26^{th} National Conference on Communications, 2020

On the Latency in Vehicular Control using Video Streaming over Wi-Fi

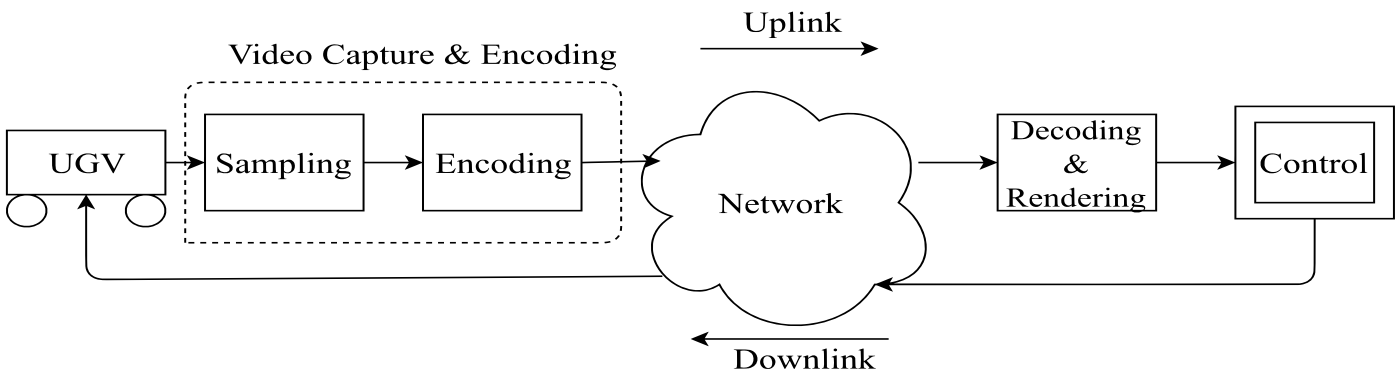
Presented by: Pratik Sharma

Joint work by:
Devam Awasare, Bishal Jaiswal, Srivats Mohan, Abinaya N., Ishan Darwhekar,
Anand SVR, Bharadwaj Amrutur, Aditya Gopalan, Parimal Parag, Himanshu Tyagi

Department of Electrical Communication Engineering,
Indian Institute of Science, Bangalore

Problem Statement

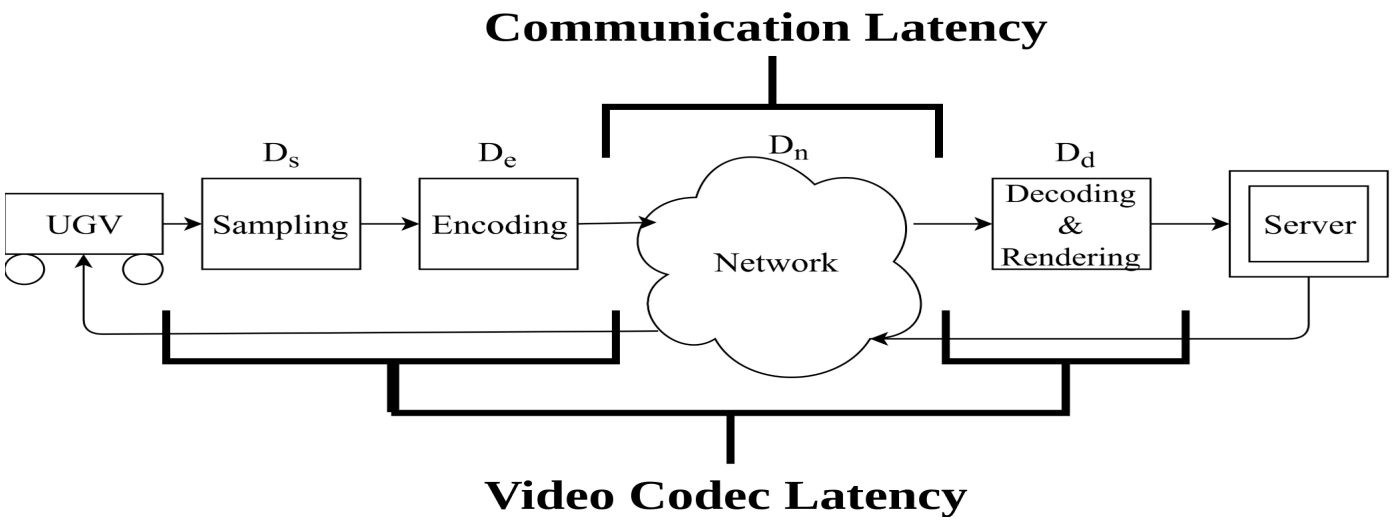
Use of Wi-Fi network for remote control of a vehicle using video transmission on the uplink and control signals for the actuator on the downlink.



