Data Science Institute
COLUMBIA UNIVERSITY

COLUMBIA | ENGINEERING
The Fu Foundation School of Engineering and Applied Science

Challenges in meeting latency targets of a computationally demanding smart city application: Object detection in COSMOS smart intersections

Columbia iCSE Research Day – 5/12/2021

initiative for Computational Science and Engineering

Zoran Kostić, Professor of Professional Practice
Electrical Engineering Department & Data Sciences Institute
Columbia University, New York City

COSMOS

RUTGERS | COLUMBIA UNIVERSITY IN THE CITY OF NEW YORK | NYU | NYC | <sh> | The City College of New York | IBM

COSMOS Smart City Intersection Research Contributors

Zoran Kostić
Gil Zussman
Ivan Seskar
Jennifer Shane
Jakub Kolodziejski
Michael Sherman

Students

Zhengye Yang
Mingfei Sun
Zihao Xiong
Hongzhe Ye
Dwiref Oza
Mahshid Dehkordi
Vedant Dave
Zhuoxu Duan
Emily Bailey

Real-Time in COSMOS Applications

Applications of Artificial Intelligence / Machine Learning

COSMOS Smart City Traffic Intersection

Key considerations are:

- Low Latency
- High Bandwidth
- Edge Computing
- Low Power
- Privacy preserving

Approach: Experimental studies on a live testbed in NYC - COSMOS pilot site.

COSMOS Project Vision
Testbed for Advanced Wireless Research

Data Science Institute
COLUMBIA UNIVERSITY

COLUMBIA | ENGINEERING
The Fu Foundation School of Engineering and Applied Science

- **Ultra-high bandwidth**, **low latency**, and powerful **edge computing** will enable important new classes of **real time applications**
- **Application domains** include **AR, VR, connected car, smart city** (with high-bandwidth sensing), **industrial control**, …

Augmented Reality

Smart City + Connected Car

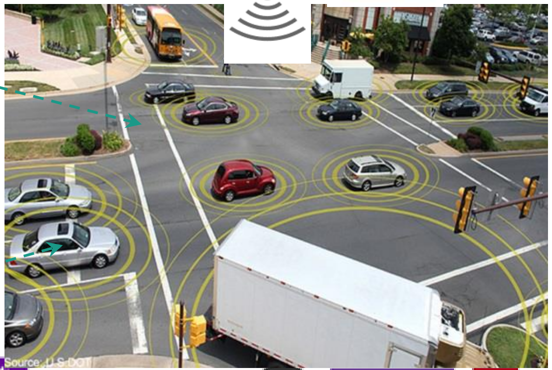
Image/
Video

Roadside
AP

Cloud
Infrastructure

Roadway sensors & lighting

In-car guidance display



Industrial Control

COSMOS

RUTGERS

COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

NYU

NYC

<sh>

The City College of New York

IBM


COSMOS Deployment Map
Morningside Heights
Manhattanville
Site Ownership:
DOE
CCNY
CU
NYCHA
Crown Castle
Existing Fiber Plant:
Nysernet
ZenFi
CUIT Network
Deployment
Large node: rooftop deployments with number of antennas and edge cloud
Medium node: street-level devices dual use with wireless/wired backhaul
Control Center: CRAN, control, management and operations center

COSMOS Application: Smart City Intersection

Metropolises such as Manhattan have complex traffic environments

Individual autonomous cars have limited situational awareness

Autonomous Vehicles in a Metropolis?

City roads are complicated

Some important data is not accessible to individual vehicle's sensors.

Pedestrians are unruly.

Manhattan

COSMOS Application: Smart City Intersection

A view from a single autonomous car needs to be enhanced by:

→ Inputs from sensors from the city infrastructure (V2X, X2V...)

