Wireless video transmission

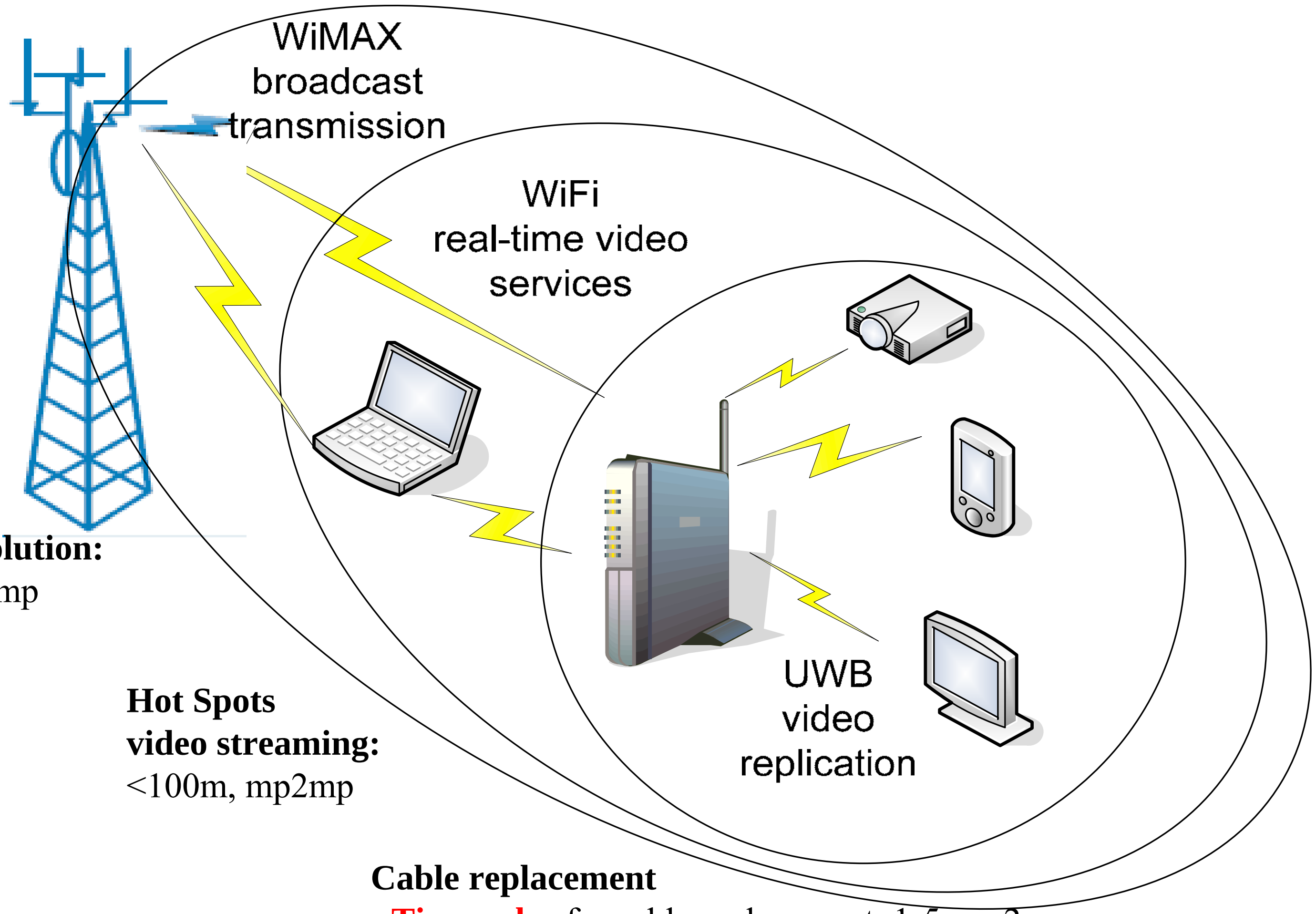
Requirements for compression algorithms

A. Sergeev

Outline

- Summary
- Scenario
- Goals
- Requirements

Video over wireless



Last mile solution:
< 10 km, p2mp

Hot Spots
video streaming:
<100m, mp2mp

Cable replacement
- **Tiny codec** for cable replacement: 1-5m, p2p
- Scalable & **distributed** compression: 3-20m, p2mp

Video over wireless: problem

- No support of video in wireless
  - Target of wireless PHY = to provide the required PER
    - it's obtained by reducing the transmission speed
    - this leads to delays, stops and drops in multimedia
- No support of wireless in video
  - Rate Control, Tiling, Streaming etc.

