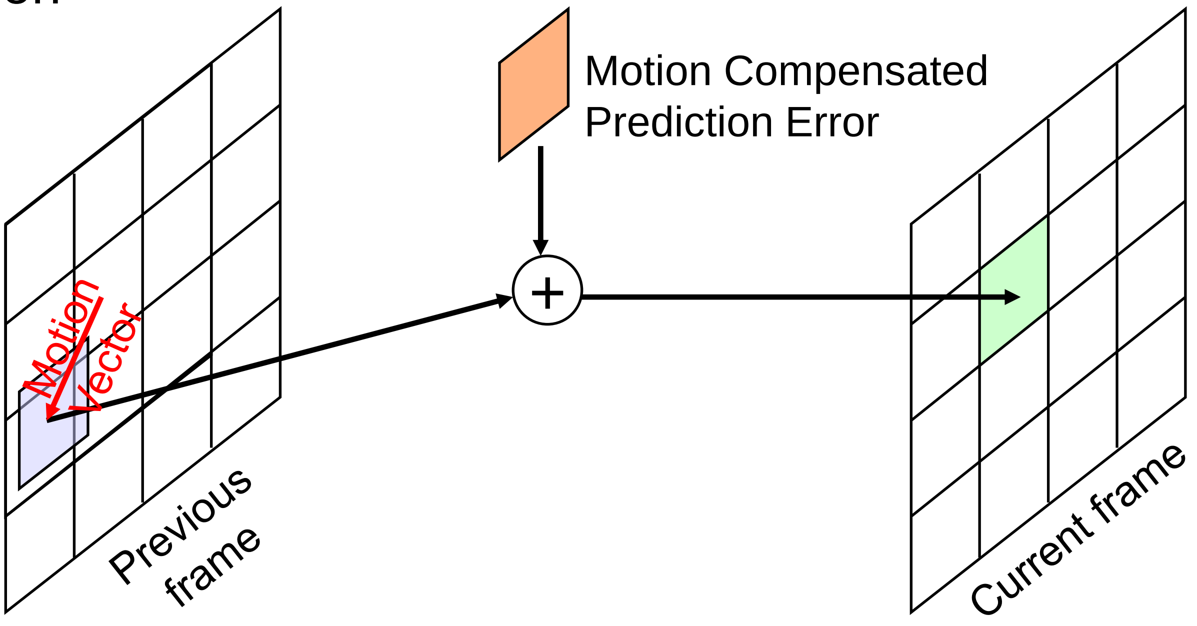
- High compression efficiency
- High resilience to transmission errors
- Flexible encoder/decoder complexity distribution
- Low latency

How to meet these requirements *simultaneously?*

Today's video codec systems

Driven by downlink model:

- High compression efficiency
- Rigid complexity distribution
  - Complex transmitter, light receiver
- Prone to transmission error
  - Decoding relies **deterministically** on one predictor



Rethink video codec architecture?

- Alternatives to **rigid complexity** partition, **deterministic** prediction-based framework?

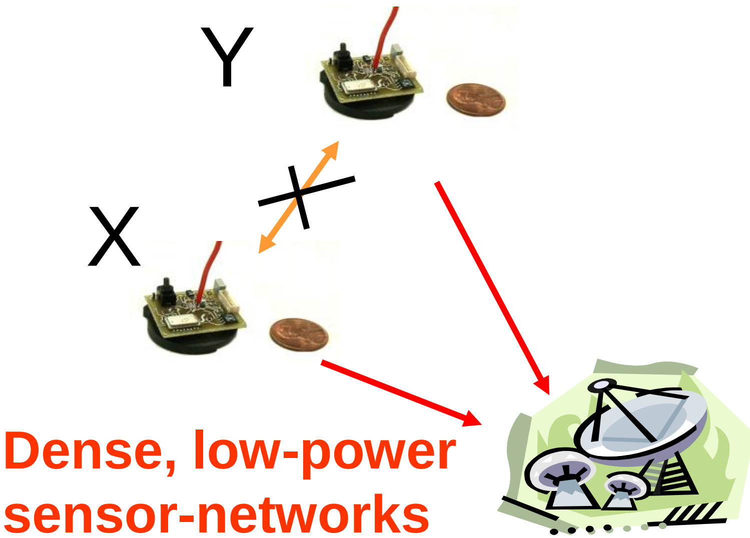
- Interesting tool: **distributed source coding**

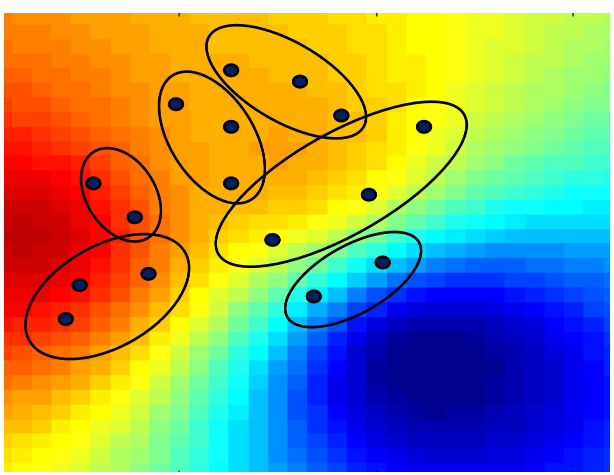
Roadmap

- Introduction and motivation
- Distributed source coding: foundations & intuition
- Application landscape

- Distributed source coding for video applications:
  - Encryption & Compression
  - Video transmission: foundations and architecture
      - Low-encoder-complexity
      - High-compression efficiency + Robustness
    - Multi-camera scenario

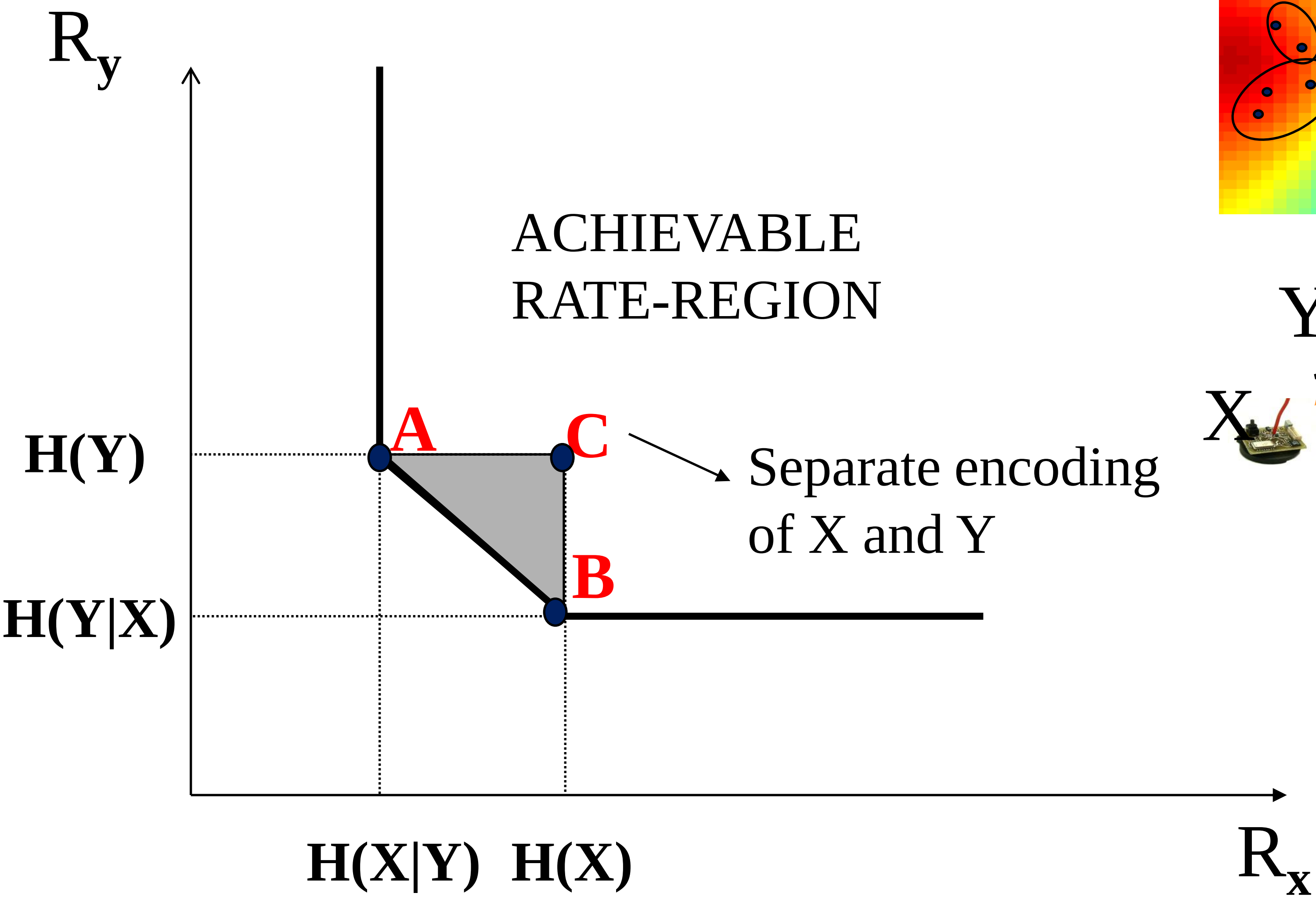
Motivation: sensor networks

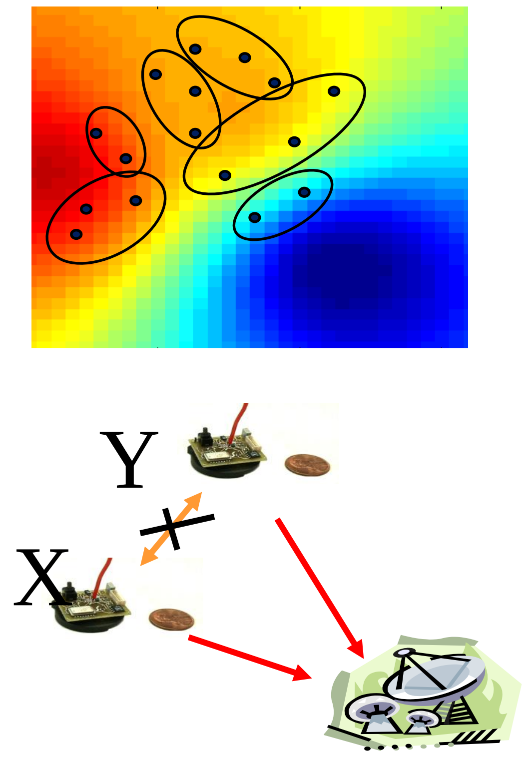




- Consider correlated nodes X, Y
- Communication between X and Y expensive.
- Can we exploit correlation without communicating?
- Assume Y is compressed independently. How to compress X close to $H(X|Y)$?
- Key idea: discount $I(X;Y)$. $H(X|Y) = H(X) - I(X;Y)$

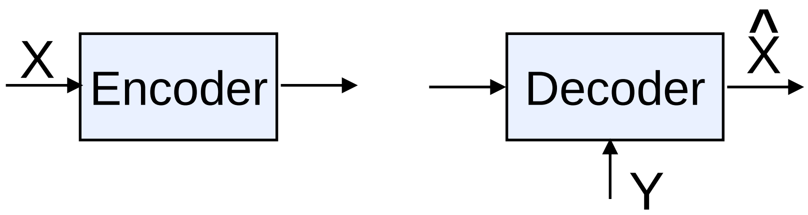
Distributed source coding: Slepian-Wolf '73





Distributed source coding

Source coding with side information: (Slepian-Wolf, '73, Wyner-Ziv, '76)



- X and Y are correlated sources.
- Y is available only to decoder.

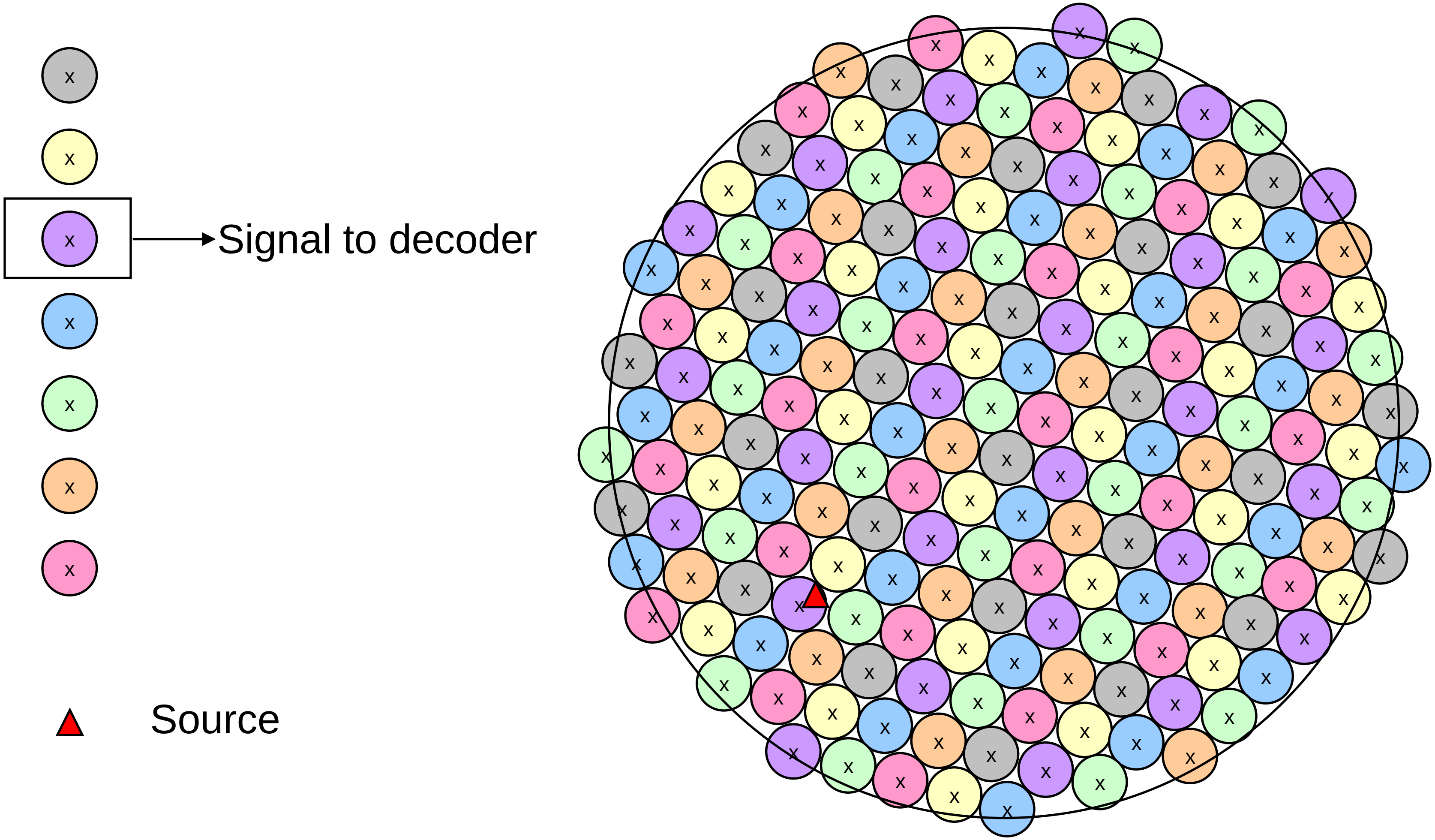
- ***Lossless coding (S-W):*** **no loss of performance over when Y is available at both ends if the statistical correlation between X and Y is known*.***

- ***Lossy coding (W-Z):*** **for Gaussian statistics, no loss of performance over when Y known at both ends*.***

- **Constructive solutions:** *(Pradhan & Ramchandran (DISCUS) DCC '99 , Garcia-Frias & Zhao Comm. Letters '01, Aaron & Girod DCC '02, Liveris, Xiong & Georghiades DCC '03,…)*

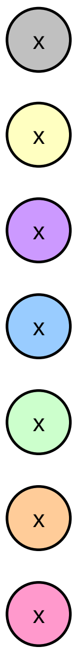
- **Employs statistical instead of deterministic** mindset.