→ Data exchange among vehicles (V2V)

→ Autonomous Vehicles -> Cloud Connected Vehicles

Smart City Intersection has to support:

- low latency high bandwidth wireless communications (5G+)
- edge-cloud computing
- machine intelligence (ML/AI)

→ COSMOS node as a "center of intelligence" of a smart city intersection ←


COSMOS pilot site
0:02:31

Data Science Institute
COLUMBIA UNIVERSITY

COLUMBIA | ENGINEERING
The Fu Foundation School of Engineering and Applied Science

Columbia University NYC

COSMOS pilot site

Amsterdam Avenue and 120th St. Northeast corner of Mudd Engineering building.

Smart Intersection

Radios, cameras, edge computing node: GPUs and FPGAs.

COSMOS

RUTGERS
COLUMBIA UNIVERSITY IN THE CITY OF NEW YORK
NYU
NYC
The City College of New York
IBM

Use Case:
Interaction with Cloud Connected Vehicles

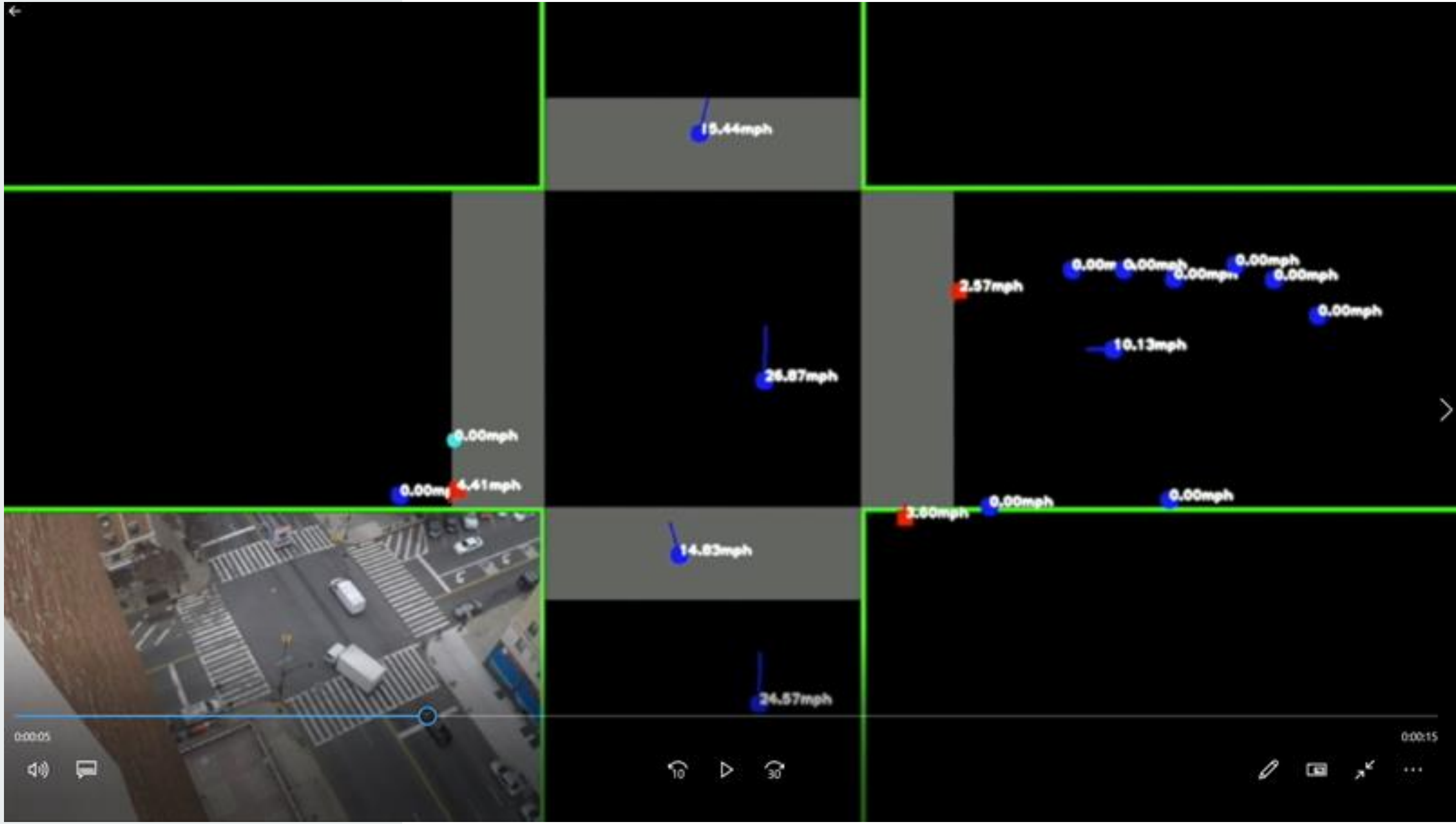
Collect data using sensors (cameras et al.) to identify all vehicles and pedestrians,