Video standards

Standard	QXGA	SXGA	SDTV 480i	HDTV 720p60	HDTV 1080i/60/2:1	standard DLP	JVC D-ILA	JVC D-ILA
type	graphics		video			digital cinema		
columns	**1280**	**2048**	**720**	**1280**	**1920**	**1280**	**2048**	**3840**
rows	**1024**	**1536**	**480**	**720**	**1080**	**1024**	**1080**	**2048**
pixels per frame	1310720	3145728	345600	921600	2073600	1310720	2211840	7864320
samples per pixel	3	3	3	3	3	3	3	3
bits per sample	8	8	8	8	8	8	8	8
bits per frame	31457280	75497472	8294400	22118400	49766400	31457280	53084160	1,89E+08
frames/sec	60	60	30	60	30	24	24	24
bits per second	1,89E+09	4,53E+09	2,5E+08	1,327E+09	1492992000	754974720	1,27E+09	4,53E+09
Gbps	1,887	4,530	0,249	1,327	1,493	0,755	1,274	4,530
CR to fit UWB 400 Mbs (3m LOS)	**4,72**	**11,32**	**0,62**	**3,32**	**3,73**	**1,89**	**3,19**	**11,32**
CR to fit UWB 180 Mbs (5m NLOS)	**10,49**	**25,17**	**1,38**	**7,37**	**8,29**	**4,19**	**7,08**	**25,17**

Compression is needed!

Goals

- Quality $\uparrow$
  - Target PSNR = ?
- Delay $\rightarrow$ min
  - Real time coding/decoding
- Complexity $\rightarrow$ min
  - Cheap device
- Transmission Protocol Compatibility
  - Compression Ratio = ?
- Adaptation for time varying channel
  - Rate Control

Coding algorithm requirements

- Features to be considered for wireless:
  - Compression efficiency (=RD function)
  - Cost of Implementation (=Complexity↓)
  - Progressive encoding (=Quality Scalability)
  - Priority coding (=streaming)
  - ROI support (+tiling)
  - Rate Control (pre- or post-compression)
  - Rate Adaptation
  - Network friendliness (=packetizing)
  - Error Resiliency & Concealment
  - Security issues (= DRM, watermarking)

Compression efficiency

- Rate-distortion function (PSNR vs bpp) for the selected rate/quality range
- Metrics
  - Traditional: PSNR
  - Human Visual System (=avoid drops + equal quality of tiles)

$$PSNR = 10\log_{10}\frac{I(\vec{x})_{\max}{}^{2}}{MSE} = 10\log_{10}\frac{(2^{n}-1)^{2}}{MSE}$$

$$MSE = \frac{1}{NM}\sum_{NM} e(\vec{x})^{2}$$

PSNR ~ 30 dB for both images

Cost of implementation

- Overall complexity for selected platform
  - Operational complexity
  - Memory consumptions
- Latency
  - Time from when a given pixel enters the system to when it exits the system
- Complexity balancing
  - Encoder/Decoder balance
- Distributed & Multi hop compression
  - recoding: recompression of compressed video without full decompression