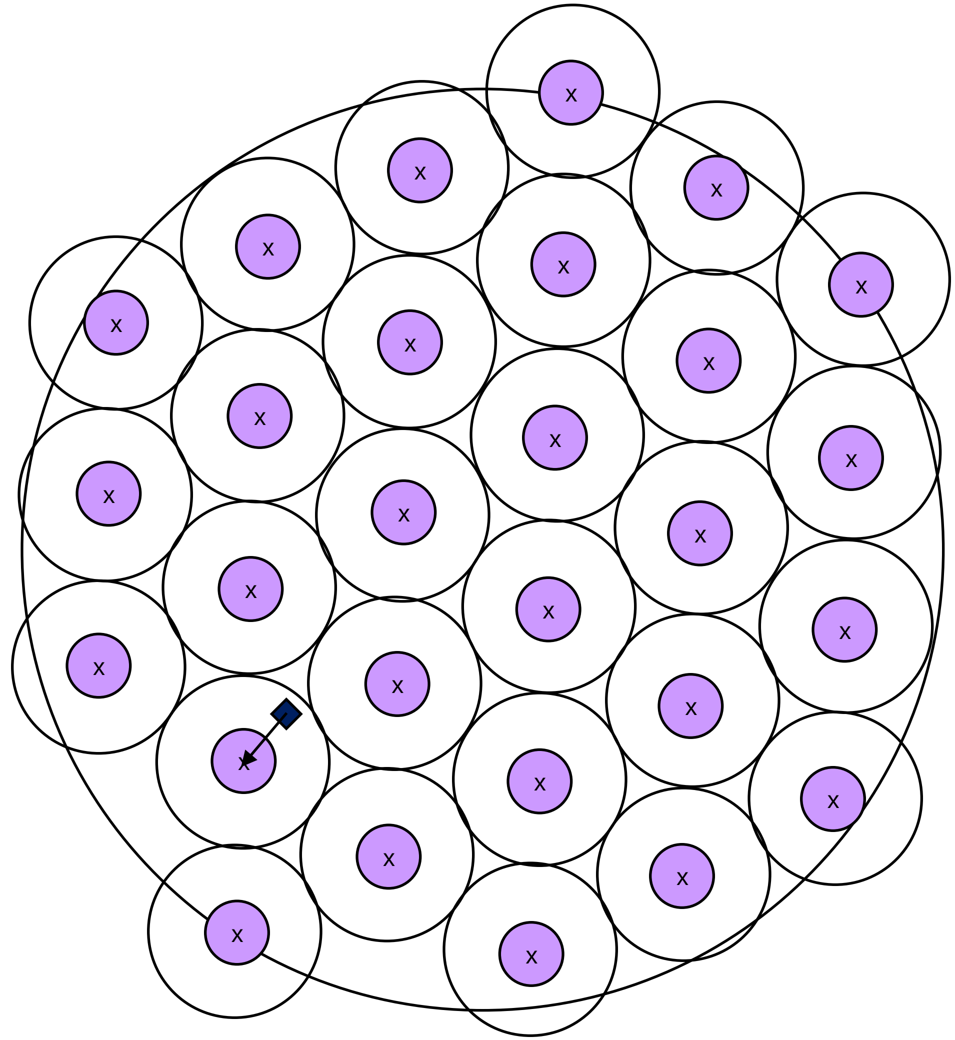
Example: geometric illustration



Assume signal and noise are Gaussian, iid

Example: geometric illustration



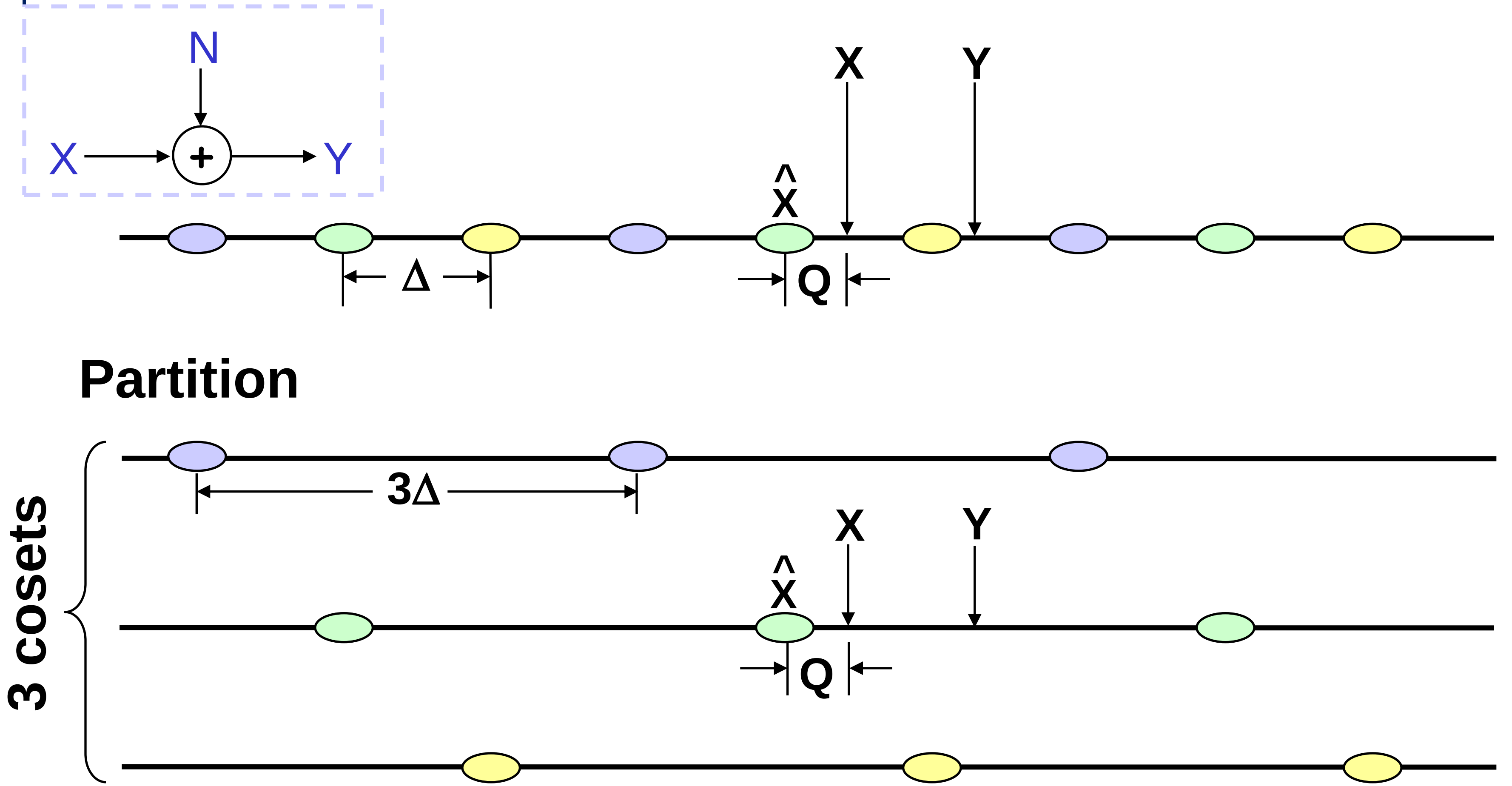


Source

Side information

Assume signal and noise are Gaussian, iid

Example: scalar Wyner-Ziv



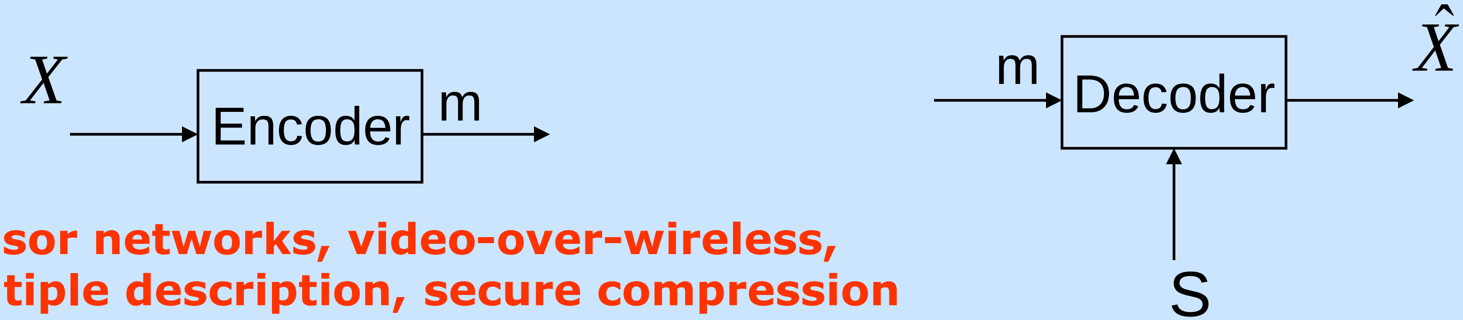
- Encoder: send the index of the coset ($\log_2 3$ bits)
- Decoder: decode X based on Y and signaled coset

Application Landscape

- **Sensor networks**
- M-channel **Multiple Description coding**
- **Media broadcast**
- **Media security**: Data-hiding, watermarking, steganography
  - Fundamental duality between source coding and channel coding with side-information
- **Compression of encrypted data**
- **Video transmission**

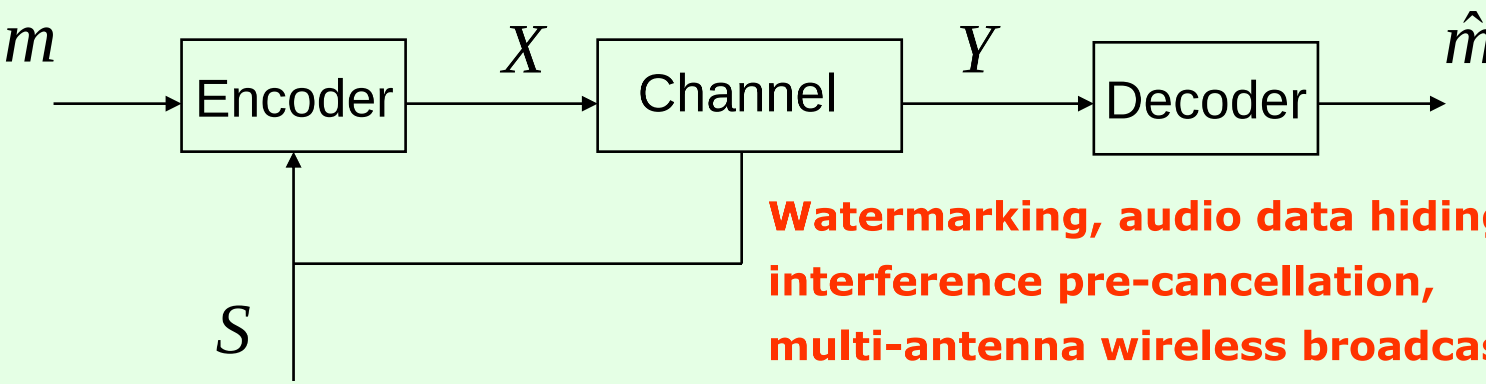
Duality bet. source & channel coding with side-info

Source coding with side information



Sensor networks, video-over-wireless, multiple description, secure compression

Channel coding with side information (CCSI)



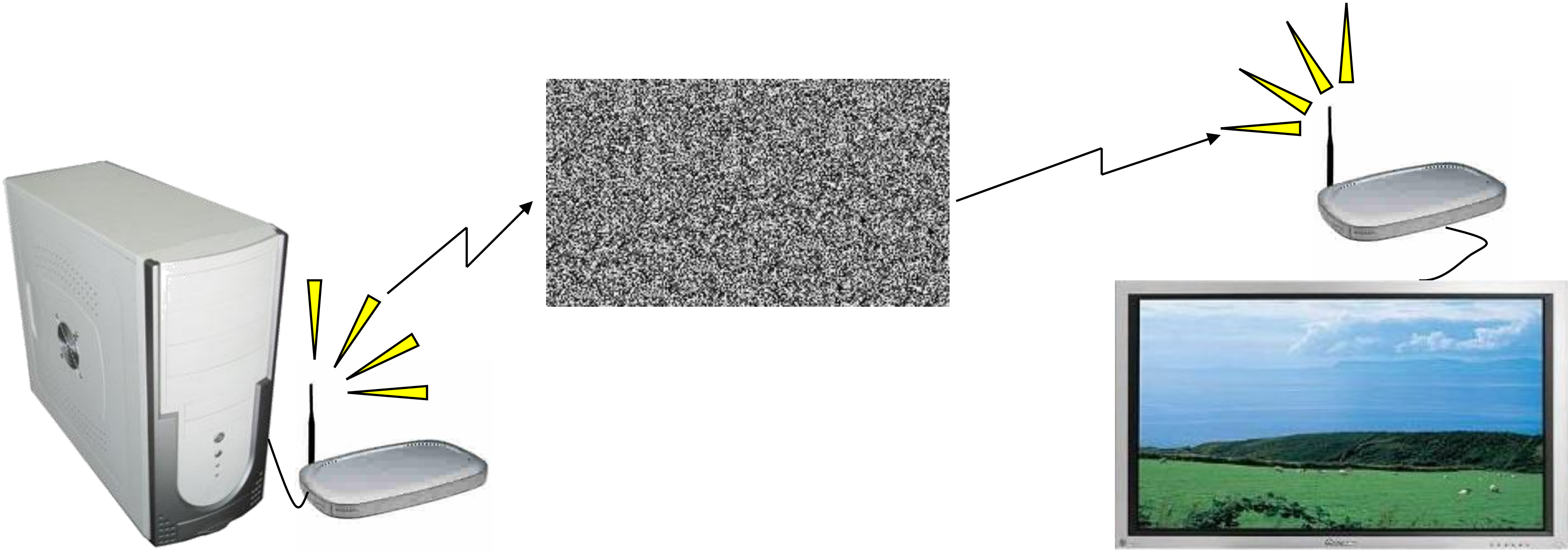
Watermarking, audio data hiding, interference pre-cancellation, multi-antenna wireless broadcast.

Pradhan, Chou and Ramchandran, Trans. on IT, May 2003

Compressing encrypted content without the cryptographic key

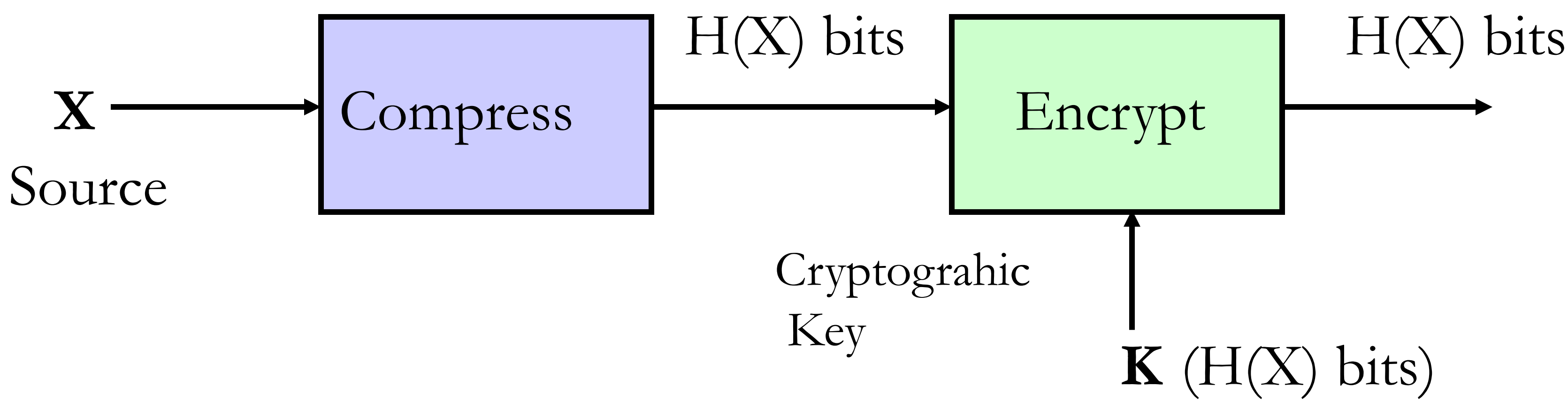
Secure multimedia for home networks

- Uncompressed encrypted video (HDCP protocol)
  - Can increase wireless range with lower data rate
  - But how to compress encrypted video **without access to crytpographic key**?

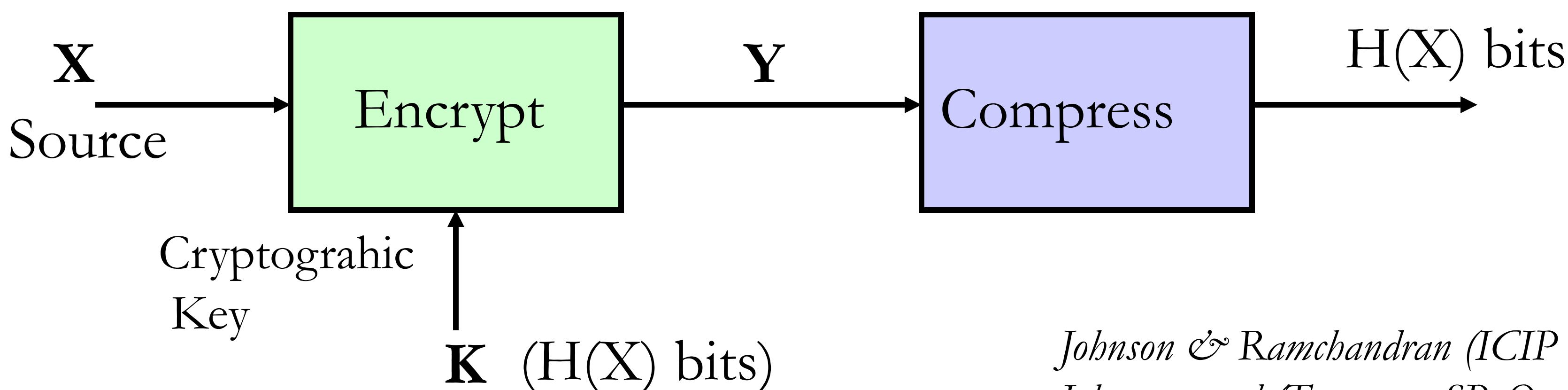


Application: Compressing Encrypted Data

Conventional method:



Unconventional method:



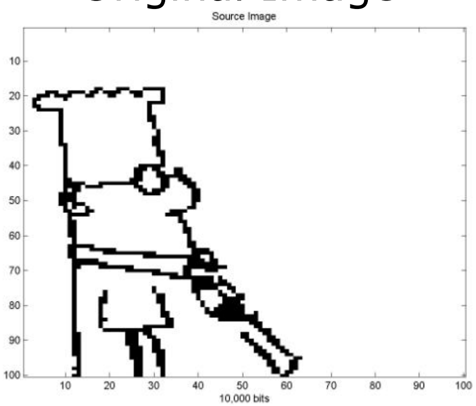
Johnson & Ramchandran (ICIP 2003), Johnson et. al (Trans. on SP, Oct. 2004)