compose a “radar map” of those objects,

and broadcast the map to all participants in the intersection,

in real time

Radar Map



Broadcast to all:

from the computing node to vehicles and to pedestrians,

...in real time.

Video link.

Complexity of the Use Case

Collect data using sensors (cameras et al.) to identify all vehicles and pedestrians,

    latencies in video encoding, communications, streaming, decoding

compose a "radar map" of those objects,

    compute complexity of deep-learning based object detection and tracking

and broadcast the map to all participants in the intersection,

    closing the communications loop

    with acceptable power consumption

(doing all of the above) in real time is challenging.

What is "Real Time"?

Detect and track vehicles and pedestrians.

Provide feedback to participants in the intersection in real time.

Real time for a smart intersection:

- has to be fast enough to support traffic interaction/management.

Consider a vehicle moving at 10 kilometers per hour (km/h):

- 6.2 miles per hour
- 2.778 ~ 3 meters per second

How far does a vehicle move in (1/30) of a second (1 frame of a video), at 10km/h:

- 3m/s / 30 = 0.1m

Arguably useful to prevent accidents

Target round-trip latency = (1/30) second

What Contributes to Time Consumption?

Focus on “inference”, after the system has been “trained”:

- Desired goal for closing the loop: 1/30 seconds = 33.3 ms.

Camera (sensor) data acquisition and transmission:

- Sensing, video encoding, RTP/RTSP streaming

Communications:

- Edge internet infrastructure, switching, protocols

Edge computing server - data processing:

- Video decoding and pre-processing, AI-based object detection and tracking, short and long term feature extraction

Completion of the closed loop.

Video Acquisition and Transmission

Typical contemporary high-quality (surveillance) camera is IP-based and uses video compression. Can it meet the demands of real-time?

Stacked CMOS Image Sensors for pixel value acquisition are fast:

- Sony IMX532 - 5328 (H) x 3040 (V)
- 12-bit pixel value
- 109fps -> less than10ms

https://www.sony.net/SonyInfo/News/Press/201910/19-098E/

Video Acquisition and Transmission

Typical contemporary high-quality (surveillance) camera is IP-based and uses compression. Can it meet the demands of real-time?

Video coding/decoding for compression can impose arbitrary delays due to:

- temporal encoding of (I,P,B) frames
- vendor-specific construction of buffers
- user specified mode of operation

One can see latencies of 1 second and more.

Video Acquisition and Transmission

Typical contemporary high-quality (surveillance) camera is IP-based and uses compression. Can it meet the demands of real-time?

RTP/RTSP streaming and higher-level protocols:

- Highly dependent on the tool setup
- Can impose multiple-second latencies

What is Low Latency for Video

Function of encoding formats and video streaming protocols: buffering, CDN buffering, connection type, adaptive bitrate streaming , bandwidth, video player itself can significantly burden data transmission.