Example: JPEG Complexity Estimation

Operational complexity

Step	JPEG Complexity	Tiny Codec Complexity	Comparison
Color Space Transform	576 muls + 384 adds + 128 shifts	576 muls + 384 adds + 128 shifts	**EQUAL**
DCT 8x8	54 muls + 462 adds + 6 shifts	54 muls + 462 adds + 6 shifts	**EQUAL**
Quantization	64 muls	64 shifts	**TINY CODEC WINS!**
Entropy coding (1)	126 Action 1 + 105 Action 2 + 0,2 Action 3 + 23 Action 4 + 21 Action 5 + 23 Action 6	126 Action 1 + 105 Action 2 + 0,2 Action 3 + 23 Action 4 + 21 Action 5 + 23 Action 6	**EQUAL**
Entropy coding (2)		126 Action 1 + 105 Action 2 + 0,2 Action 3 + 46 Action 4 + 21 Action 5	**TINY CODEC WINS!**
Entropy coding (3)		126 Action 1 + 23 Action 4 + 21 Action 5	**TINY CODEC WINS!**
TOTAL COMPLEXITY GAIN		**-126 A1 - 105 A2 - 64 muls (~ -60%)**	**TINY CODEC WINS!**

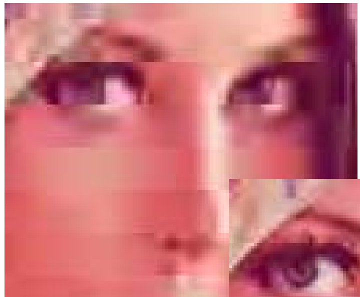
Memory consumptions

Step	JPEG Complexity	Tiny Codec Complexity	Comparison
Input Domain Storage	8 x 8 x 3 = 192 bytes	8 x 8 x 3 = 192 bytes	**EQUAL**
Color Space Transform	3 x 3 = 9 bytes	3 x 3 = 9 bytes	**EQUAL**
DCT 8x8 Domain Storage	8 x 8 x 3 = 192 bytes	8 x 8 x 3 = 192 bytes	**EQUAL**
Quantization	8 x 8 x 2 = 128 bytes	-	**TINY CODEC WINS!**
Entropy coding (1)	855 byte	855 byte	**EQUAL**
Entropy coding (2)		72 byte	**TINY CODEC WINS!**
Entropy coding (3)		24 byte	**TINY CODEC WINS!**
TOTAL COMPLEXITY GAIN		**- 983 byte (-70%)**	**TINY CODEC WINS!**

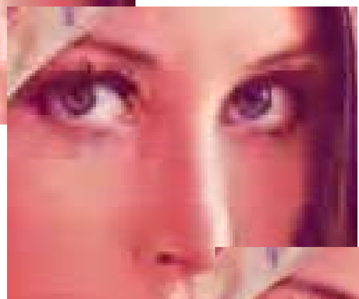
Progressive encoding & ROI

- Progressive encoding - data division into several streams, according to their contribution to image quality