Example

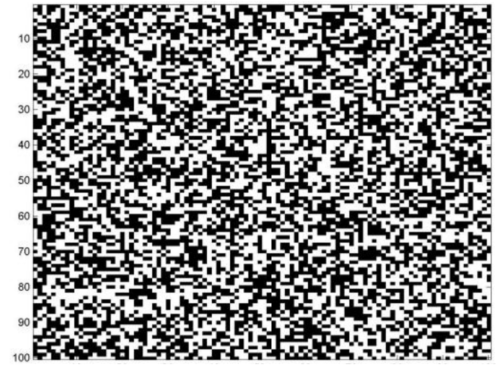
10,000 bits

5,000 bits

Original Image

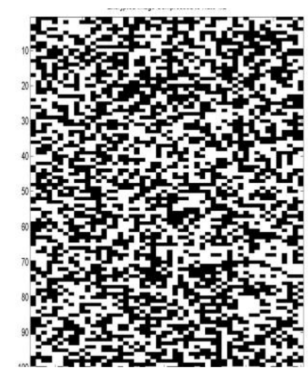


Encrypted Image

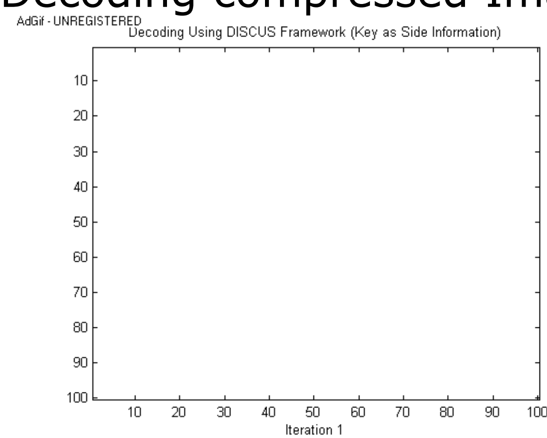


Compressed

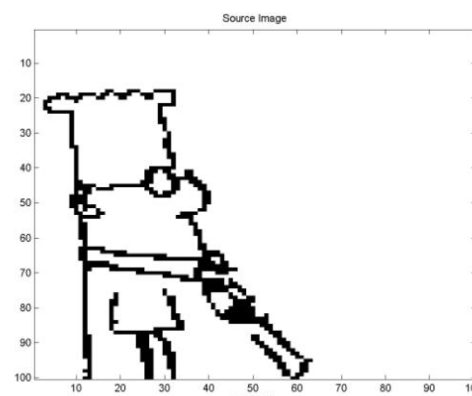
Encrypted Image



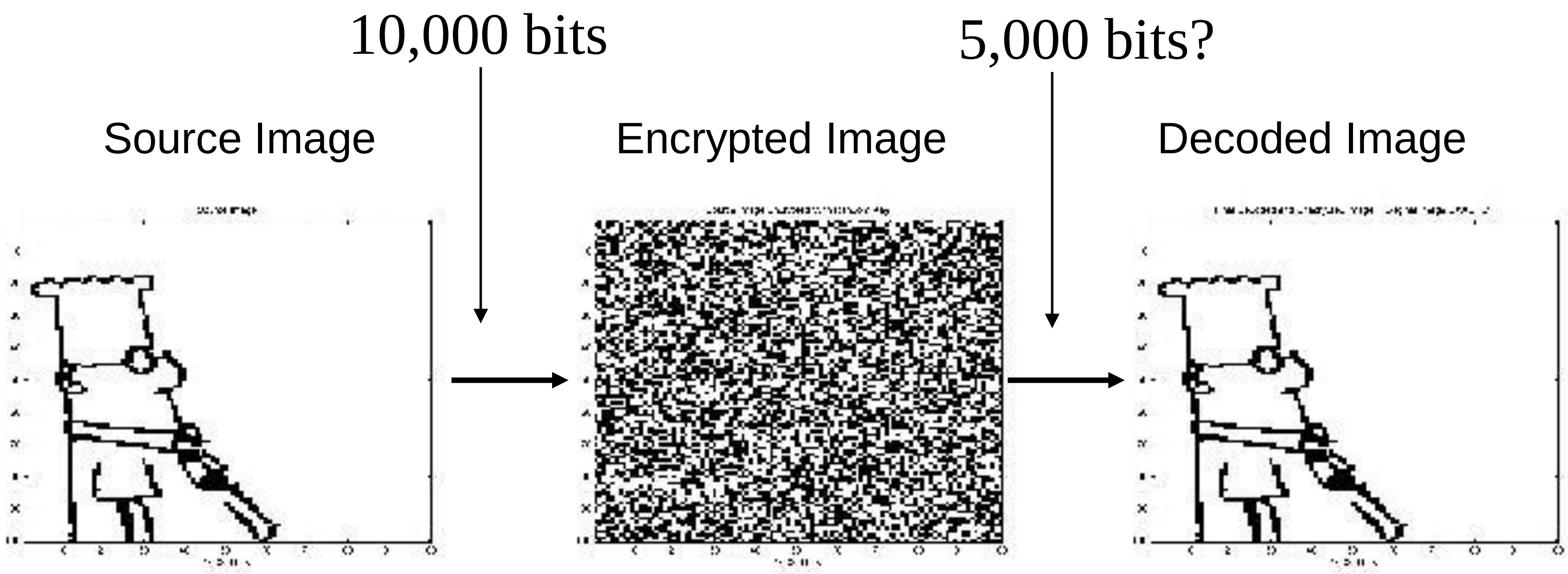
Decoding compressed Image



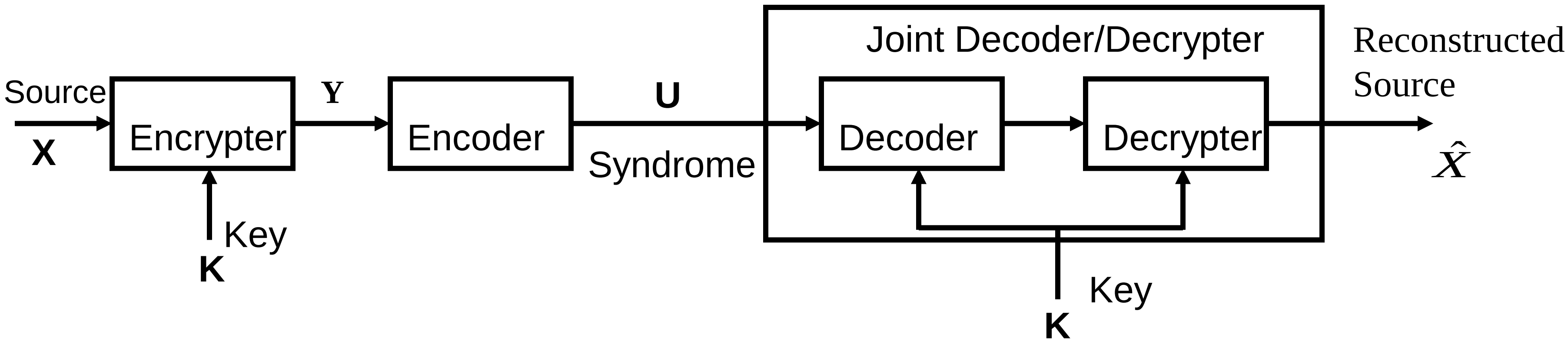
Final Reconstructed Image



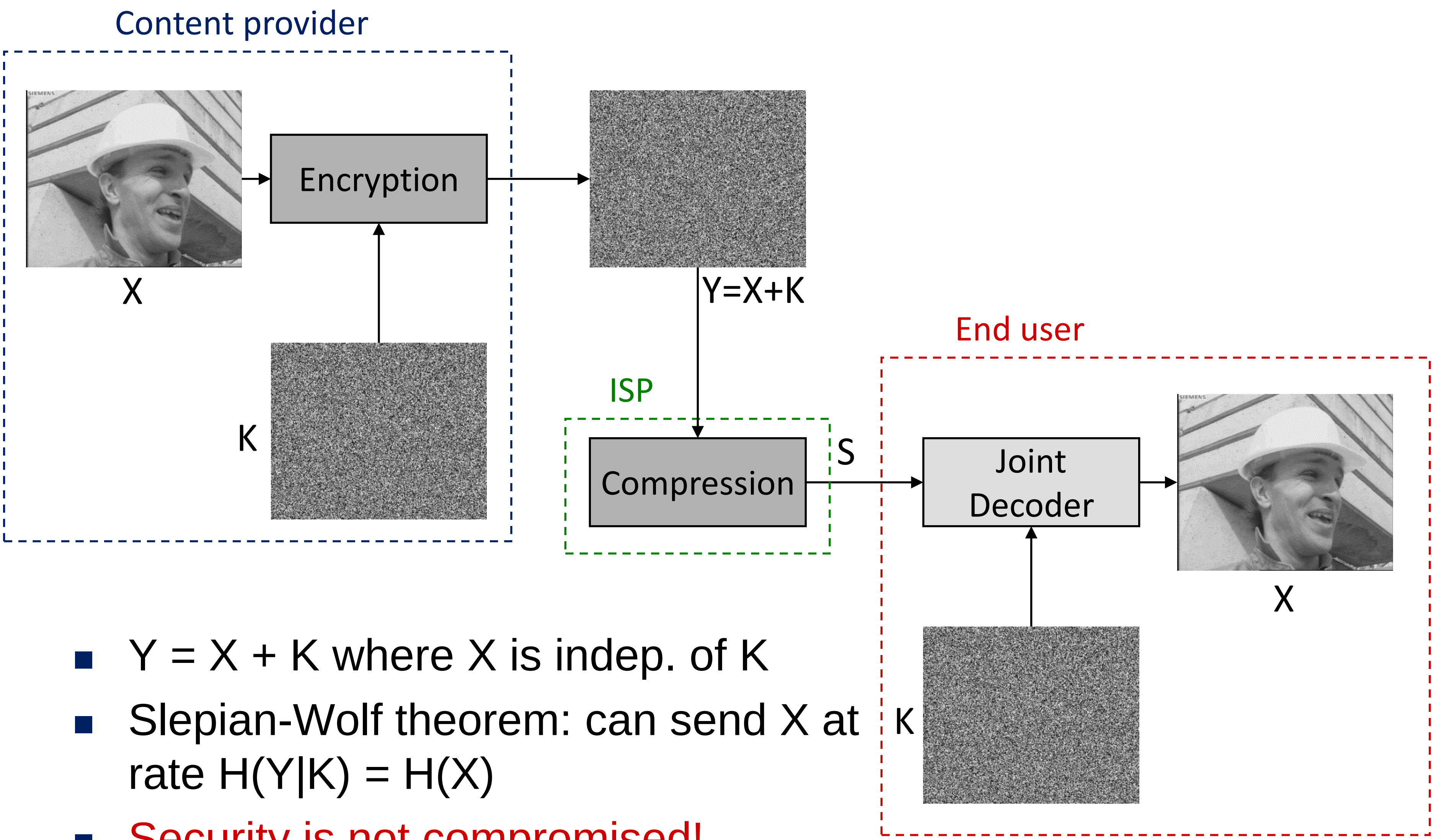
Application: compressing encrypted data



Key Insight!



Overview

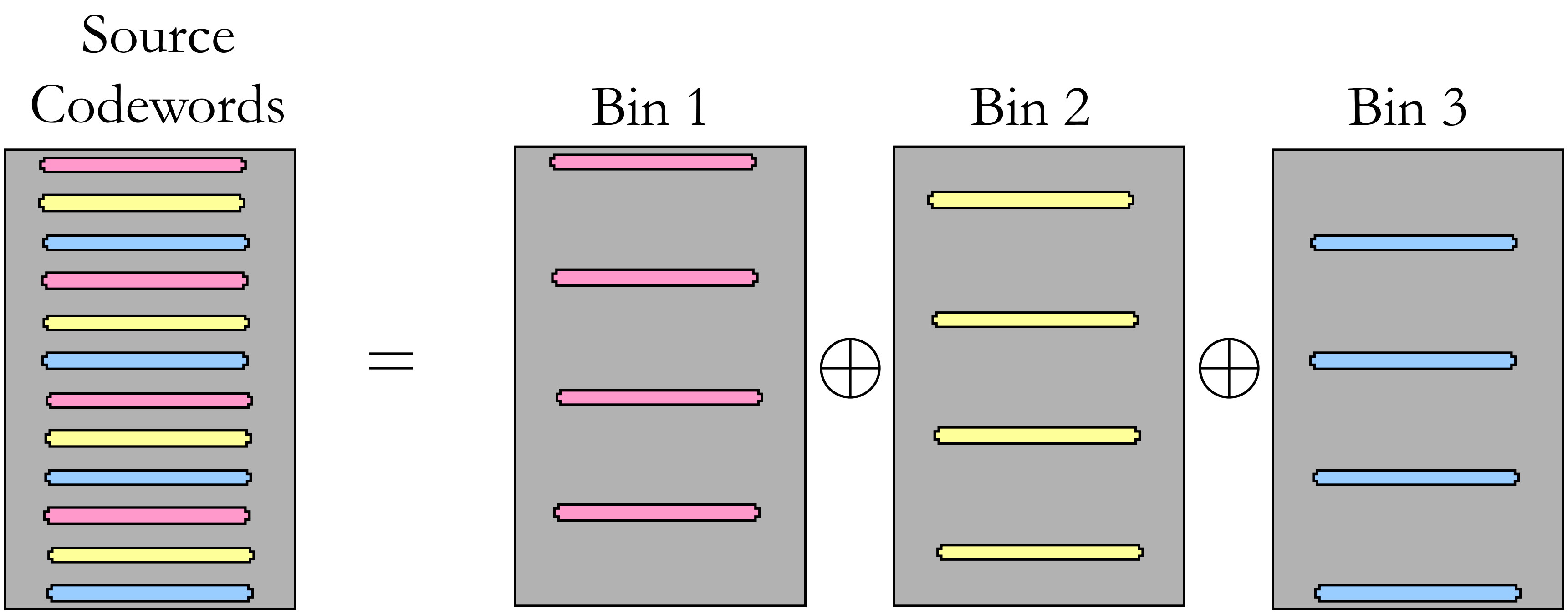


- $Y = X + K$ where X is indep. of K
- Slepian-Wolf theorem: can send X at rate $H(Y|K) = H(X)$
- Security is not compromised!

Johnson, Ishwar, Prabhakaran & Ramchandran (Trans. on SP, Oct. 2004)

Practical Code Constructions

- Use a linear transformation (hash/bin)
- Design cosets to have maximal spacing
  - State of the art linear codes (LDPC codes)
- Fixed length to fixed length compression



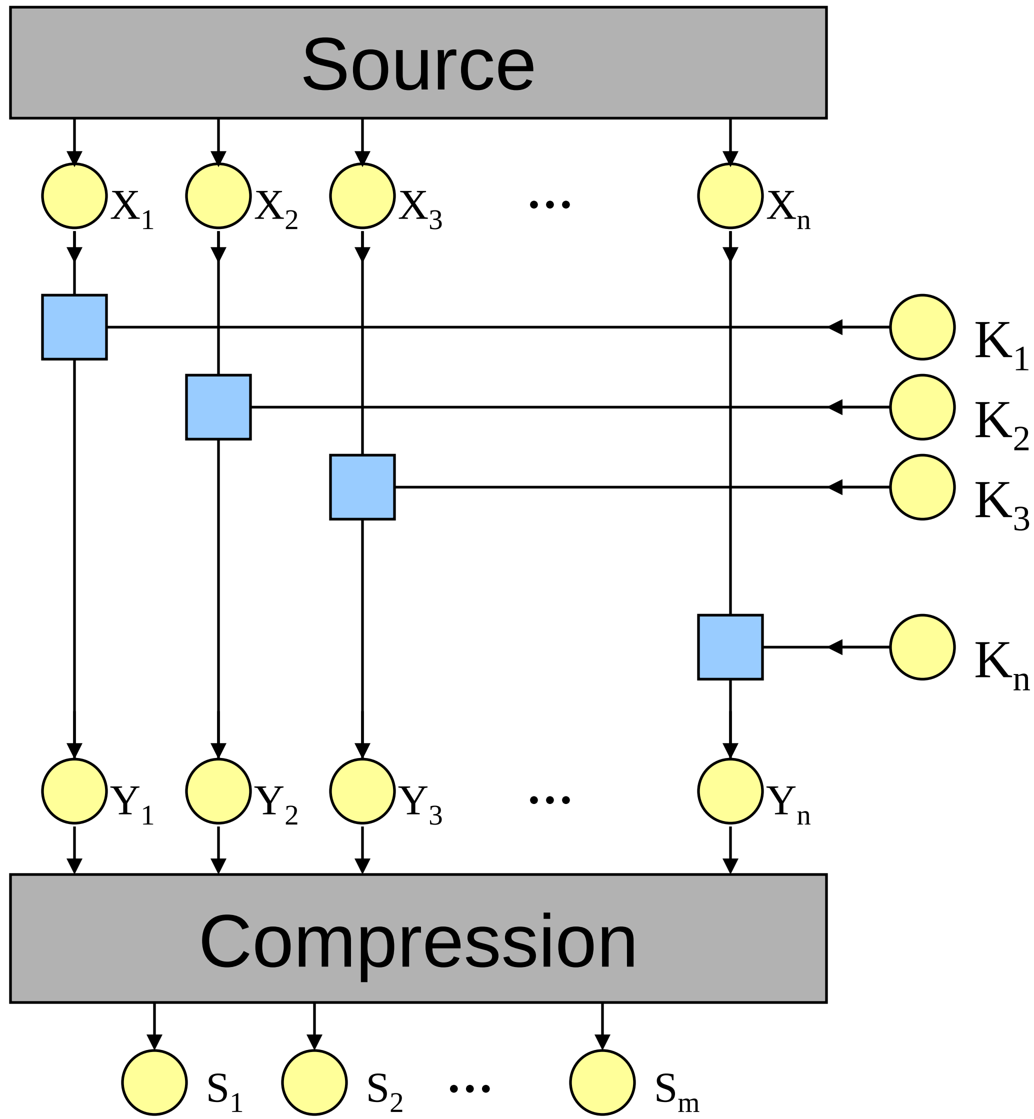
Framework: Encryption

Encryption:

- Stream cipher

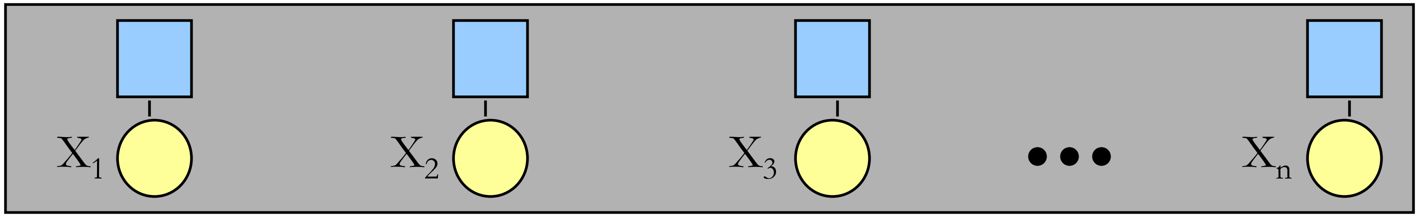
$$y_i = x_i \oplus k_i$$

- Graphical model captures exact encryption relationship

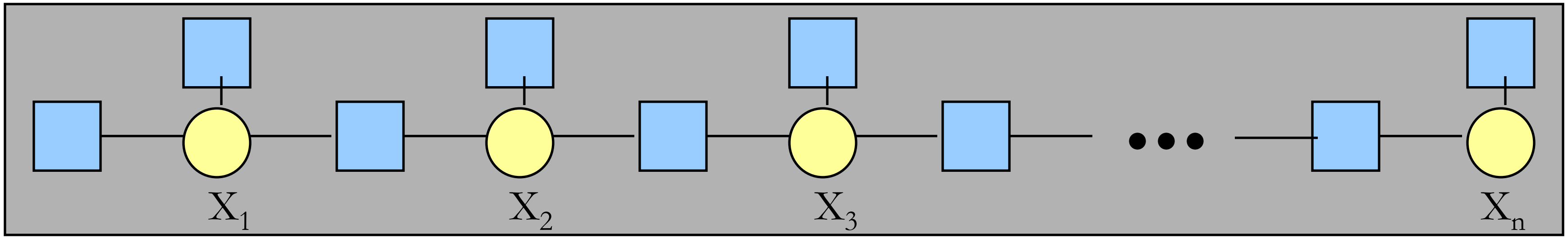


Source Models

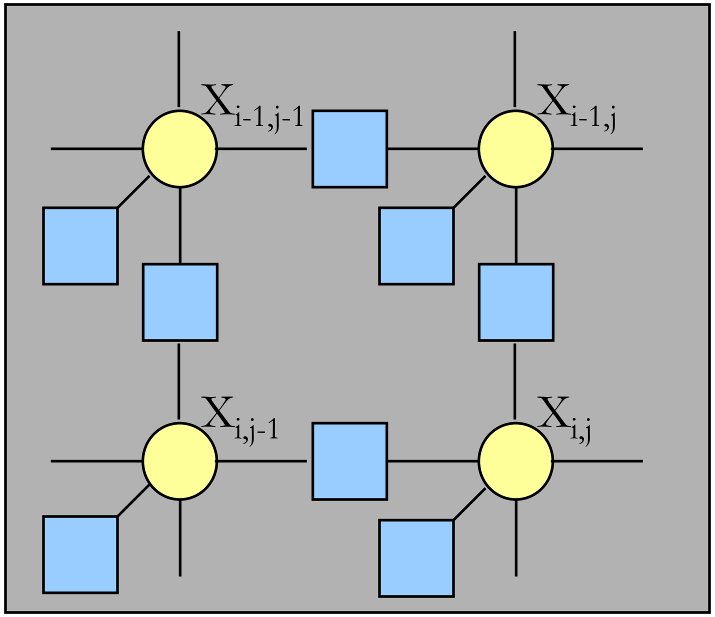
- IID Model



- 1-D Markov Model



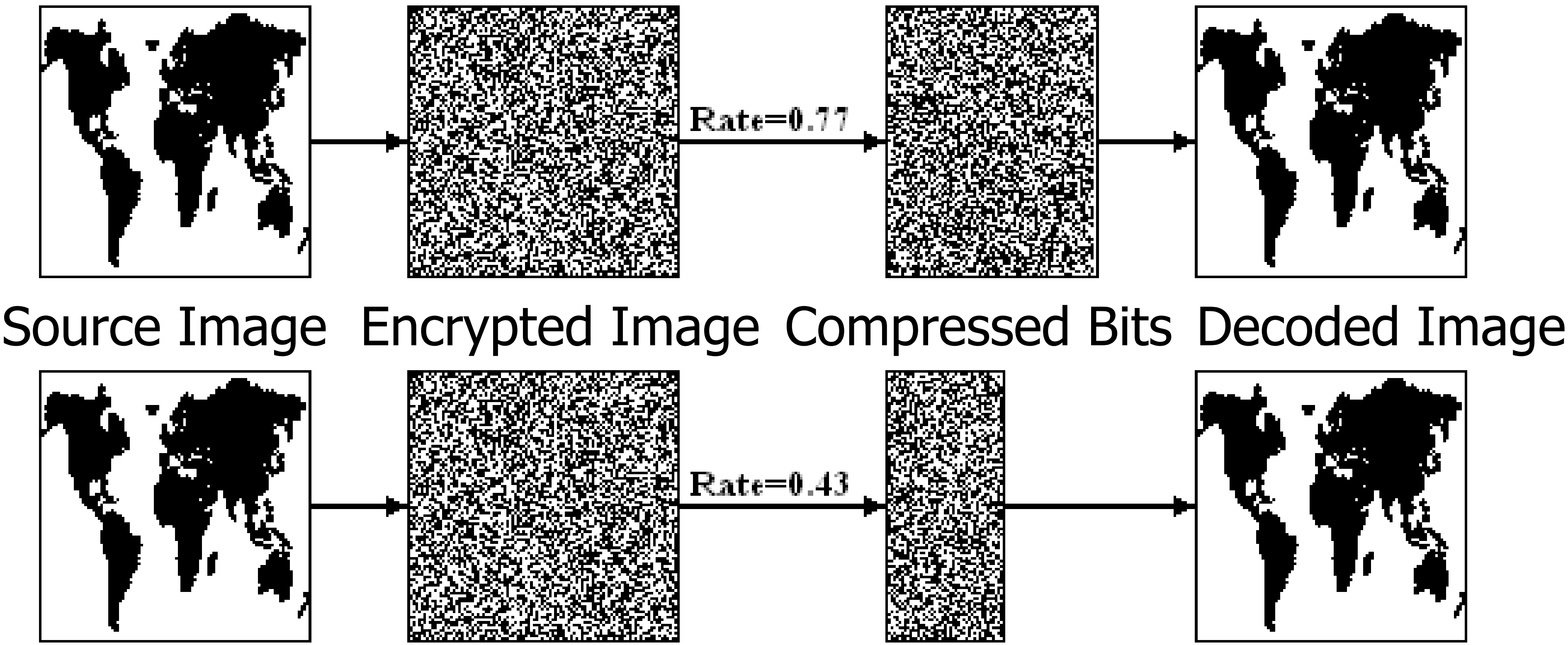
- 2-D Markov Model



Encrypted image compression results

- 100 x 100 pixel image (10,000 bits)
- No compression possible with IID model

1-D Markov Source Model

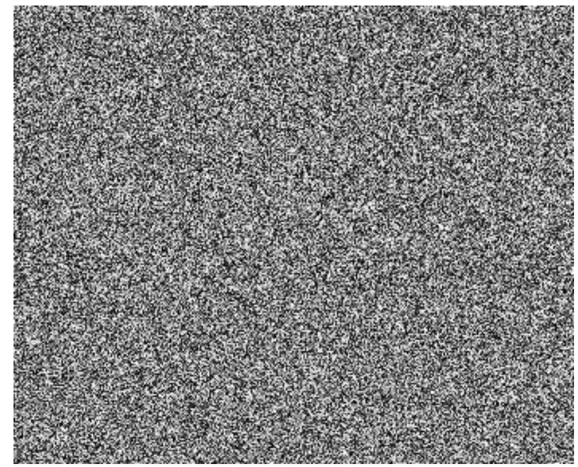


2-D Markov Source Model

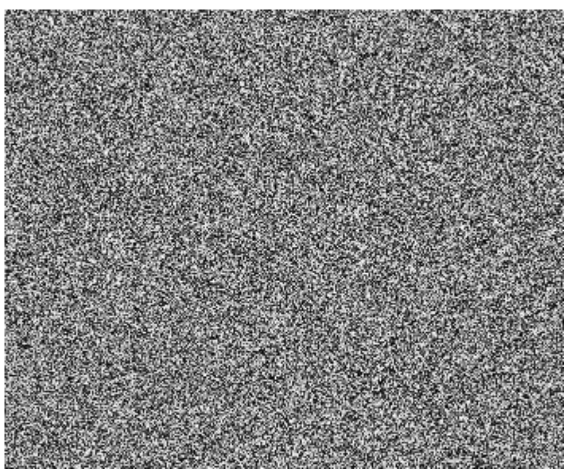
Compression of encrypted video

- **Video offers both temporal and spatial prediction**
  - **Decoder has access to unencrypted prior frames**

$\oplus$



=



	Blind approach (encoder has no access to key)
Foreman	**Saves 33.00%**
Garden	**Saves 17.64%**
Football	**Saves 7.17%**

Schonberg, Yeo, Draper & Ramchandran, DCC '07

Encrypted video compression results

- Show rate savings percentage
  - Rate used (output bits/source bit) is shown for reference
- Compare to operation on unencrypted video
  - JPEG-LS – lossless intra encoding of frames
  - Leading lossless video codec – exploits temporal redundancy

	JPEG-LS (unencrypted video)	Leading lossless video codec (unencrypted video)	Proposed approach (encrypted video, encoder has no access to key)
Foreman	**50.96%** R=0.4904	**58.87%** R=0.4113	**33.00%** R=0.6700
Garden	**26.80%** R=0.7320	**40.92%** R=0.5908	**17.64%** R=0.8236
Football	**33.00%** R=0.6700	**40.44%** R=0.5956	**7.17%** R=0.9283

Distributed source coding for video transmission: overview

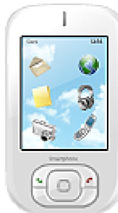
When is DSC useful in video transmission?

- **Uncertainty** in the side information
  - **Low complexity** encoding
  - Transmission **packet drops**
  - **Multicast** & scalable video coding
  - **Flexible** decoding

- Physically **distributed sources**
  - **Multi-camera** setups

Low complexity encoding

- Motivation



Low-Complexity Encoder

Trans-coding proxy

High-Complexity Decoder → High-Complexity Encoder

Low-Complexity Decoder

current frame

DSC Encoder

Low-complexity

(no motion search)

DSC Decoder

Side-info Generator

reference frame

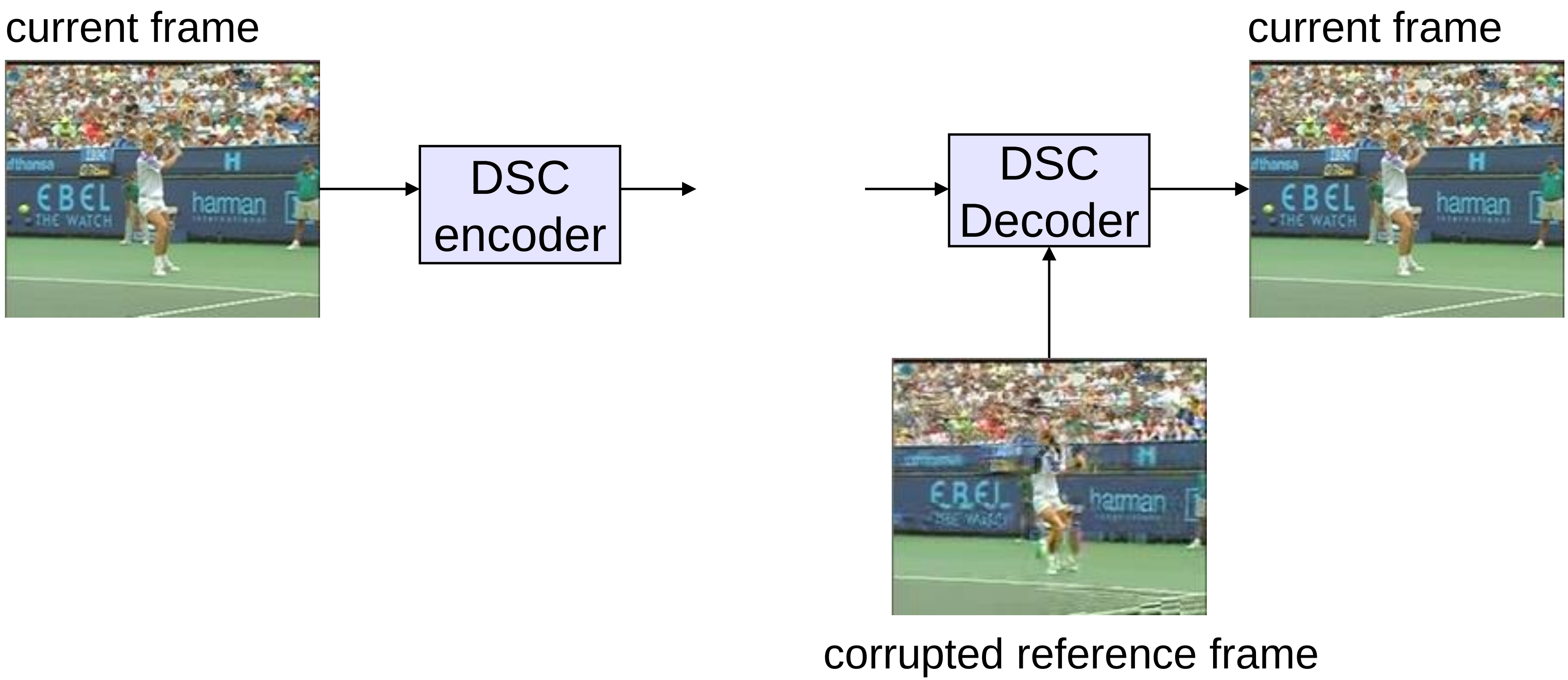
current frame

High-complexity

(interpolated or compensated motion)

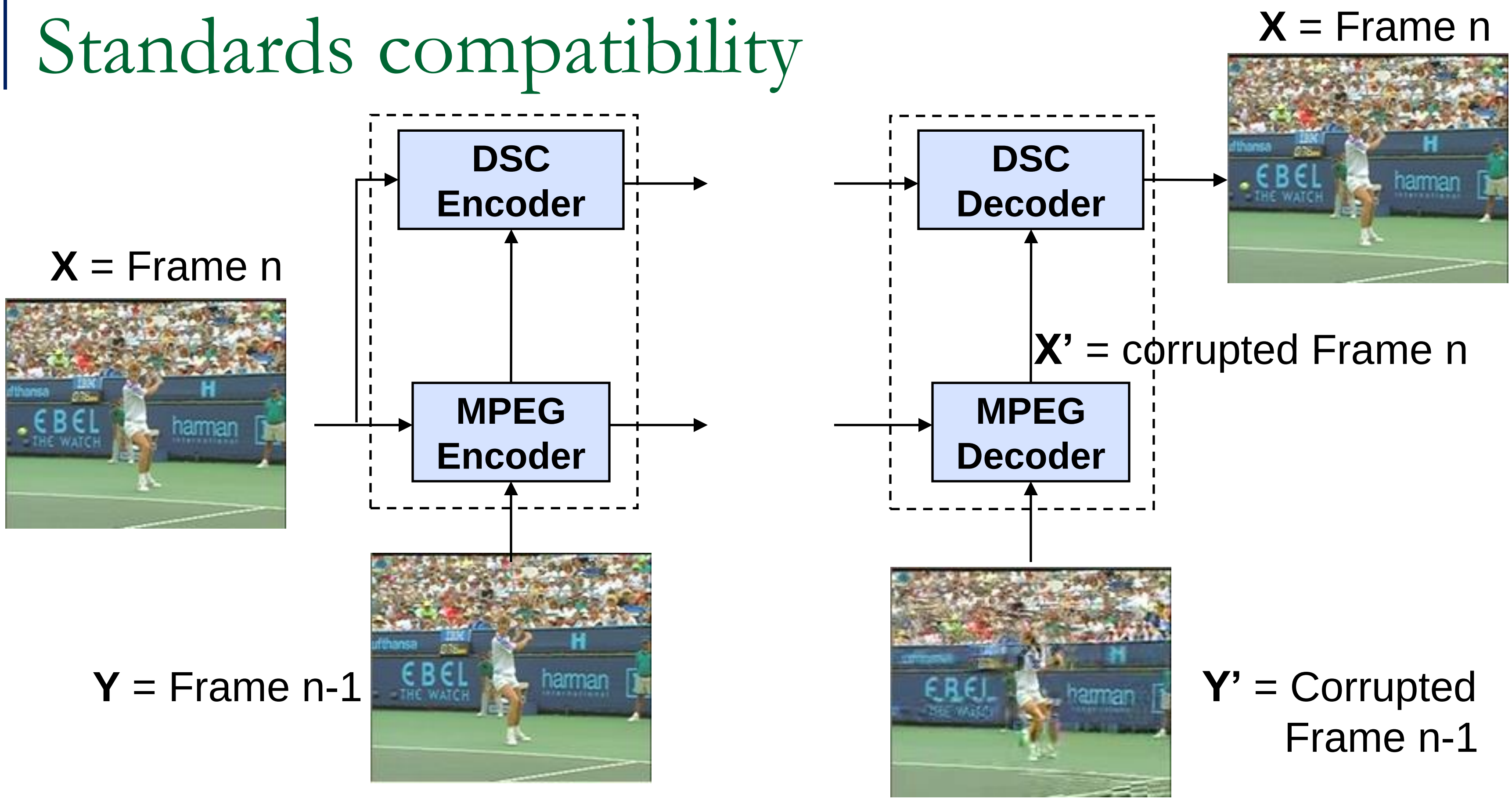
(Puri & Ramchandran, Allerton '02, Aaron, Zhang & Girod, Asilomar '02)