Video streaming community (very active in 2020 due to COVID):

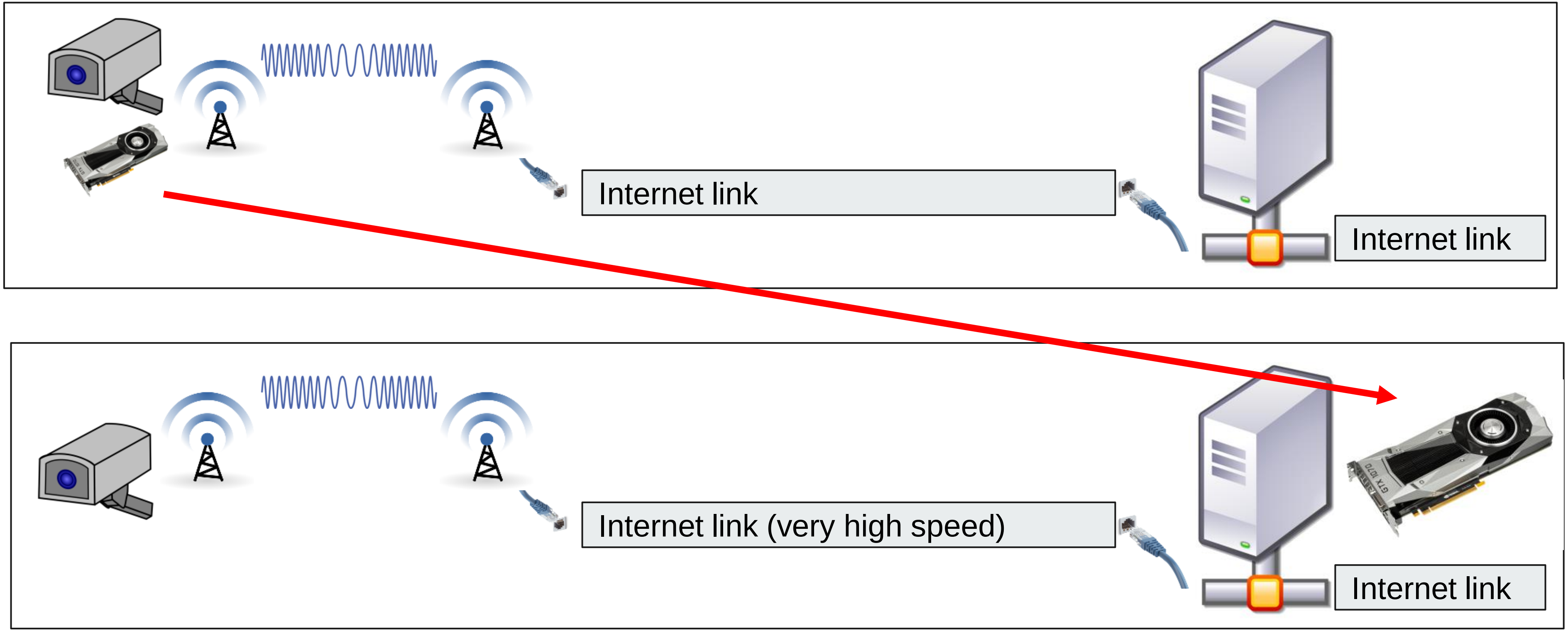
- Standard video latency (+20 seconds); Reduced video latency (20-5 seconds, HLS or DASH); Low video latency (5-1 seconds, LLHLS, LLDASH, SRT, and RTP/RTSP), Ultra-low video latency or near-real-time (500 ms, WebRTC)

Encoding experts (example Fraunhofer Heinrich Hertz Institute):

