Adaptive Real-Time Video For Vehicle Remote Control

Caterpillar/Bradley Alliance Project

Team: Zuguang Liu, Sai Rohith Vavilala, Venkata Varaha Yatesh Makena Advisor: Dr. Yufeng Lu

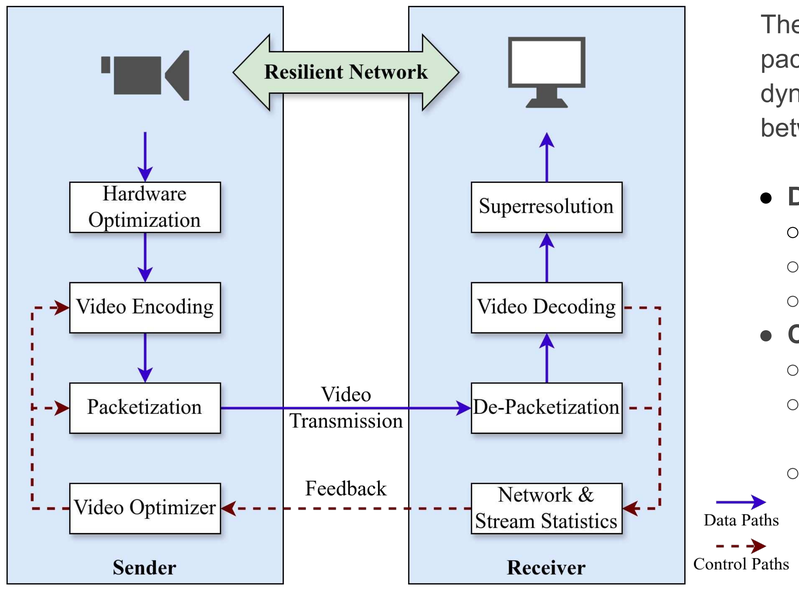
Department of Electrical and Computer Engineering

Objective

When teleoperation technology is used in piloting mining and construction vehicles, one of the critical factors is maintaining **clear, stable and responsive machine vision** from vehicle camera to operator control station

- Design, develop, and benchmark a system capable of delivering high-definition video streams in real time
- Adjust its video parameters in response to unpredictable network conditions

System Design



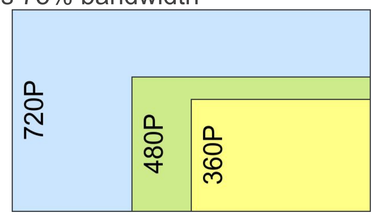
The optimal video encoding and packetization configuration is dynamically updated for balance between quality and speed

- **Data paths**
  - From sender to receiver
  - Carries raw video data
  - High bandwidth and frequency
- **Control paths**
  - From receiver to sender
  - Carries feedback & control parameters
  - Low bandwidth and frequency

Super Resolution

Advanced resolution upscaling method that potentially saves video transmission bandwidth

- 480P from sender → 720P at receiver (1.5x) saves 56% bandwidth
- 360P from sender → 720P at receiver (2x) saves 75% bandwidth



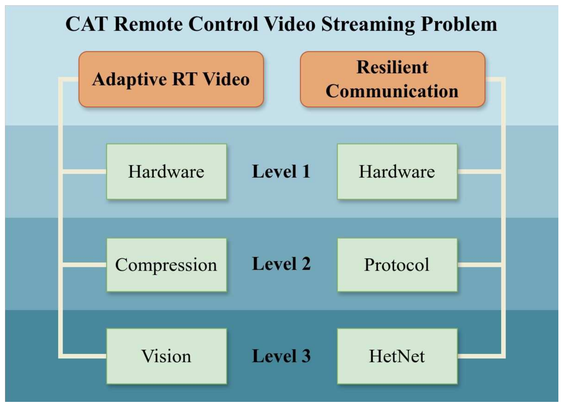
We investigated three different SR methods

- Pre-trained FSR-CNN models
- NVIDIA Image Scaling
- AMD FidelityFX Superresolution

With MPV shader implementations, all SR methods measure **≤ 10 ms**

Research Directions

Members of this team is working closely with Resilient Communication team for one integrated system



Hardware

Camera for surround view industrial systems

- Full HD resolution
- Up to 60 FPS
- Uncompressed video
- GMSL2 interface

NVIDIA Jetson Orin

- High-performance embedded SBC
- GPU in SoC
- Linux OS and SDK readily available

Compression

H264 (AVC)

- Inter- and intra-frame prediction
- Used by 91% of video industry, including Blu-ray discs and streaming services

H265 (HEVC)

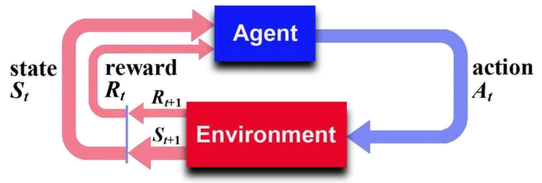
- Successor of H264
- 25% to 50% better data compression
- Costs relatively higher processing

Implementation uses **FFmpeg**, an open source software for video processing

When compress 1080P webcam to 360P 20FPS, we were able to get **sub 10 ms latency** with both H264 (with one thread) and H265 (with hardware acceleration)

Future Work

- Accurate end-to-end frame latency benchmarking
- Research advanced video parameter optimization methods (for example, reinforcement learning)



Acknowledgements

Special thanks to Karl Weyeneth for providing guidance on our project and how it can apply to industry use