Decentralized modular architecture for live video analytics at the edge

Sri Pramodh Rachuri
srachuri@cs.stonybrook.edu
Stony Brook University

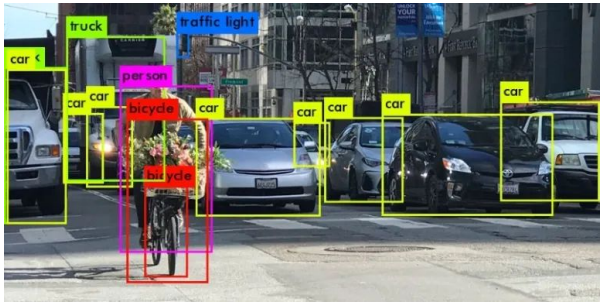
Francesco Bronzino
francesco.bronzino@univ-smb.fr
Université Savoie Mont Blanc

Shubham Jain
jain@cs.stonybrook.edu
Stony Brook University

Why Edge for Video Analytics?

- Increase in CCTV cameras
  ⇒High data influx
  ⇒More video analytics pipelines
  - for safety, security and traffic control

- Why not continue using cloud?
  - Network Congestion
  - Real-time requirement

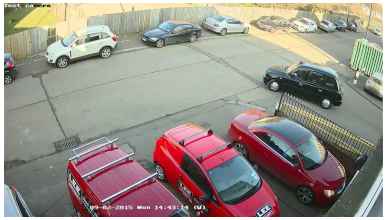


Challenges

- Scenes observed by cameras change over time
  - Lighting conditions
  - Visibility
  - Traffic conditions

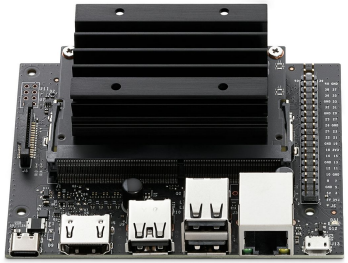
- Mobility of cameras
  - More information
  - Blind spots

⇒Changing network conditions

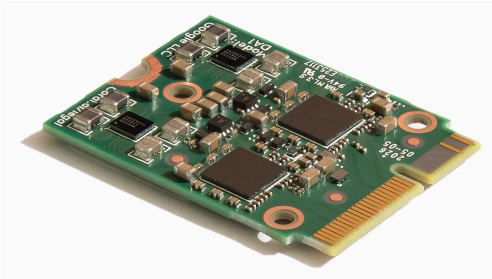


Edge deployment challenges

- Resource constraints
- Video analytics ⇒ GPU
- Heterogeneity
  - Accelerators - GPU, TPU
  - New nodes with new technologies like FPGA, ASIC
  - Upgrading ⇒ Gradual roll-out



- Distribution of load



Prior works

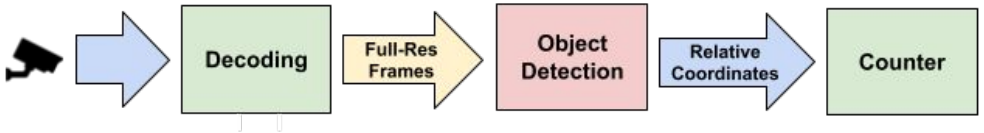
- Rocket — Microsoft Research Blog, 2020
  - Live video analytics
  - Pipeline with pluggable models
  - Offload to Azure cloud
- Spatula — SEC, 2020
  - Cross-camera analytics
  - Temporal and spatial correlations
- Chameleon — SIGCOMM, 2018
  - Adaptation to scene of video stream - Accuracy vs speed
  - Adaptation using cross camera inference
- JCAB — INFOCOM, 2020
  - Optimize config and bandwidth allocation
  - Network conditions, Energy Util, Processing latency and video scene
- Hetero-Edge — INFOCOM, 2020
  - Distributes tasks and exploit concurrency
  - Not decentralized
- VideoEdge, Follow Me at Edge — SEC, 2018;JSAC, 2018
  - Task placement and migration in mobile cameras