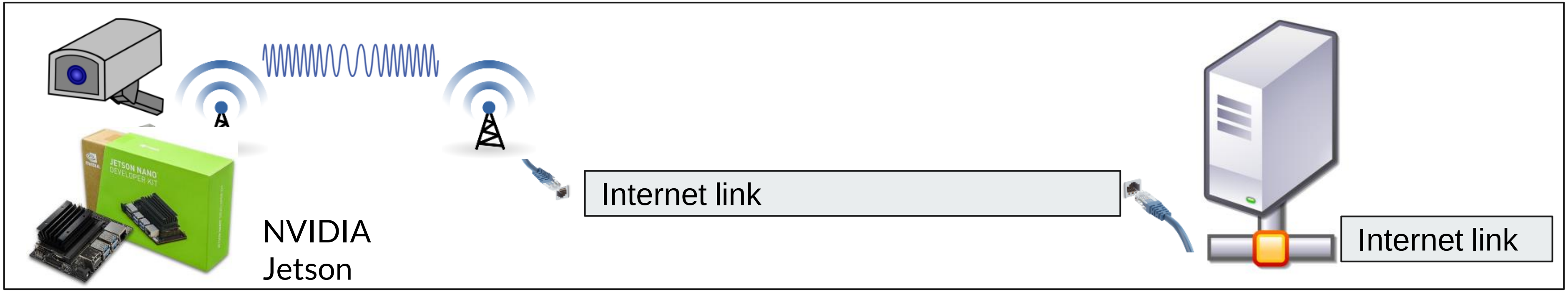
- H.264, 1080p, VHDL for FPGA/ASIC, 170Kgates, 0.18uM
- (all) I-frames only -> less than one macroblock line (at least 3ms)
- https://www.hhi.fraunhofer.de/en/departments/vca/technologies-and-solutions/h264-avc/h264-ultra-low-latency-video-codec.html