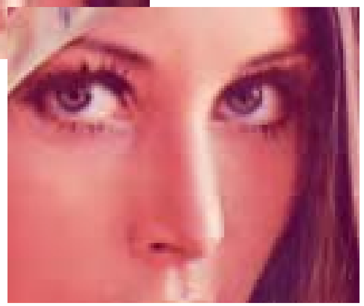
1st group
of packets



1st + 2nd
groups
of packets



1st + 2nd + 3rd
groups
of packets



- Region-Of-Interest Coding

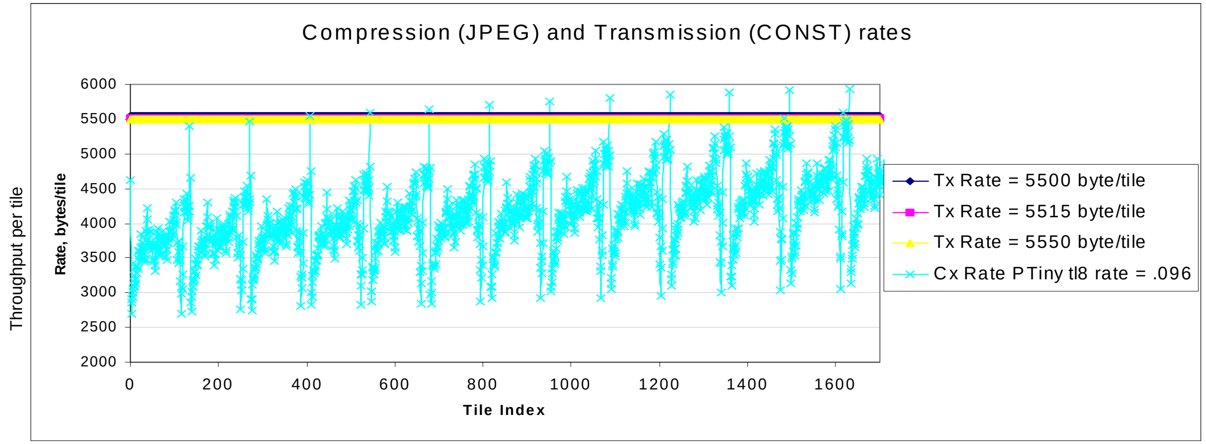
Error resiliency

- Markers
- Error-resistant encoding/decoding
  - Self-synchronizing Huffman codes
  - Arithmetic coding
- Unequal Error Protection (UEP)
  - Protection of headers
  - Unequal error robustness of encoded streams

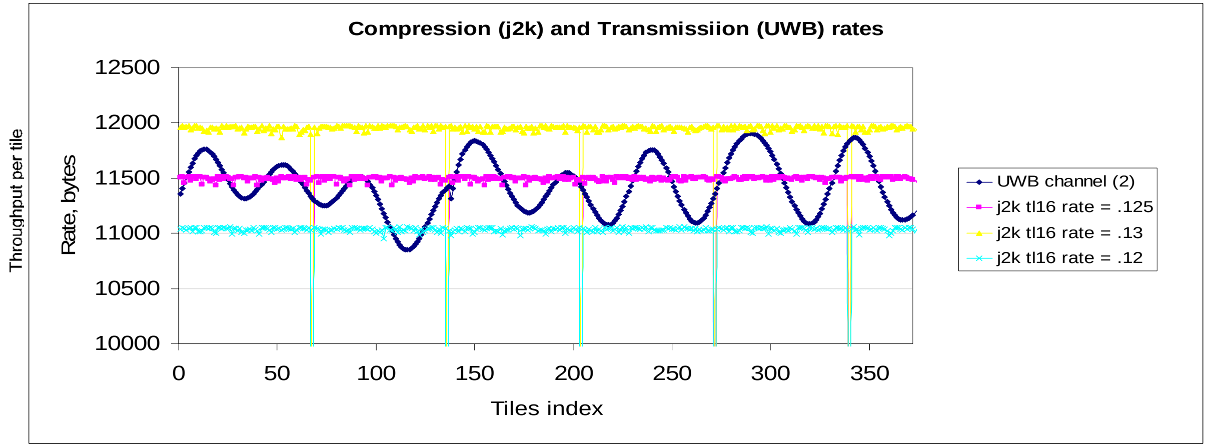
Rate Control

- Rate control in digital imaging is a technique of generating an optimal image subject to the bit-rate constraint: PSNR $\rightarrow$ max(Rate)
  - for all blocks of image, not in average only
  - may be pre- or post compression