Transmission packet loss



- Recover current frame with (corrupted) reference frame **that is not available at the encoder**
- Distributed source coding: can help if **statistical correlation** bet. current and corrupted ref. frames known at the encoder

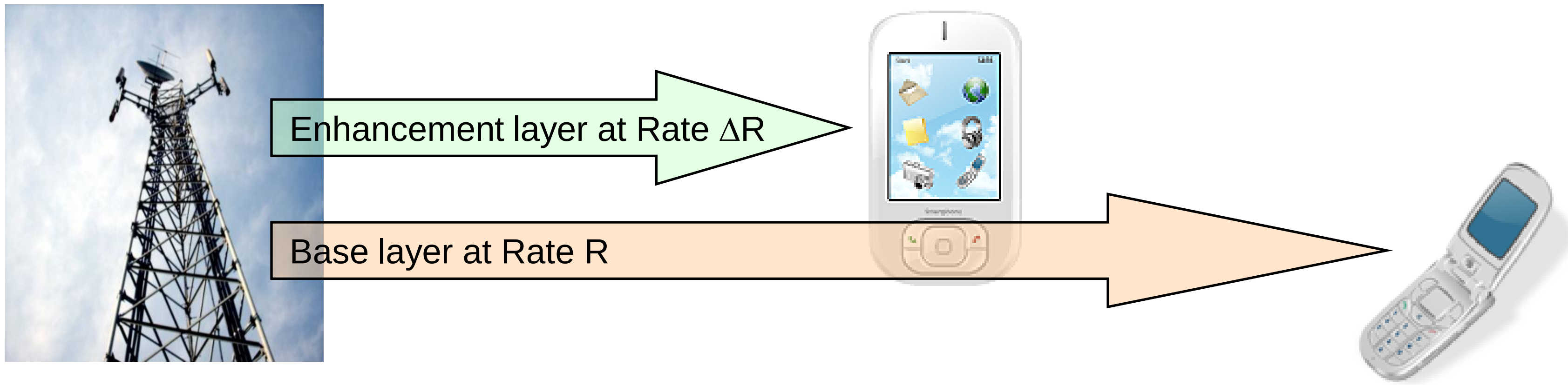
Standards compatibility



- Can be made **compatible** with **standards-based** codecs
- **Corrupted** current frame is **side-info** at DSC decoder

(Aaron, Rane, Rebollo-Monedero & Girod '04, '05, Sehgal, Jagmohan & Ahuja: '04, Wang, Majumdar & Ramchandran: '04, '05)

Multicast & scalable video coding

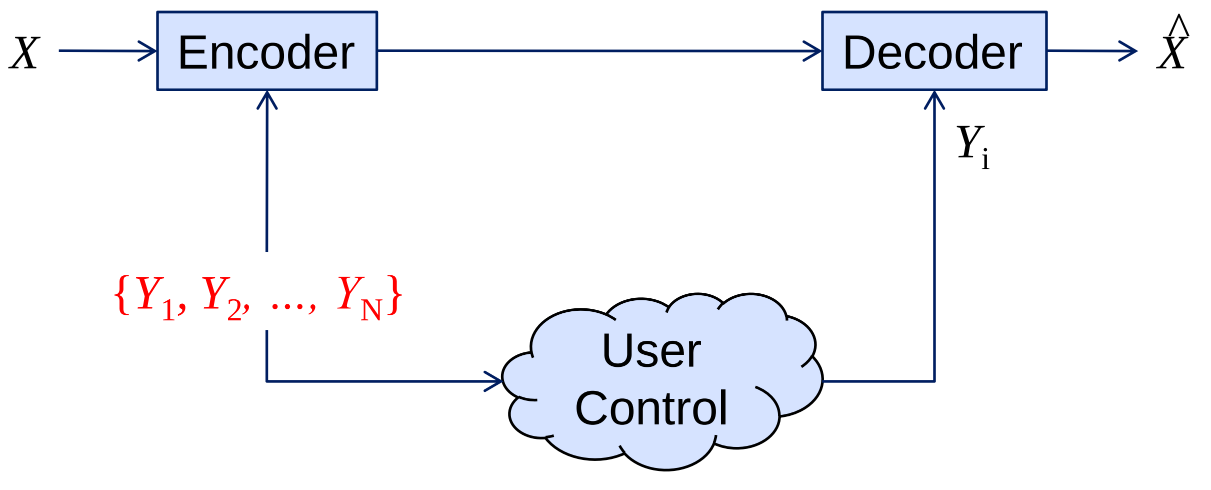


- Multicast
  - Accommodate heterogeneous users
    - Different channel conditions
    - Different video qualities (spatial, temporal, PSNR)

Majumdar & Ramchandran, ‘04
Tagliasacchi, Majumdar & Ramchandran, ‘04
Sehgal, Jagmohan & Ahuja, PCS ‘04
Wang, Cheung & Ortega, EURASIP ‘06
Xu & Xiong, ‘06

Flexible decoding

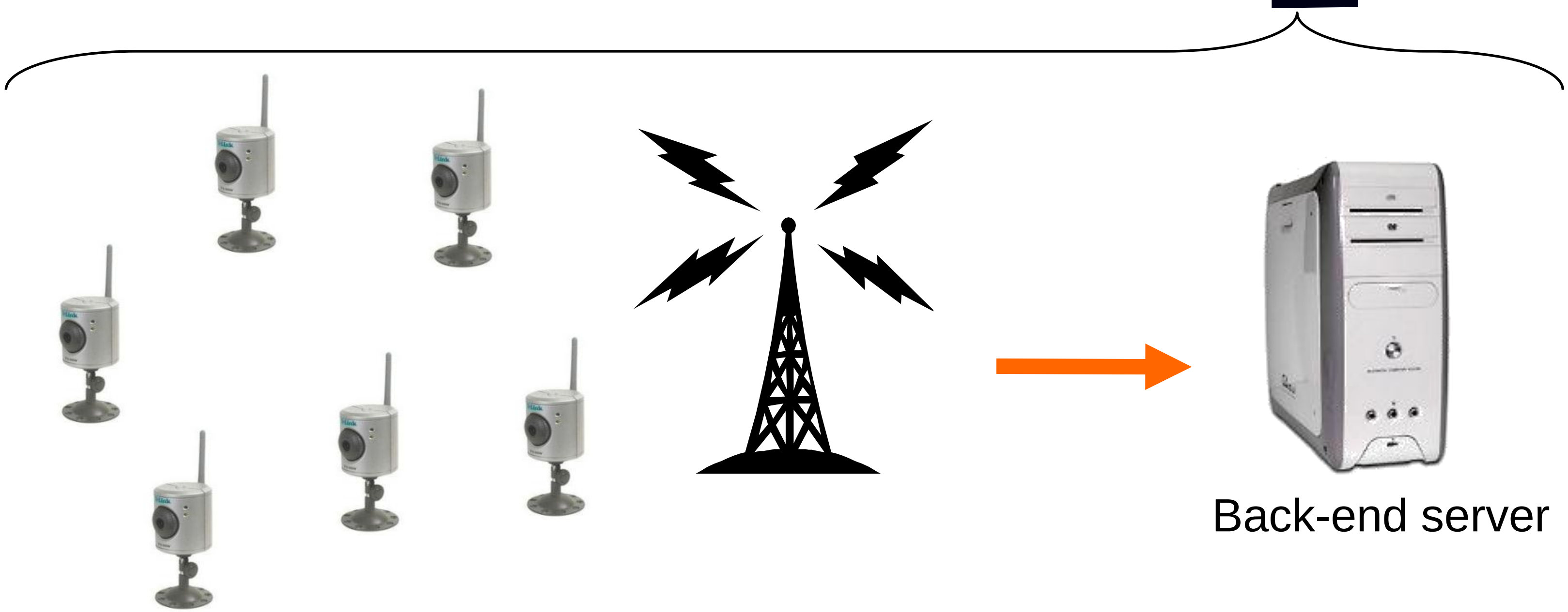
- $\{Y_1, Y_2, \ldots, Y_N\}$ could be
  - Neighboring frames in **time**
    → Forward/backward playback without buffering
  - Neighboring frames in **space**
    → Random access to frame in multi-view setup
  - …



Cheung, Wang & Ortega, VCIP 2006, PCS 2007
Draper & Martinian, ISIT 2007

Multi-camera setups

- Dense placement of low-end video sensors
- Sophisticated back-end processing
  - 3-D view reconstruction
  - Object tracking
  - Super-resolution
- Multi-view coding and transmission

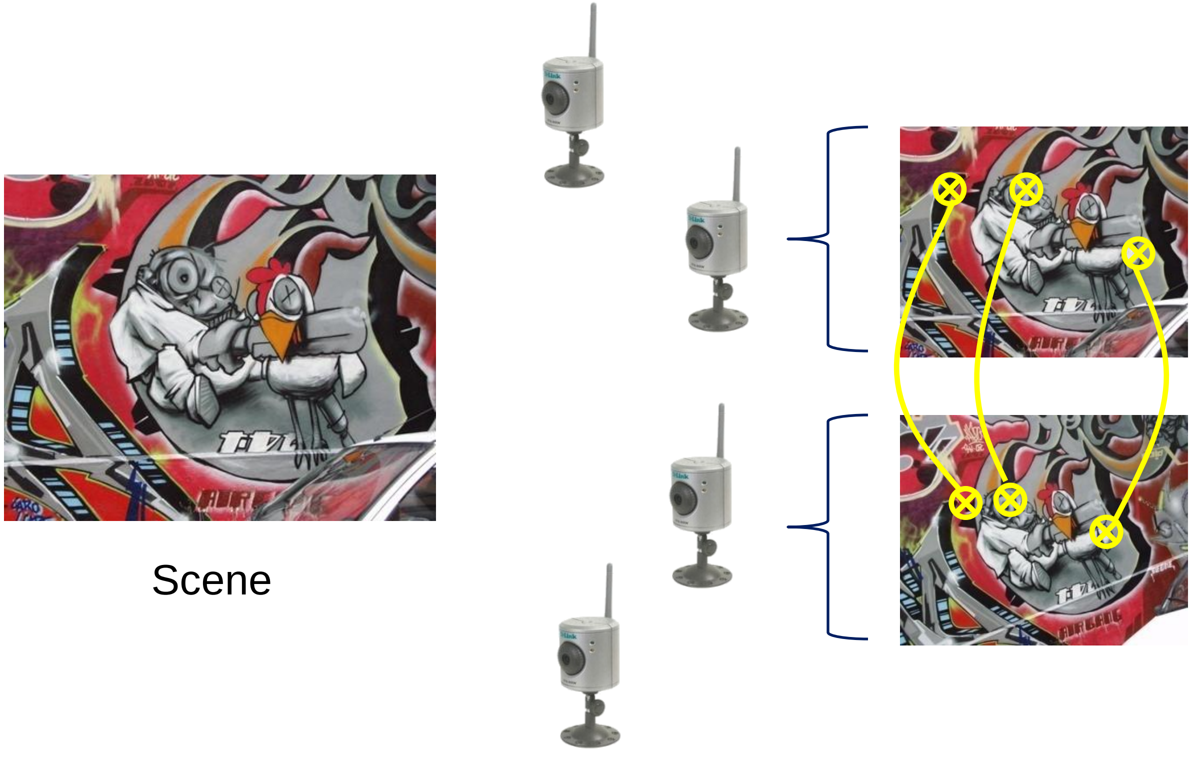


Important enabler

- Rate-efficient camera calibration
  - Visual correspondence determination

Tosic & Frossard, EUSIPCO 2007

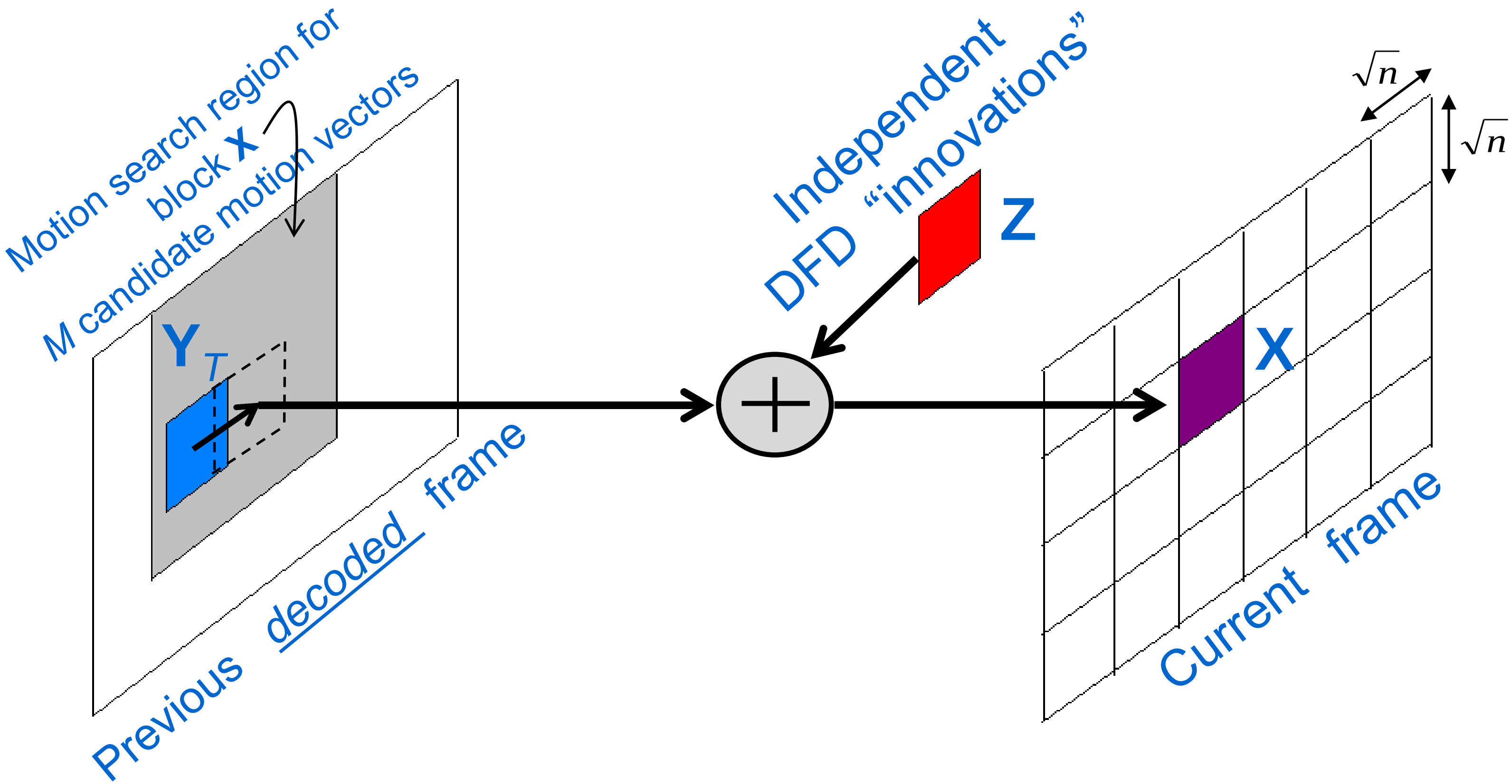
Yeo, Ahammad & Ramchandran, VCIP 2008

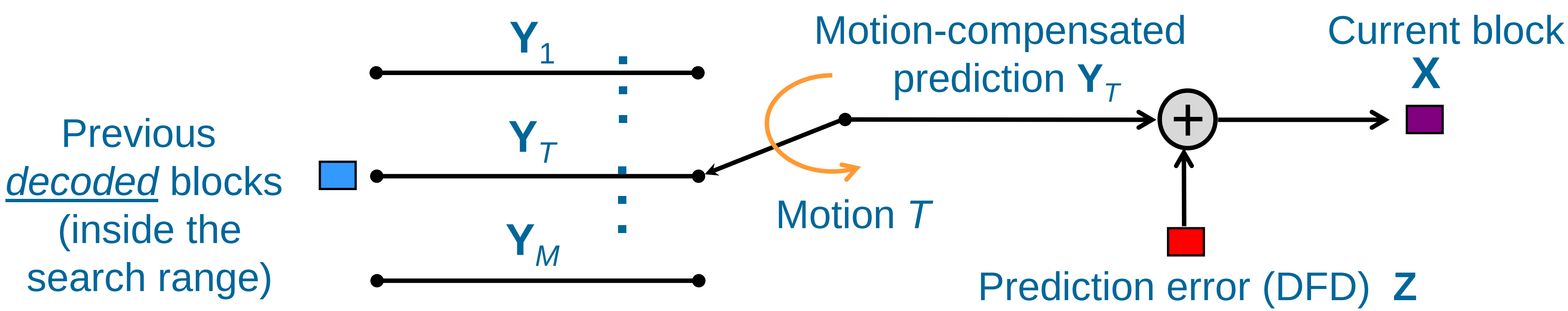


Scene

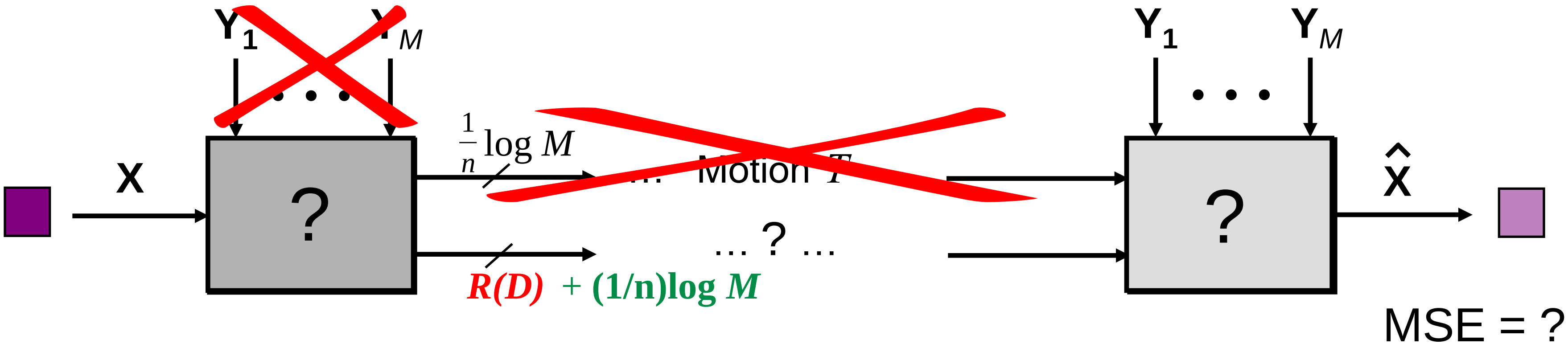
DSC for video transmission: PRISM- I –targeting low-complexity encoding