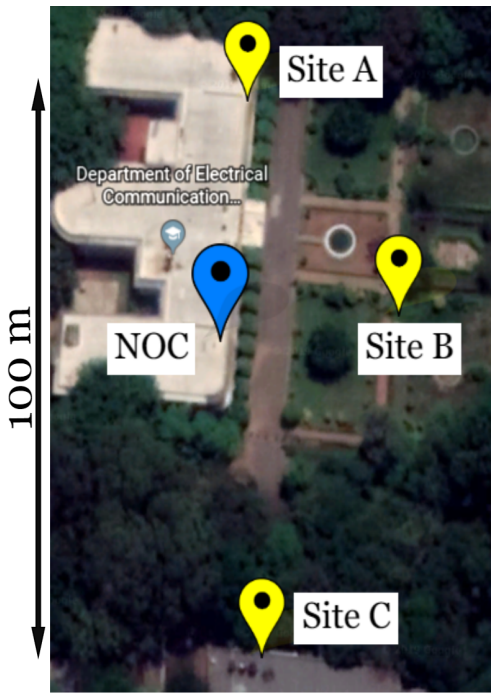
Block Diagram for communication between Unmanned Ground Vehicle (UGV) and central controller

Latency in the Setup

- Communication Latency - Due to mobility of the vehicle
- Video Codec Latency - Due to the processing and associated delays related to video transmission



Experimental Setup - Deployment



Wi-Fi AP Deployment

Outdoor deployed AP

Unmanned Ground Vehicle

Remote Driver

Experimental Setup - Specifications

- Access Points (AP) - Three Outdoor Wi-Fi (IEEE 802.11n) APs at distance of 50 m enabled with IEEE 802.11r (for roaming) using OpenWRT (Open-Source softare)
- Unmanned Ground Vehicle (UGV) (with 7MP camera) -
  - Raspberry Pi(RPi) 3B+ OR
  - Nvidia Jetson (TX2)
- Central Controller - Linux OS, i5 processor, and 8GB RAM
- IEEE 802.11p - Redundant Downlink -
  - Road Side Unit (RSU)
  - OnBoard Unit (OBU)
- Video Codec - Customized FFmpeg encoder with H.264 codec

Profiling

Scanning

- Scanning takes most of the handover time
- 'Channel Hold Time' reaches Max. 340 ms per channel
- Default Wi-Fi configuration: All 25 channels are scanned

Roaming

- Default RSSI is higher to invoke roaming around -90 dBm
- Default handover time is high ($\approx$42 ms)
- Default configuration is not optimized for mobility (swift handover)

Communication Latency - Quicker scanning process

Customization

Scanning

- Optimized 'Channel Hold Time' in the RPi's Wi-Fi driver Experimentally reduced from 340 ms to 14 ms per channel
- Selective scanning to avoid redundant channels (Reduced from 25 to 3 channels)

Delays during handovers

T_{max} (ms)	11	12	13	**14**	15	16	17	18	19
APs found	4	5	5	**7**	7	7	7	7	7
Total time (ms)	43	46	48	**52**	54	58	61	64	66

Customization

Roaming (Handover)

- Determined right signal strength to invoke handover for seamless video transmission(-68 dBm)
- Use of OpenWRT (Open-Source Software) to integrate 802.11r along with 802.11n (Reduced from 42 ms to 26 ms)

Commuication Latency - Redundant downlink

- Need for reliable and low latency delivery of control messages over downlink
- Standard based on DSRC (Dedicated Short Range Communication) with vehicular communication
- IEEE 802.11p - Association-less connectivity between access point (RSU) and UGV (OBU)
- Downlink latency $\approx$ 1.5 ms
- Practical implementation with dedicated hardware

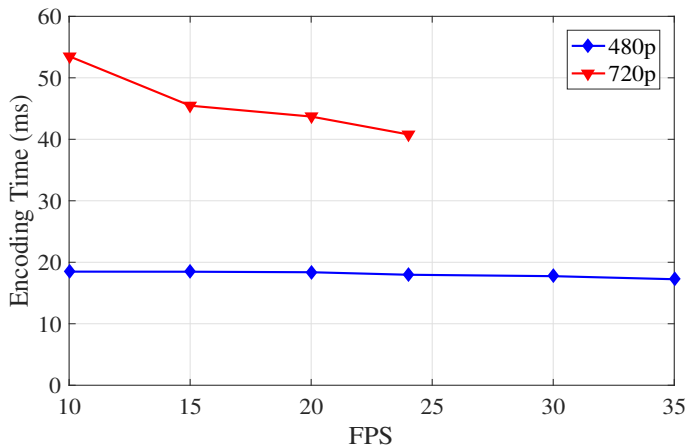
Is communication link the only bottleneck in end-to-end latency?