Constant throughput

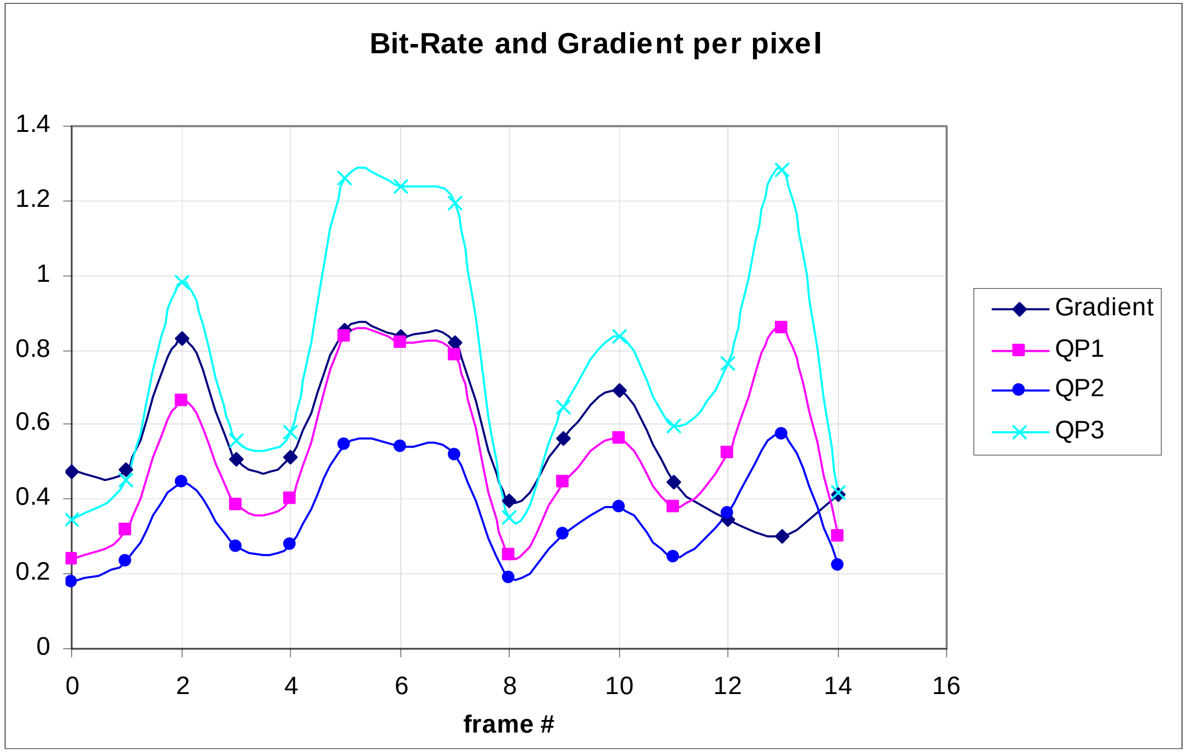


Variable throughput



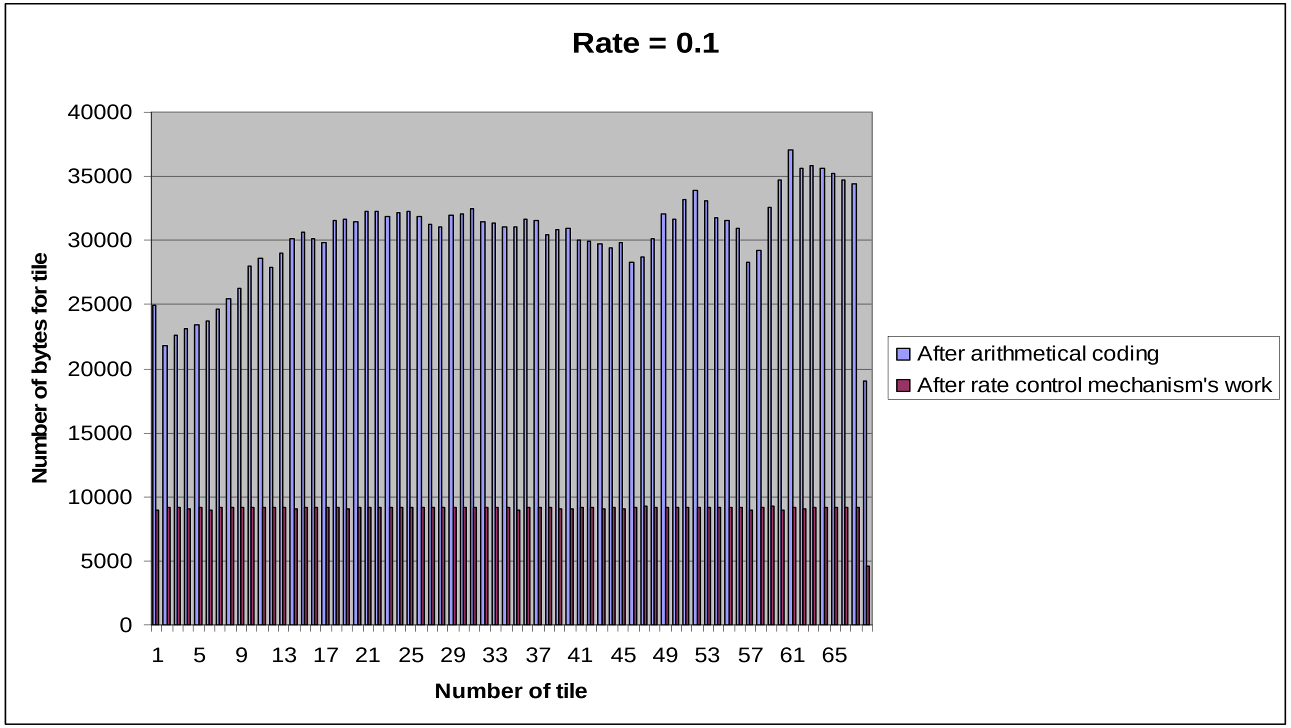
Pre-compression rate control

- Allows to define the most appropriate Quantizing Parameter (QP) before encoding the image (tile)
  - Example: gradient-based complexity measure
  - \+ Quite fast
  - – Not accurate: it's not guaranteed that the highest PSNR can be achieved for this rate even if QP is calculated considering previously encoded tiles



Post-compression rate control

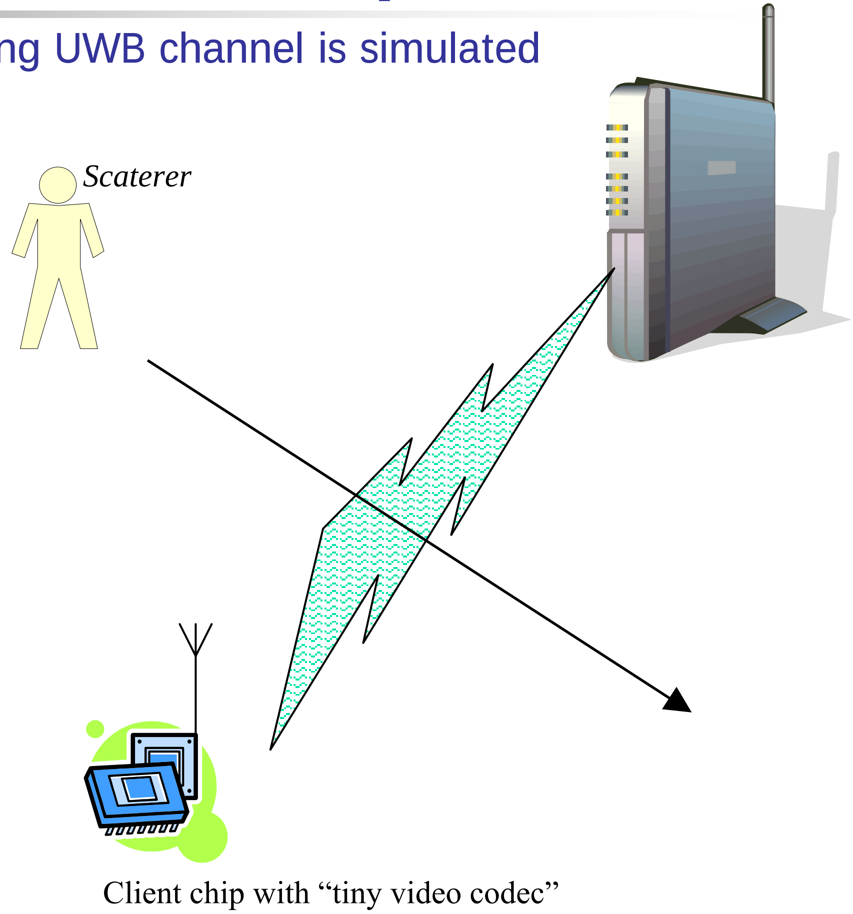
- Solves problem of PSNR maximizing for the required rate
  - Example: J2k Rate-Distortion Optimization
  - + Accurate: gets the maximum PSNR for the target rate
  - – Complexity: RDO needs all the compressed data. So context-based arithmetic coder encodes more than 70% of data vainly and wastes this time for nothing



Rate Detection & Adaptation

Video transmission over the time varying UWB channel is simulated

	1st & 2nd pkts successfully delivered	1st pkt only successfully delivered
SNR	34.23	27.00
PSNR	40.24	33.00



Rate adaptation and detection scheme is needed!