MCPC: a closer look



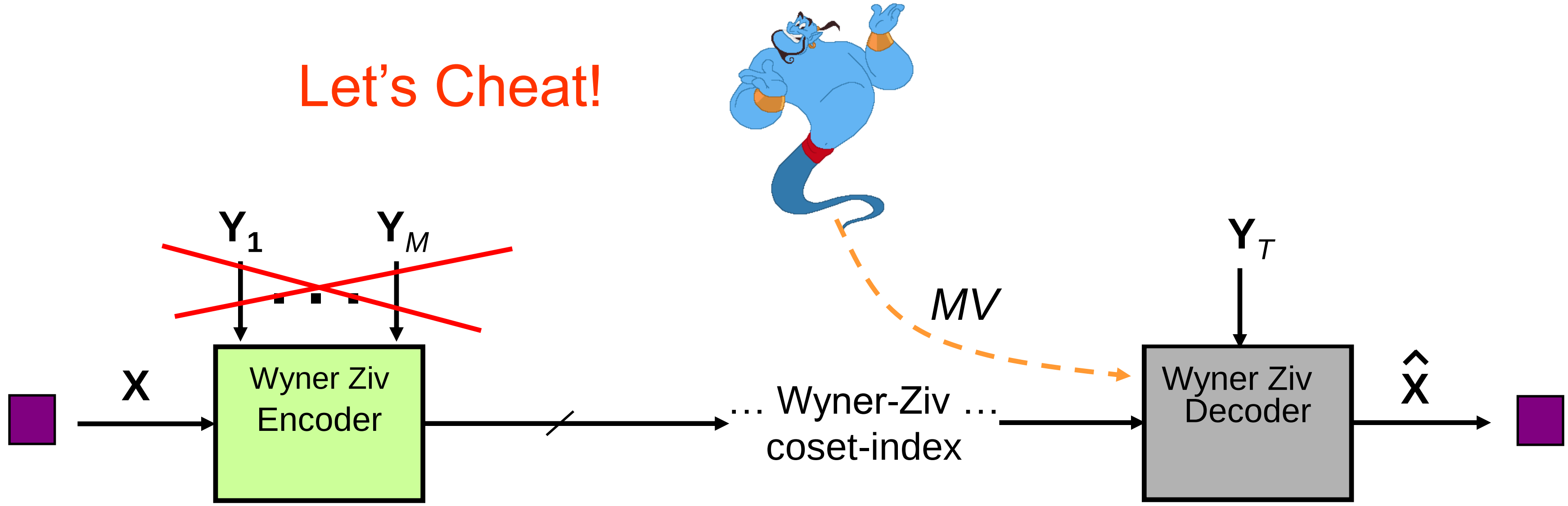


Motion-free encoding?



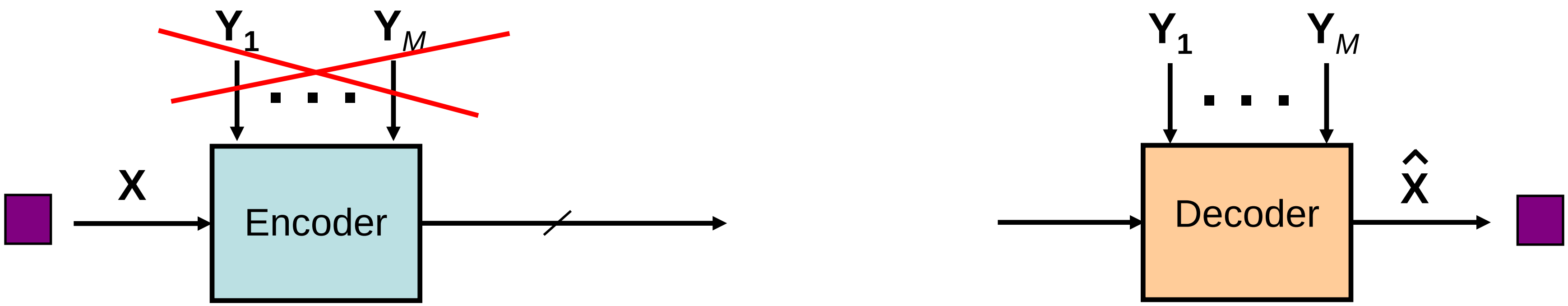
- The encoder does not have or cannot use $\mathbf{Y}_1, \ldots, \mathbf{Y}_M$ and
- The decoder does not know *T.*
- The encoder may work at rate: ***R(D)*** + ***(1/n)*****log *M*** bits per pixel.
- How to decode and what is the performance?

Is a No-Motion Encoder Possible?



- Let's cheat & let the decoder have the MV ➔ "classical" W-Z problem
- The encoder works at same rate as predictive coder

Is a No-Motion Encoder Possible?



- Can decoding work without a genie?
  - *Yes*
- Can we match the performance of predictive coding?
  - *Yes* (when DFD statistics are Gaussian)

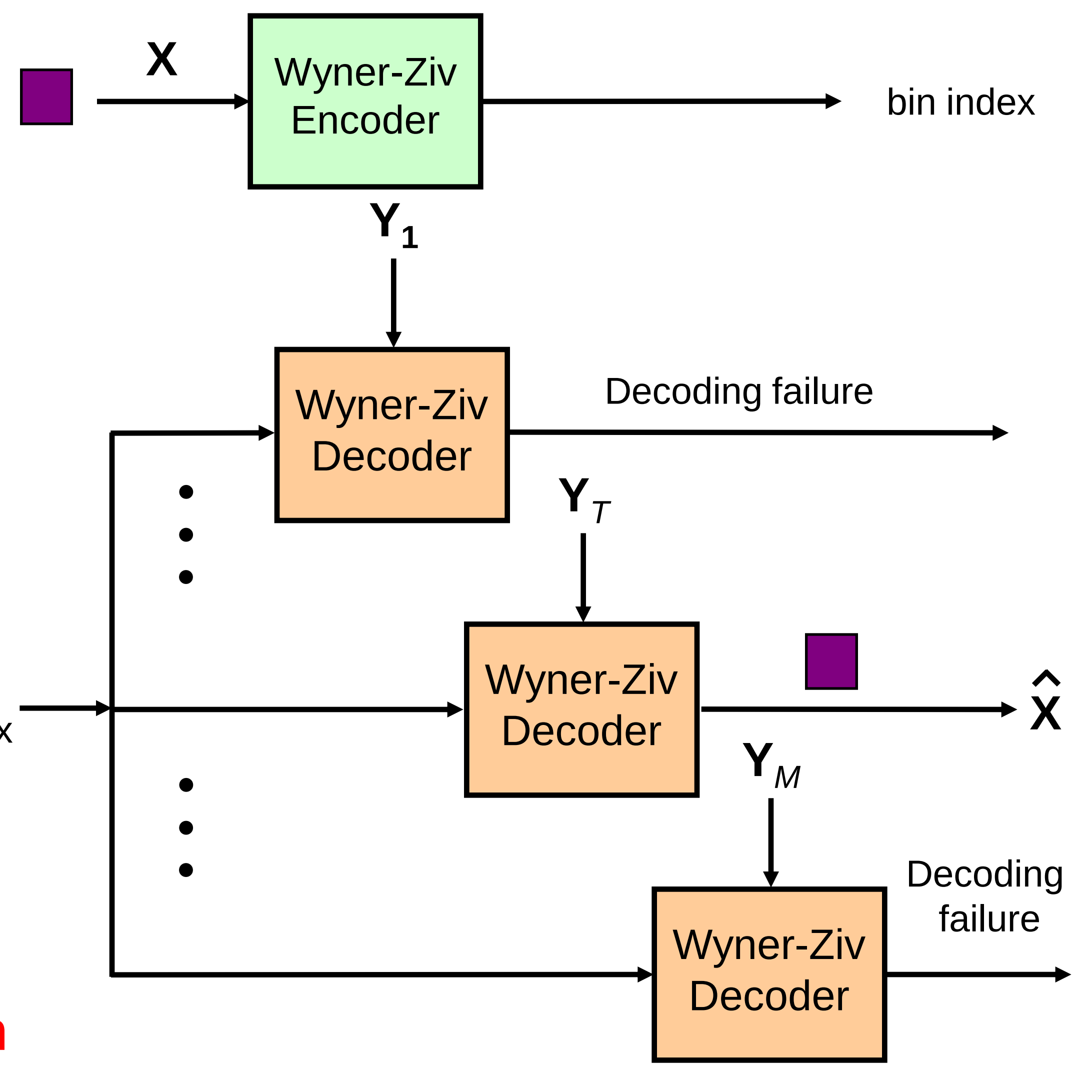
Ishwar, Prabhakaran, and Ramchandran ICIP ’03.

Motion search at decoder

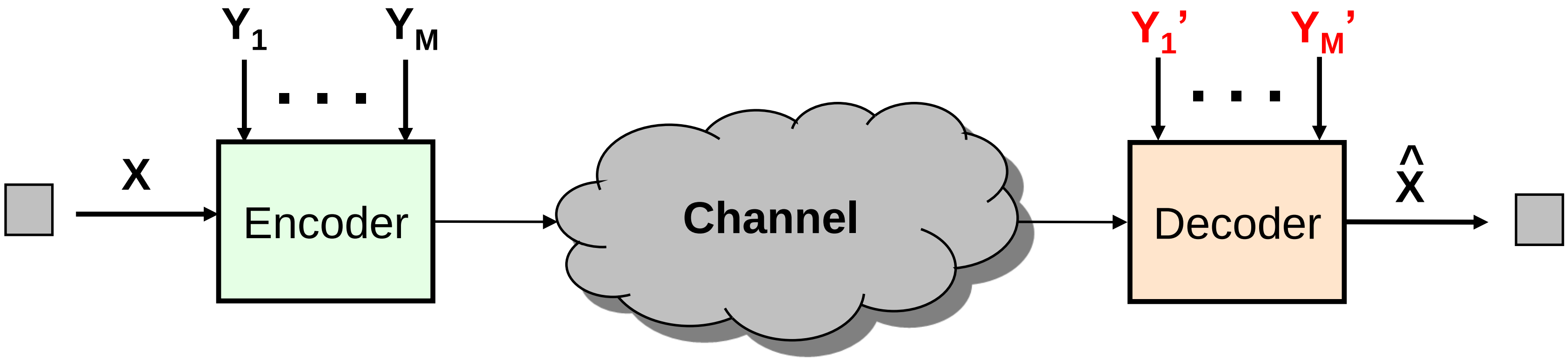
- Low-complexity motion-free encoder

- Need mechanism to detect decoding failure
- In theory: joint typicality (statistical consistency)
- In practice: use CRC
- **Complexity "knob" to share search complexity between enc. & decoder**

Need concept of "motion compensation at decoder"



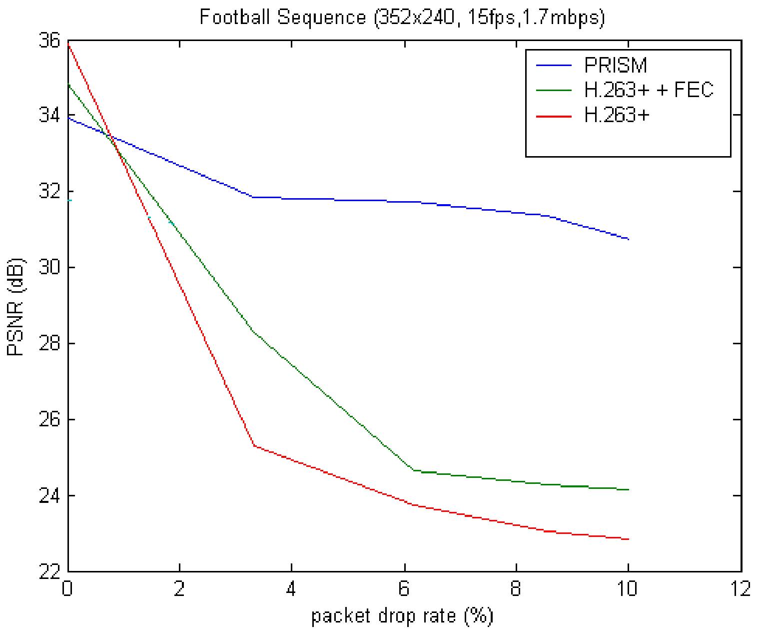
Practical implementation



- Can be realized through decoder motion search
  - Extendable to when side-information is corrupted
    - **robustness** to channel loss
  - Correlation between X and Y_i' difficult to estimate due to low-complexity encoding
    - compression efficiency compromised

Robustness Results: PRISM-I video codec

- Qualcomm's channel simulator for CDMA 2000 1X wireless networks



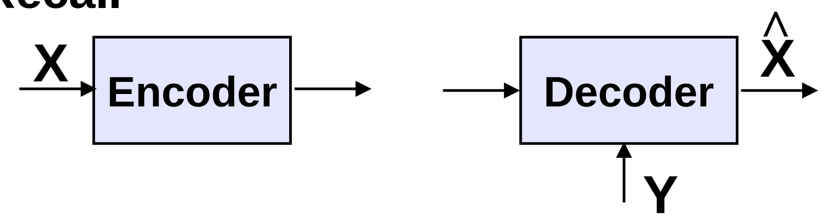
- Stefan
  (SIF, 2.2 Mbps, 5% error)

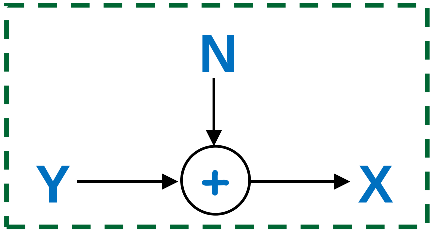
PRISM vs. H.263+ FEC

DSC for video transmission: PRISM II – targeting high-compression efficiency & robustness

Cause of compression inefficiency

Recall





- Challenge: correlation estimation, i.e. finding $H(X|Y) = H(N)$
- N = Video innovation + Effect of channel + Quantization noise

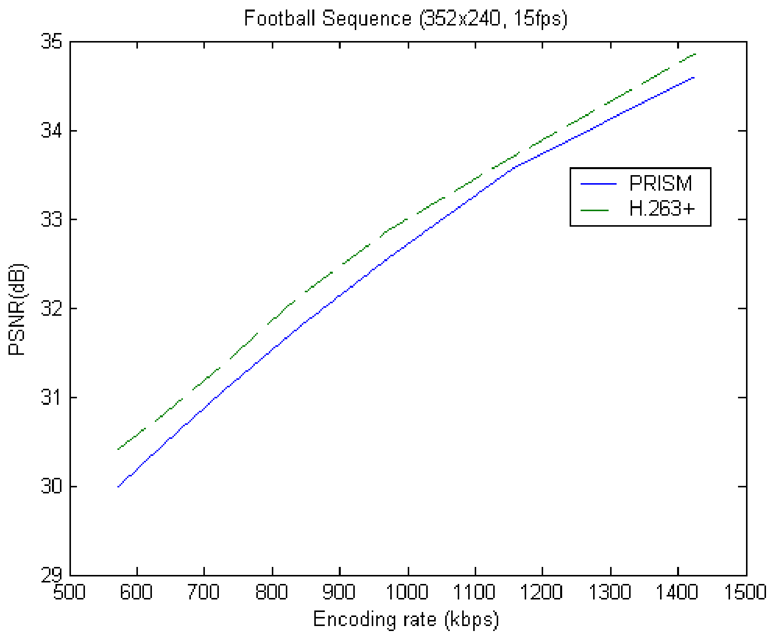
  Hard to model without motion search
- Without *accurate* estimate of the total noise statistics, need to over-design → compression inefficiency.
- **What if complexity were less of a constraint and we allow motion search at the encoder?**