Video Codec Latency - Frame capture rate

Profiling

Sampling

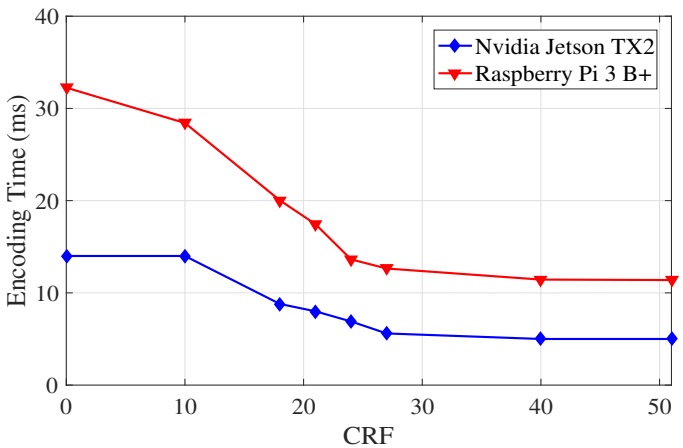
- Frames per second (FPS) - Higher frame capture rate leads to lower sampling delays
- 30 FPS translates to 33.3 ms (1/30 s) delay

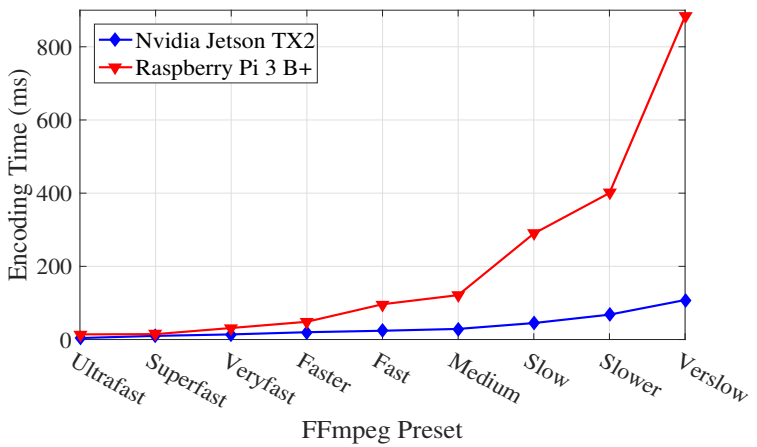


Profiling

Encoding

- Constant Rate Factor (CRF) - Scalar value - 0 (Lossless) to 51 (Highest compression)
- FFmpeg Preset - Unique collection of settings for video encoding



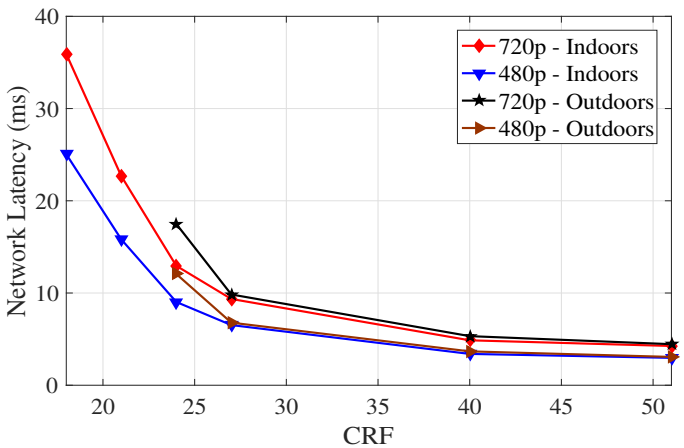


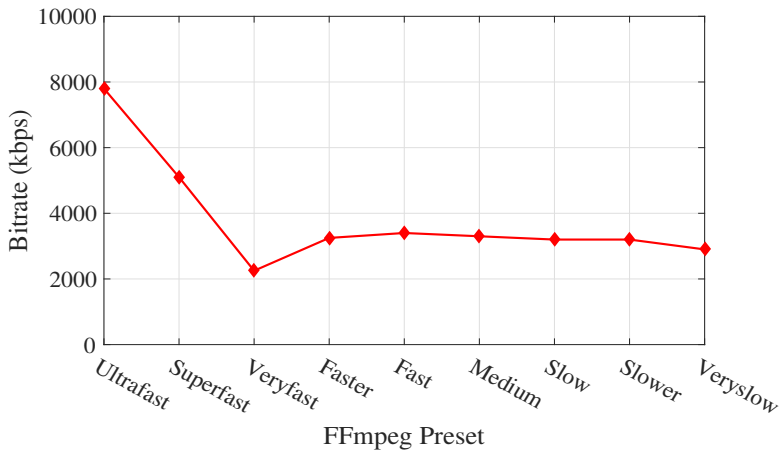
Video Codec Latency - Effects of encoding

Profiling

Network

- Stochastic component in latency dependent on channel conditions and other parameters
- Affected by change in the encoding parameters values