1st packet (NLOS)

1st + 2nd packet (LOS)

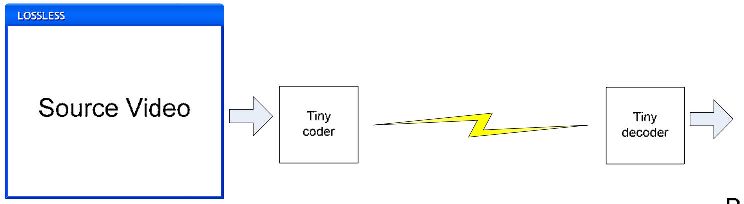
Rate Detection & Adaptation: WiFi Demo

Resolution Scale for Demo (WiFi:UWB) = 1:12

SCENARIO 1:

- WITHOUT Rate Switching

1



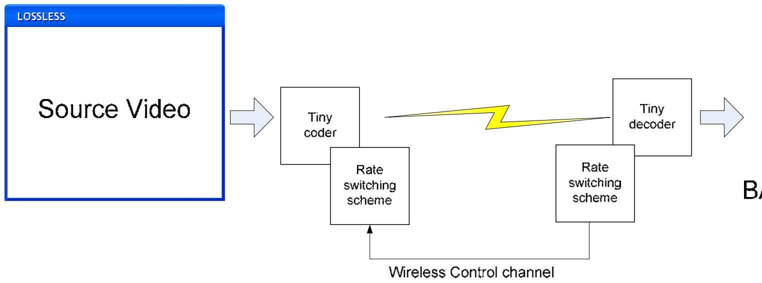
GOOD channel conditions
=> Hi Tx Rate
=> Low Delay
=> Normal FPS

BAD channel conditions
=> Low Tx Rate
=> Hi Delay
=> Small FPS, stops etc.

SCENARIO 2:

- RATE ADAPTATION
  - Hi → Low: FPS threshhold
    Tx says Rx to reduce Compression Rate
  - Low → Hi: add additional packets and check average delay between
    Tx says Rx to increase Compression Rate

2

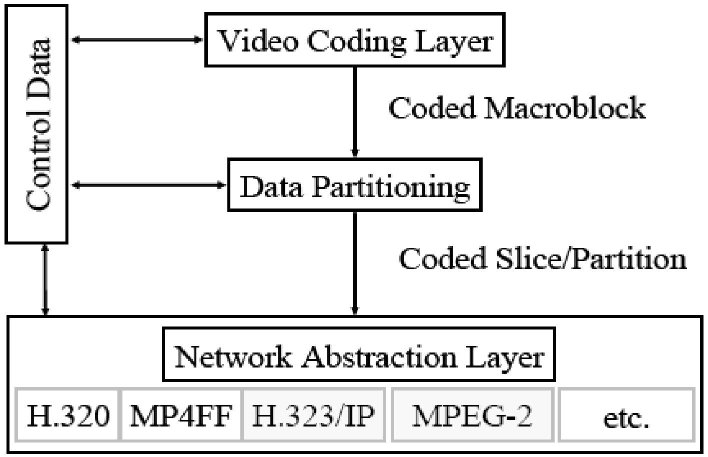


GOOD channel conditions
=> Hi Tx Rate
=> Low Delay
=> Normal FPS

BAD channel conditions
=> Low Tx Rate
=> Hi Delay
=> Normal FPS with
Decreased Quality

Network friendliness & Packetizing

- Network friendliness means a codec's ability to make a simple and effective customization of the use of the output bit stream for a broad variety of systems
  - packetizing into independent packets
    - J2k Tier-2, H.264 NAL
  - support of network interfaces
    - to map compressed data to transport layers of different protocols



Joint coding =

source + channel coding