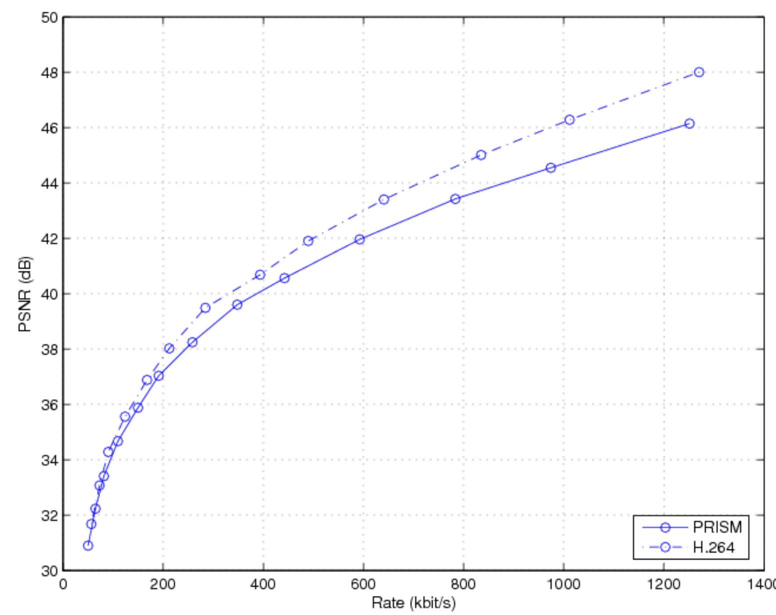
Video innovation can be accurately modeled

When there are no channel errors:

N = Video innovation + Quantization noise

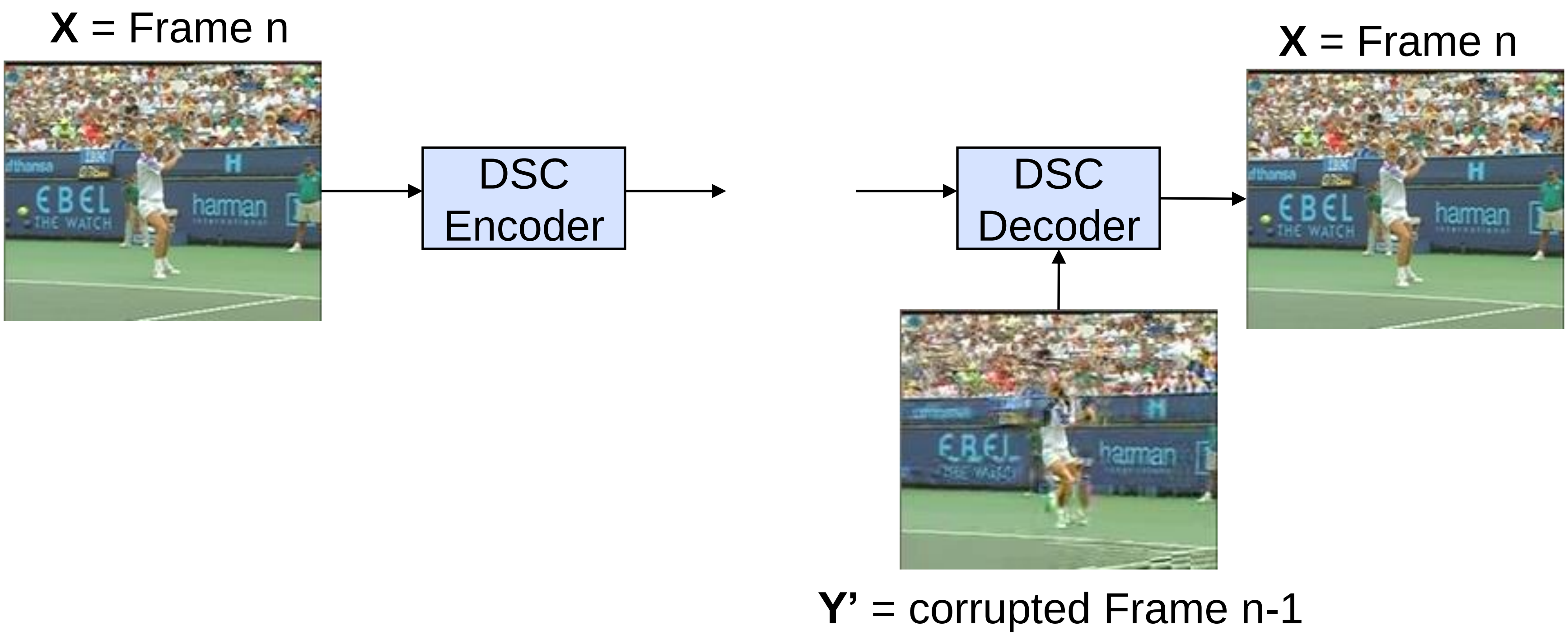
- DSC vs. H.263+
- DSC vs. H.264





Milani, Wang & Ramchandran, VCIP 2007

Modeling effect of channel at encoder



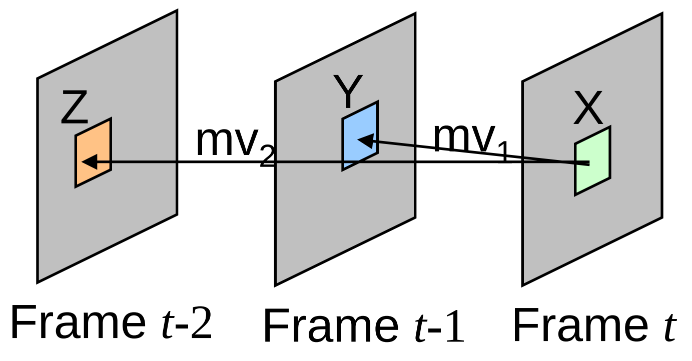
- Goal: estimate H (X|Y')

Finding H(X|Y')

- Philosophy: have control over uncertainty set at decoder
  - e.g. orchestrate decoder designs for

$$Y' = \begin{cases} Y \text{ if } Y \text{ is available} \\ Z \text{ if } Y \text{ is not available} \end{cases}$$

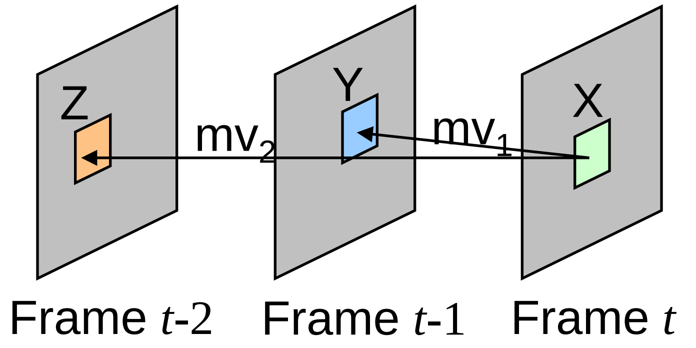
- Example:



  - Encoder has access to ***both*** Y and Z
  - Natural temporal redundancy in video: "diversity" gain
    - an **intact** predictor in **Frame *t*-2** (Z) is typically a better predictor than a **corrupted** predictor Y' in **Frame *t*-1**

J. Wang, V. Prabhakaran & K. Ramchandran: ICIP'06

Finding H(X|Y')



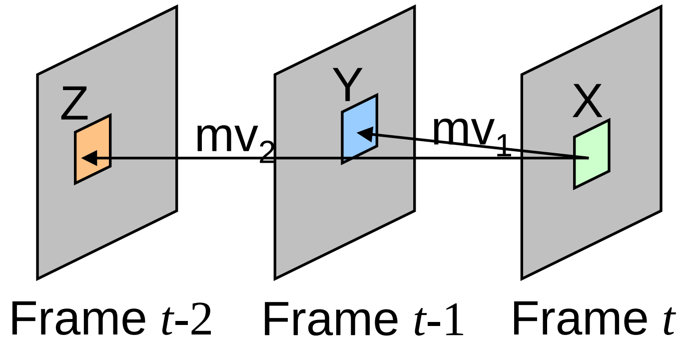
- If we have some knowledge about the channel:

$$Y' = \begin{cases} Y \text{ if } Y \text{ is intact} & \text{with probability } (1\text{-}p) \\ Z \text{ if } Y \text{ is corrupted} & \text{with probability } p \end{cases}$$

- We obtain

$H(X|Y', \text{decoder state}) = (1\text{-}p)*H(X|Y) + p*H(X|Z)$

Another way to think about it



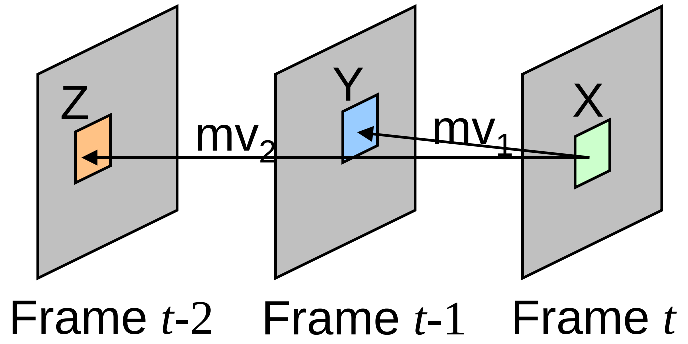
- H(X|Y', decoder state) = (1-p)*H(X|Y) + p*H(X|Z)

$$= \boxed{p^*[H(X|Z) - H(X|Y)]} + \boxed{H(X|Y)}$$

Effect of channel

Video innovation

Yet another way to think about it



- H(X|Y', decoder state) = (1-p)*H(X|Y) + p*H(X|Z)

$$= \boxed{p} * \boxed{[H(X|Z) - H(X|Y)]} + \boxed{H(X|Y)}$$

Can be achieved by applying **channel code** to sub-bin indices

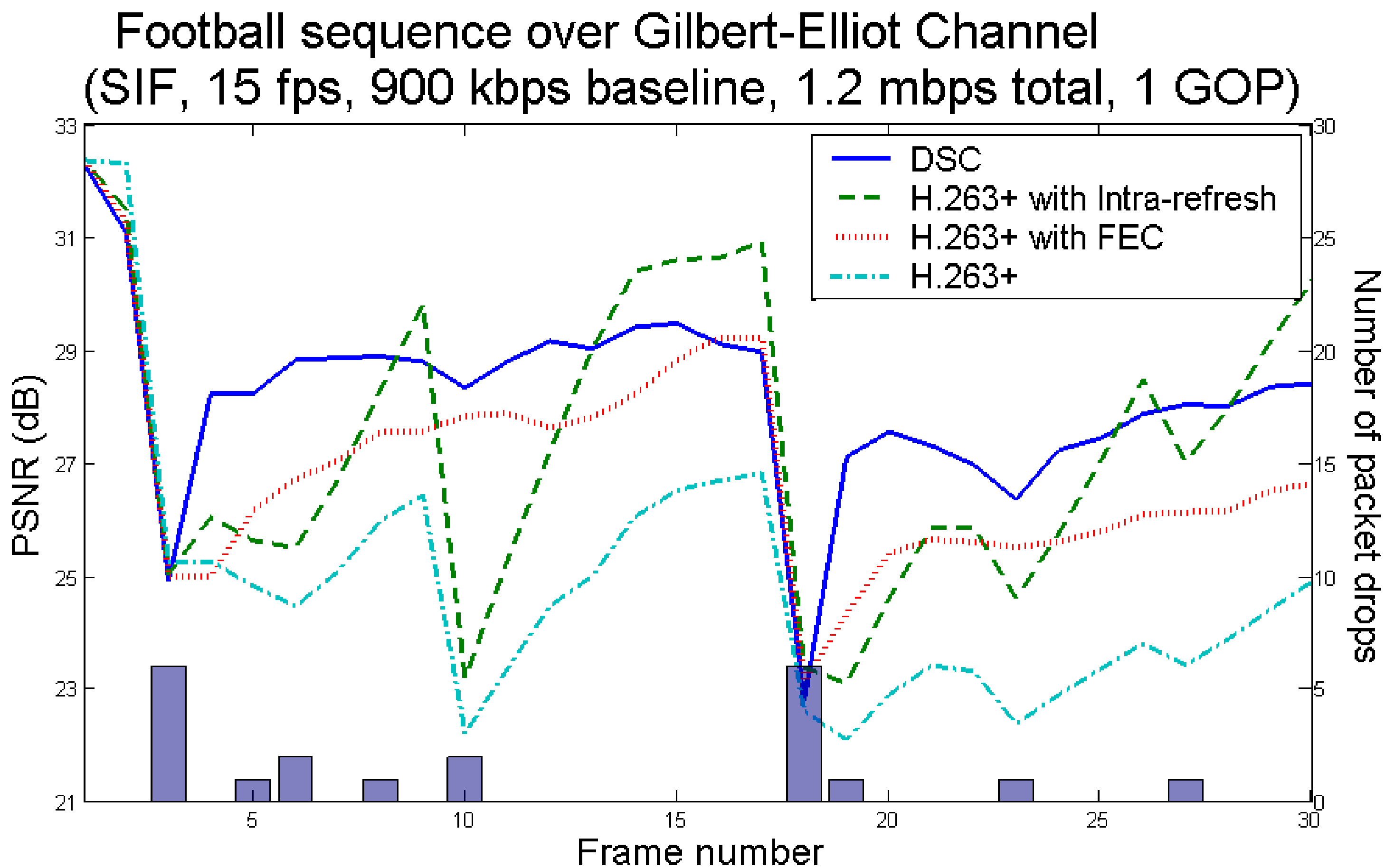
Additional syndrome (sub-bin index) for **drift correction**

Bare **minimum** syndrome (bin index) needed when channel is clean

Robustness result

Setup:

- Channel:
  - Simulated Gilbert-Elliot channel with $p_g = 0.03$ and $p_b = 0.3$

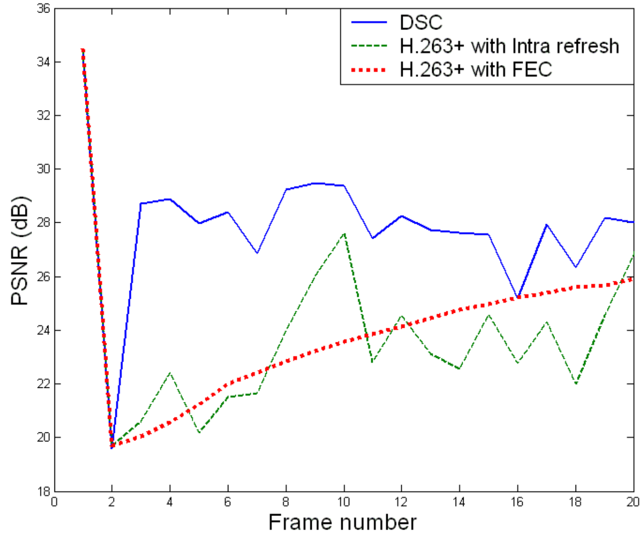


Robustness result

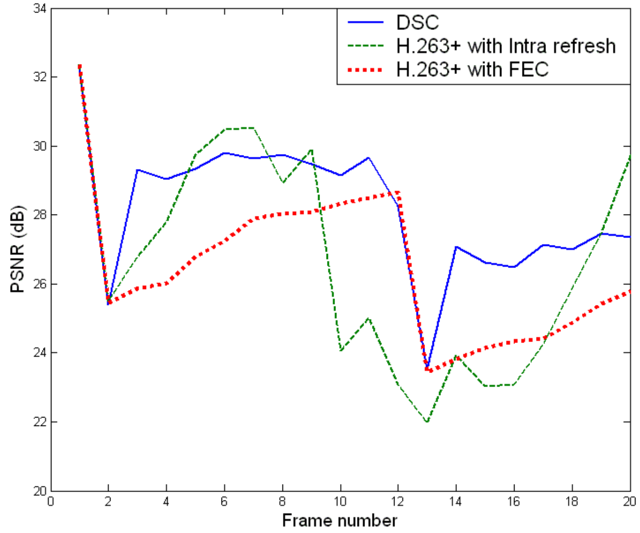
Setup:

- Channel:
  - Simulated CDMA 2000 1x channel

Stefan (SIF) sequence
1 GOP = 20 frames
1 mbps baseline, 1.3 mbps total (15 fps)
7.1% average packet drop rate



Football (SIF) sequence
1 GOP = 20 frames
900 kbps baseline, 1.12 mbps total (15 fps)
7.4% average packet drop rate



Videos

- Garden

  352x240, 1.4 mbps, 15 fps, gop size 15, 4% error
  (Gilbert Elliot channel with 3% error rate in good state and 30% in bad state)