Profiling

Decoding & Rendering

- Frames in buffer can build up queue at the receiver and add to delay (upto 300 ms)
- Queuing of frames can lead to jittery video

Processor Type

- Computational power of processor important
- TX2 has lower encoding time than RPi because of higher processing power

Video Codec Latency - Optimized codec parameters

Customization

FPS

- 24 FPS at 720p or 30 FPS at 480p
- Consideration given to the encoding delays

CRF

- 28 considered (Imperceptible change between 28 to 35)
- Tradeoff between network latency and video quality

Customization

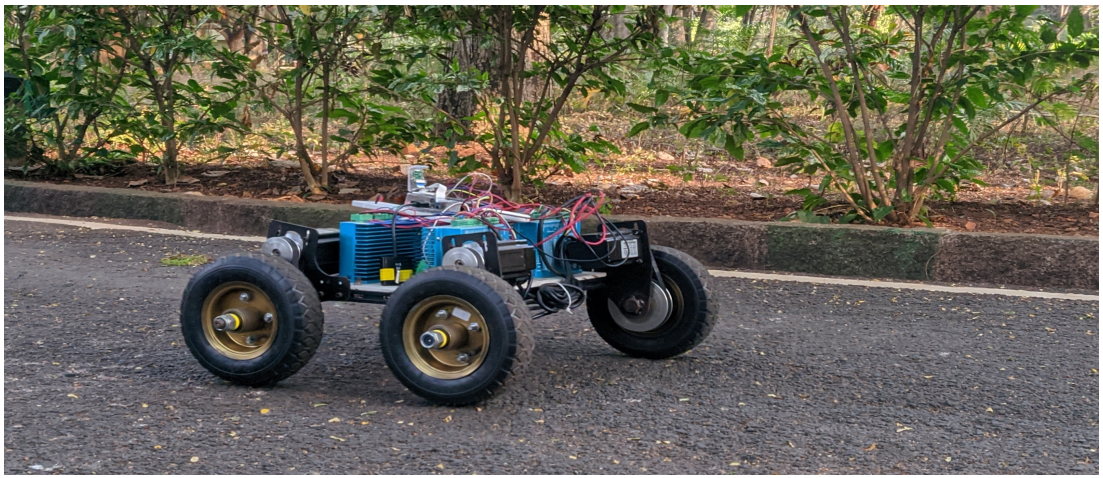
FFmpeg Preset

- 'Veryfast' preset selected as a trade-off between encoding time and video quality
- 'Faster' option can also be selected for system with higher compute

Decoding & Rendering Algorithm

- Reduction in decoding frame buffer from 3 to 1
- Reduced the rendering time around 150 ms

Outdoor Testing



Field View of UGV

Camera feed

Command Sequence

Results - Optimized handover process

Delays during handovers

	Default (ms)	**Optimized (ms)**
Scanning	143.88 $\pm$ 9.76	54.5 $\pm$ 4.47
Roaming	41.75 $\pm$ 8.01	26 $\pm$ 8.33
TOTAL	**186.63 $\pm$ 12.32**	**80.8 $\pm$ 8.53**

- Scanning takes 75% of the handover time
- 60% reduction in handover time after customization

Results - Trade-off among components

Optimized Latency values for video transmission over uplink

	Latency value (ms)
Maximum Sampling	33.33
Encoding	13.8 ± 2.79
Network	12.4 ± 3.825
Decoding & Rendering	12.16 ± 3.03
TOTAL	$\mathbf{71.68 \pm 5.31}$

- Optimum configuration:
  Raspberry Pi, 30 FPS, 28 CRF, 'Veryfast' Preset, 480p Video Resolution, Decoder buffer size $= 1$

Results

End-to-End Latency Measurements

	Default		Optimized	
	Regular operation (ms)	**Handover (ms)**	**Regular operation (ms)**	**Handover (ms)**
Uplink	210 ± 16.83	396 ± 12.14	71 ± 5.31	149 ± 5.85
Downlink	12 ± 3.45	198 ± 7.55	9 ± 2.93	89 ± 5.67
Processing	13 ± 1.44	13 ± 1.44	13 ± 1.44	13 ± 1.44
TOTAL	$\mathbf{235 \pm 16.91}$	$\mathbf{606 \pm 15.25}$	$\mathbf{93 \pm 5.89}$	$\mathbf{251 \pm 7.52}$

- Processing Time: Emergency Braking Experiment - Time take to detect a RED signal and automatically execute STOP command
- 60%, 54% reduction in end-to-end latency during regular operation and handover respectively

Recent Work

Implementation with Cellular-V2X

- LTE Deployment

LTE eNB (Base Station)

LTE Remote Radio Head (RRH)

- Ultra-Reliable Low-Latency Communication over 5G

Thank You!