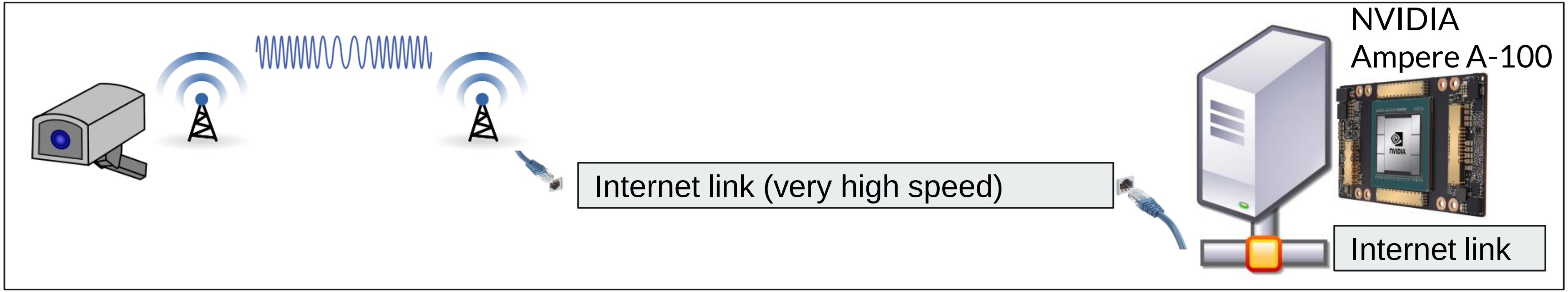
Rough Timing Check

AI Computing at The Edge, or in The Edge Cloud



AI Computing on The Edge, or in The Edge Cloud





Power Consumption vs. DL Inference Needs, NVIDIA Jetson Nano

- 8 W is significant
- PoE is 12.5 W

- Other NVIDIA GPUs
  - 10 times as much power

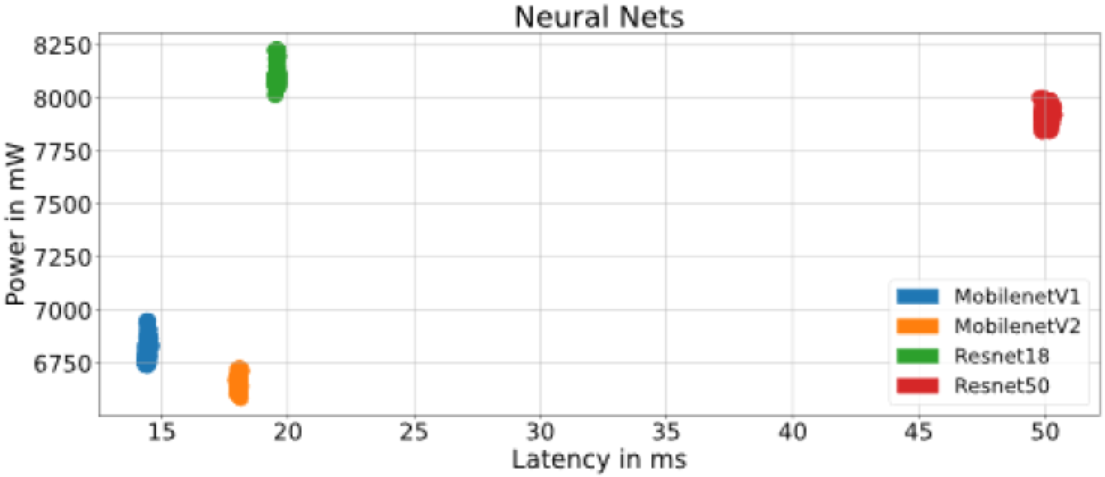


Figure 8. Power over latency of different Nets

Profiling Energy Consumption of Deep Neural Networks on NVIDIA Jetson Nano, https://publik.tuwien.ac.at/files/publik_293778.pdf

Bird's Eye Video

Privacy preserving
Very small pedestrians

Video link

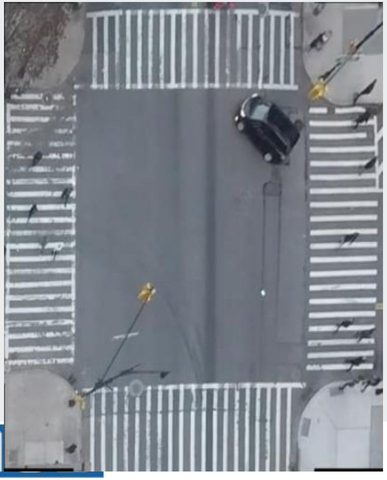
Detection & Tracking Custom Dataset

COSMOS pilot site

Intersection of 120th street and Amsterdam Avenue, New York City.

Bird's eye videos (friendly for privacy and security - IRB issue)

- Recorded over 100 hours videos with different light and weather conditions
- Calibrated to 90 degree bird's eye view
- Manually annotated more than 10,000 frames



Detection/Tracking Accuracy vs. Inference Speed

Customized/improved off-the-shelf deep learning models:
Compressed 1080p, and 1080p cropped to 832x832

- Accuracies for Yolo V4

model version	AP(Vehicle)	AP(Pedestrian)	mAP
832 version	0.8587	0.5665	0.7126
1080P version	0.8176	0.5457	0.6781

- Speed for Yolo: Resolution-driven, running on NVIDIA Deepstream
  - 832x832 -> 43.33 fps
  - 608 x 608 -> 71.42fps
  - 416 -> 123.87fps
- Latencies are still being measured→ fixed delays

Video Transmission Bandwidths

1080p video: 1920x1080 pixels, 24 bits per pixel, 30 frames per second

Bandwidth for raw 1080p:

- **Single image** → Raw bits = 1920x1080x24 = 49,766,400 bits **~50Mbp**
- **Video at 30fps** → 30*49,766,400 = 1,492,992,000 bits/second **~1.5Gbps**

Compressed/coded 30fps video

- Coded/compressed ~ (average) up to **12Mbps**
- Compression ratio of ~ 1 : 125

Video Transmission Bandwidths

1080p video: 1920x1080 pixels, 24 bits per pixel, 30 frames per second

- **Compare 1.4Gbps** and **12Mbps.**

The challenges are:

- Challenge for 5G+:
  - How high can transmission bandwidth be?
  - Video streaming protocols may impose latencies which are not as low as needed.
- Or challenge for coding/compression - change the paradigm - combined/adaptive feature extraction + compression

Tracking Example

Video link

Key Technical Components

High bandwidth:

- High bandwidth wireless communication, high bandwidht video
- Interfaces between wireless and fixed communications

Inference speed and low latency in communication and computing

- Low latency communication
  - Sensor -> computing resource interfaces and protocols
- Low latency computing -> high speed parallel computing
- Timing control - watchdog timers

Power consumption budget

- Limited by PoE on the edge (12.5W and maybe 25/50W)
- As needed for high bandwidth video source

Thanks!

-