  DSC
  vs.
  H.263+ FEC

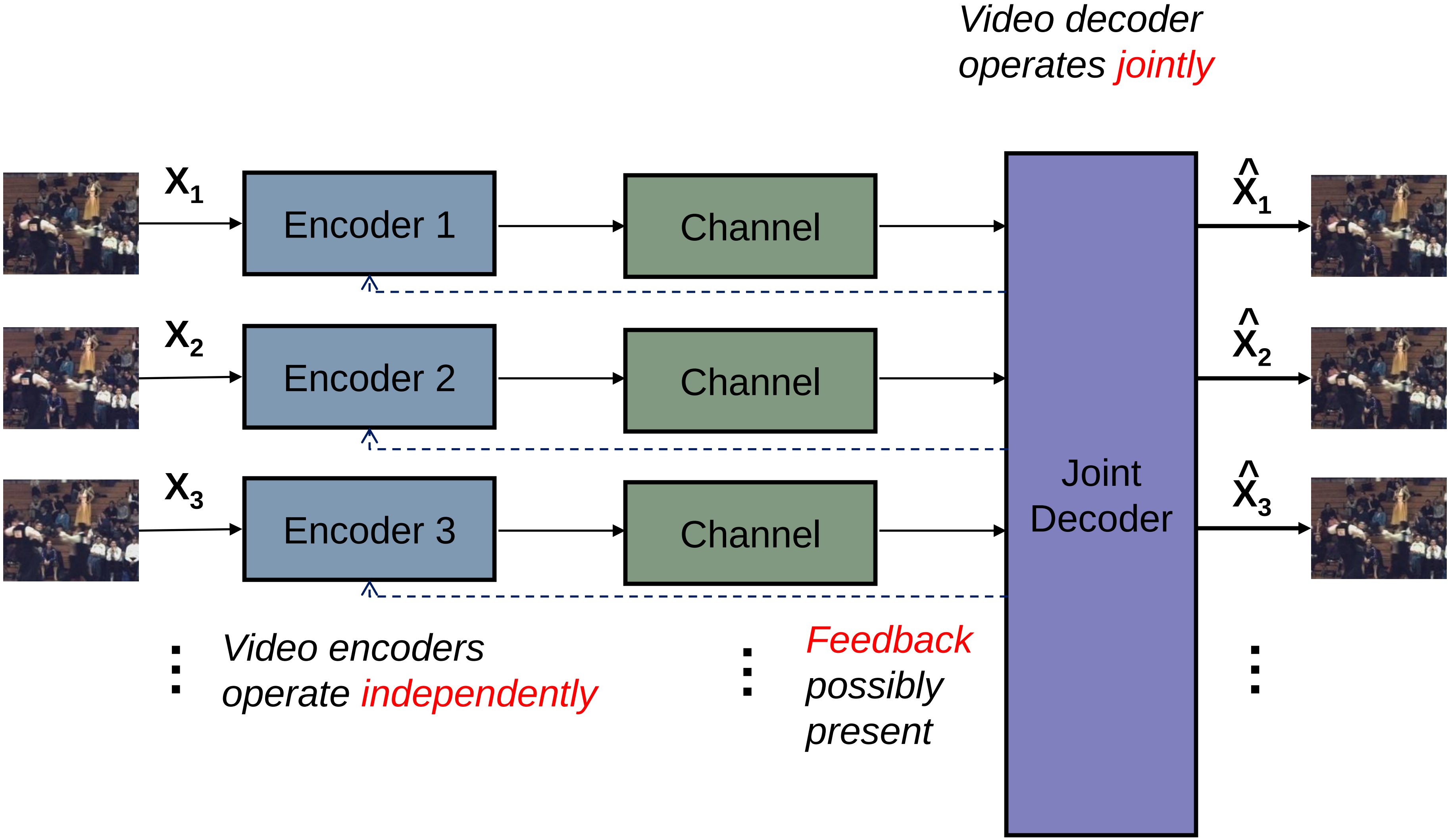
- Football

  352x240, 1.12 mbps, 15 fps, gop 15, simulated CDMA channel with 5% error

  DSC
  vs.
  H.263+ FEC

DSC for multi-camera video transmission:

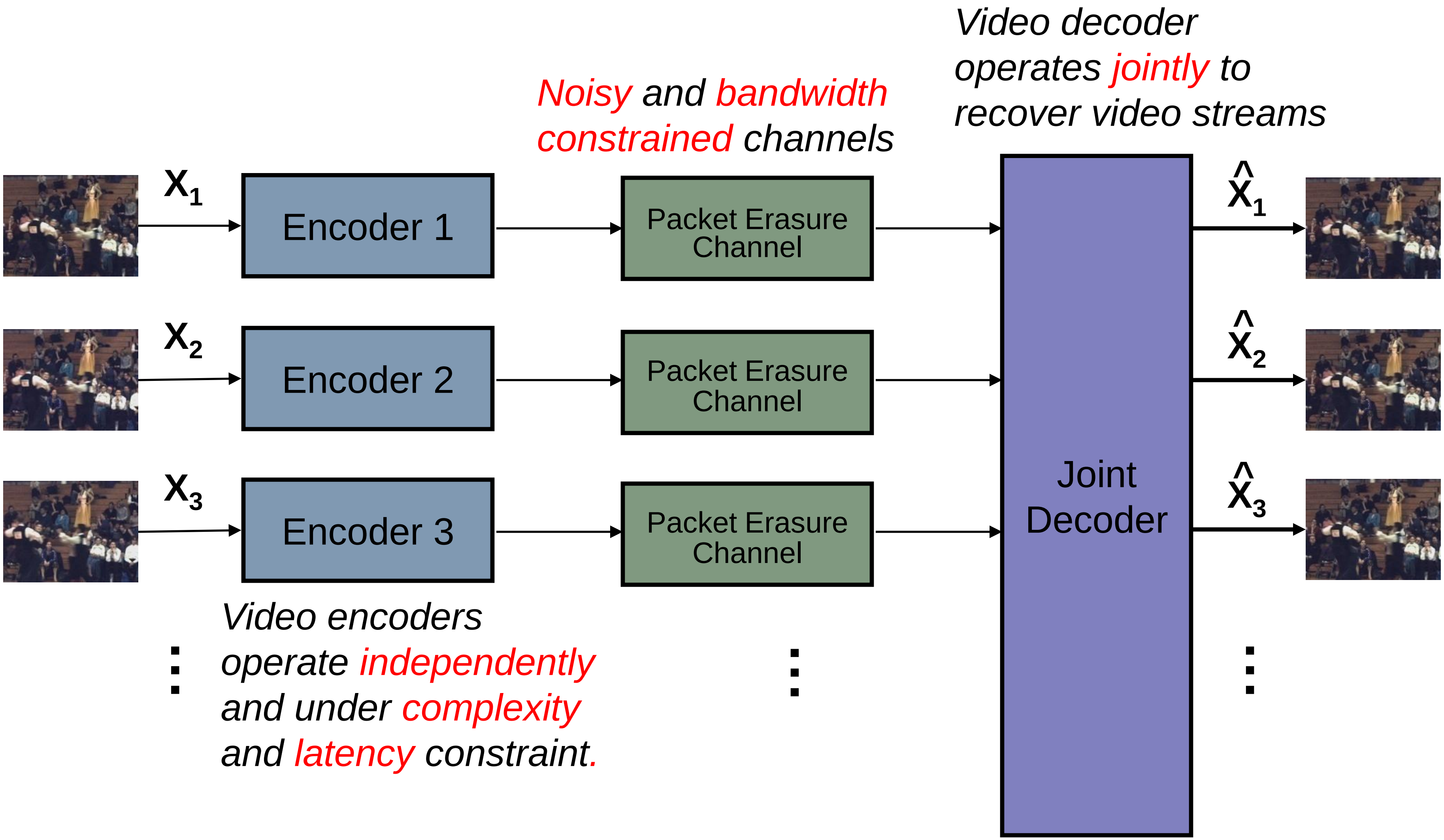
Distributed multi-view coding



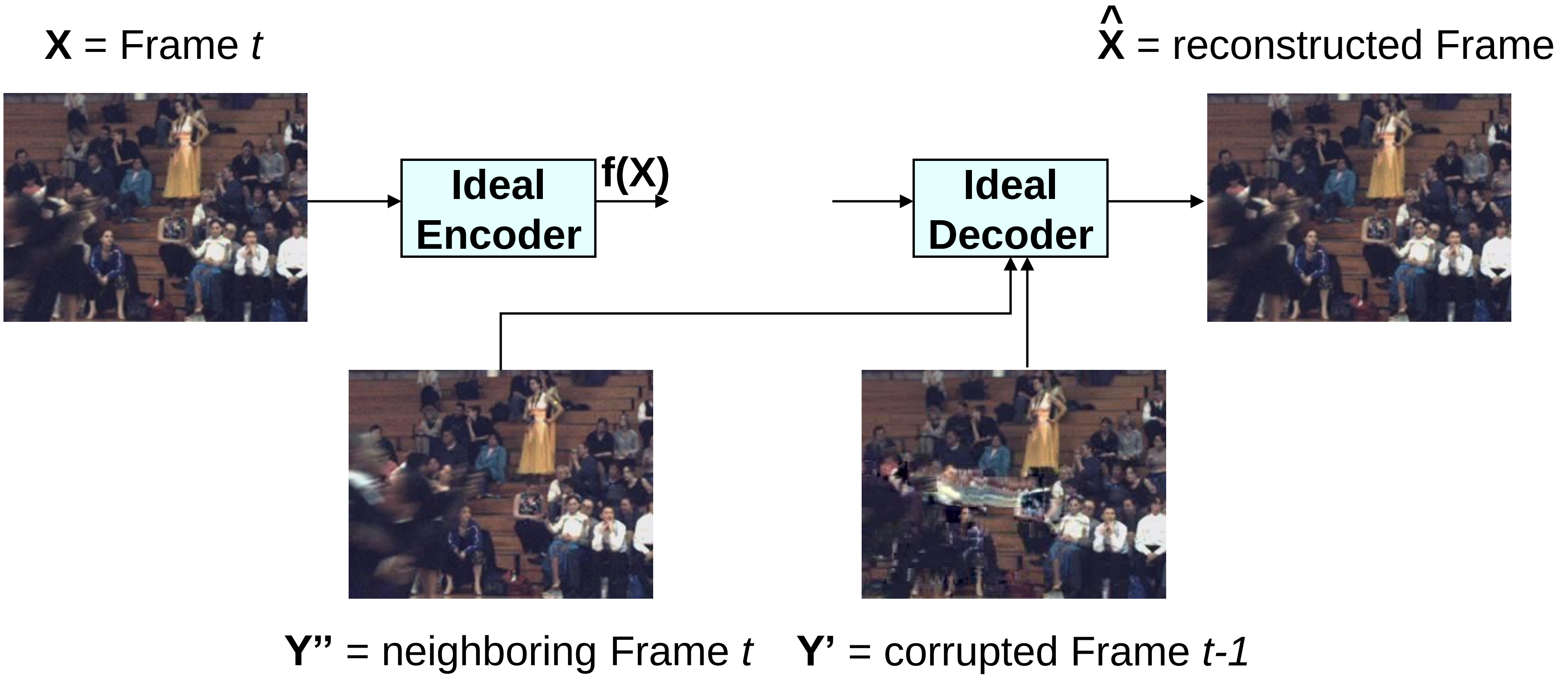
Active area of research

- Distributed multi-view image compression
  - Down-sample + Super-resolution *[Wagner, Nowak & Baraniuk, ICIP 2003]*
  - Geometry estimation + rendering *[Zhu, Aaron & Girod, SSP 2003]*
  - Direct coding of scene structure *[Gehrig & Dragotti, ICIP 2005] [Tosic & Frossard, ICIP 2007]*
  - Unsupervised learning of geometry *[Varodayan, Lin, Mavlankar, Flierl & Girod, PCS 2007]*
  - *…*
- Distributed multi-view video compression
  - Geometric constraints on motion vectors in multiple views *[Song, Bursalioglu, Roy-Chowdhury & Tuncel, ICASSP 2006] [Yang, Stankovic, Zhao & Xiong, ICIP 2007]*
  - Fusion of temporal and inter-view side-information *[Ouaret, Dufaux & Ebrahimi, VSSN 2006] [Guo, Lu, Wu, Gao & Li, VCIP 2006]*
  - MCTF followed by disparity compensation *[Flierl & Girod, ICIP 2006]*
  - *…*
- *Robust* distributed multi-view video compression
  - Disparity search / View synthesis search *[Yeo, Wang & Ramchandran, ICIP 2007]*

Robust distributed multi-view video transmission



Side information from other camera views

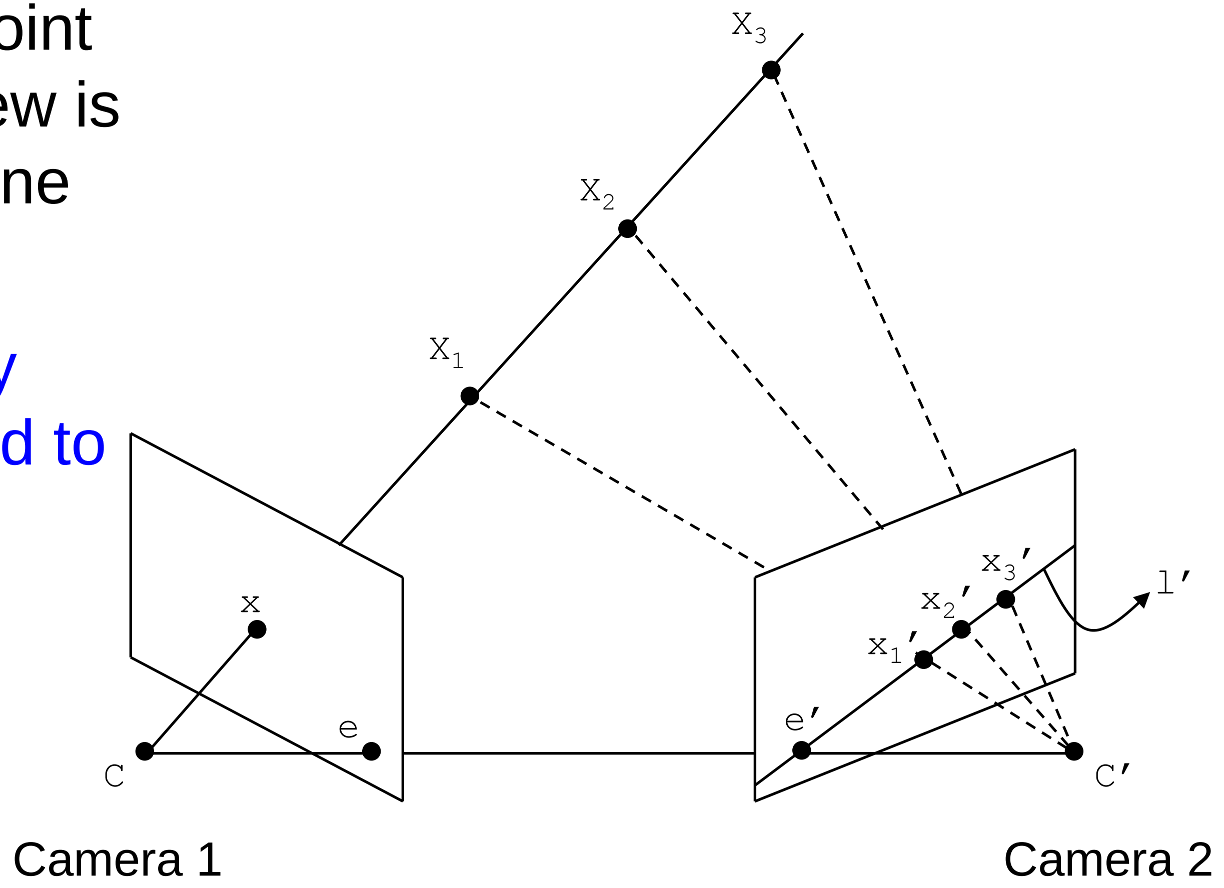


- How should we look in other camera views?
  - Naïve approach of looking everywhere can be extremely rate-inefficient
- Possible approaches
  - View synthesis search
  - Disparity search

Epipolar geometry

- Given an image point in one view, corresponding point in the second view is on the epipolar line

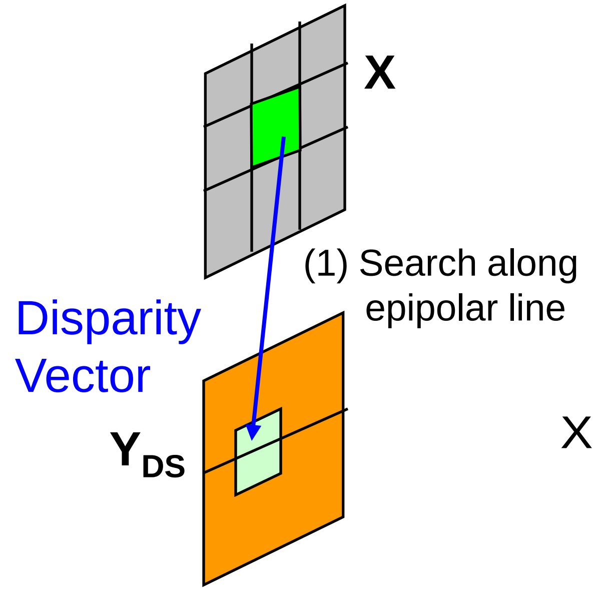
- Upshot: Disparity search is reduced to a 1-D search



Decoder disparity search

Camera 1

Temporal – Poor reference



$$X = Y_{DS} + N_{DS}$$

Camera 2

Spatial – Good reference

Frame *t-1* Frame *t*

- Extension of decoder motion search using epipolar geometry

[Yeo & Ramchandran, VCIP 2007]

PRISM-DS vs MPEG with FEC

- "Ballroom" sequence (from MERL)
  - 320x240, 960 Kbps, 30fps, GOP size 25, 8% average packet loss

Original

MPEG+FEC

PRISM-DS

- Drift is reduced in PRISM-DS

[Yeo & Ramchandran, VCIP 2007]

Summary and concluding thoughts

- Overview of distributed source coding
  - Foundations, intuitions and constructions
  - Application landscape

- DSC for video transmission
  - Compression of encrypted content
  - DVC for “single-camera” systems:
    - complexity and robustness attributes
  - DVC for multi-camera systems
    - truly distributed application

Lots of open challenges

- Core problems deeply intertwined
  - Side-information generation
  - Correlation modeling and estimation:
    - fundamental tradeoffs between encoding complexity, compression performance and robustness?
  - “Optimal” co-existence with existing standards?
  - …
- Multi-camera systems
  - Distributed correlation estimation among sources
  - Spatial versus temporal correlations
    - when will the correlation among sources dominate correlation within each source?
  - Interplay with wireless networking protocols?

THANK YOU!