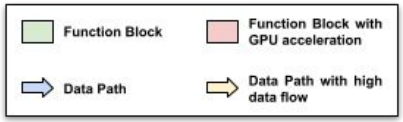
Design goals

1. Vision pipeline modularity
2. Improved latency and resource utilization
3. Adaptability

Vision pipeline modularity: Split-process execution

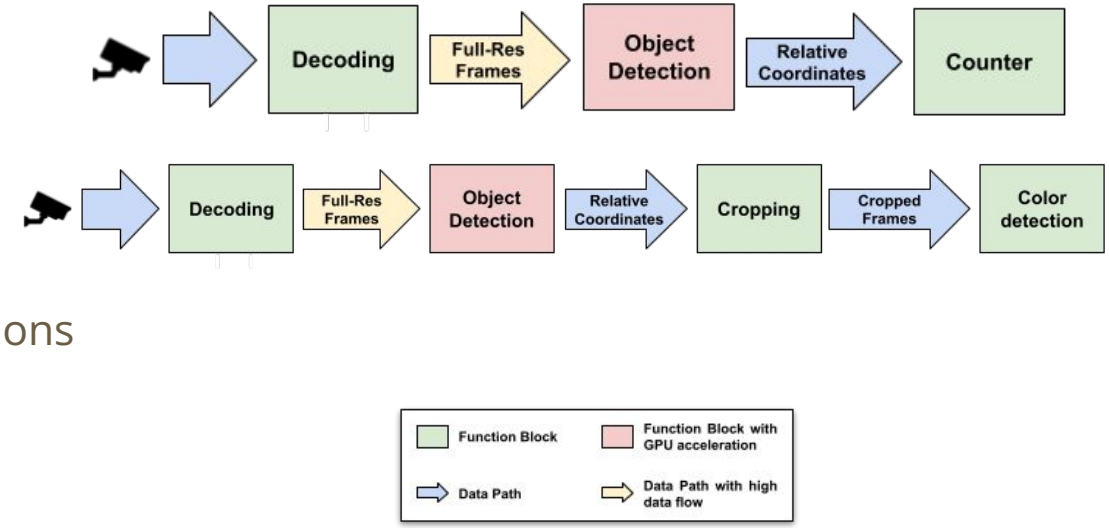
- Processing is sequential
- Each block as an independent microservice
- Easy addition of new functions





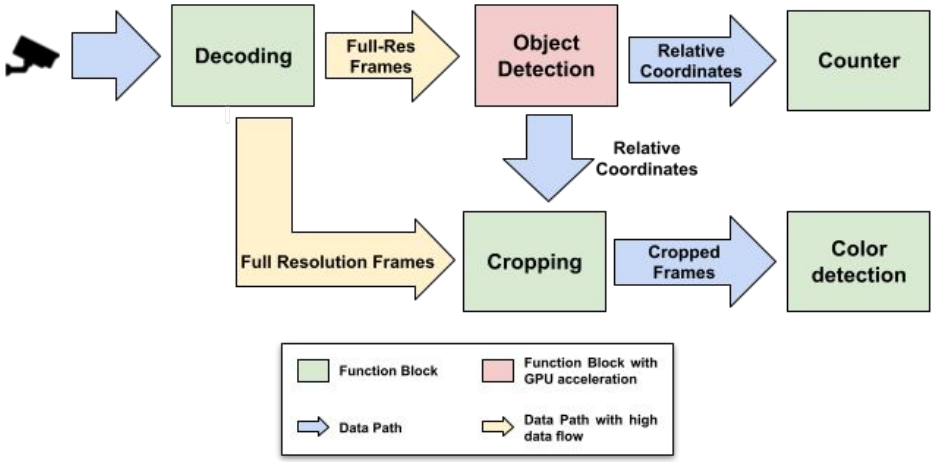
Improved latency and resource utilization

- Parallel utilization
- Sharing of common functions
- Conditional processing of functions



Improved latency and resource utilization

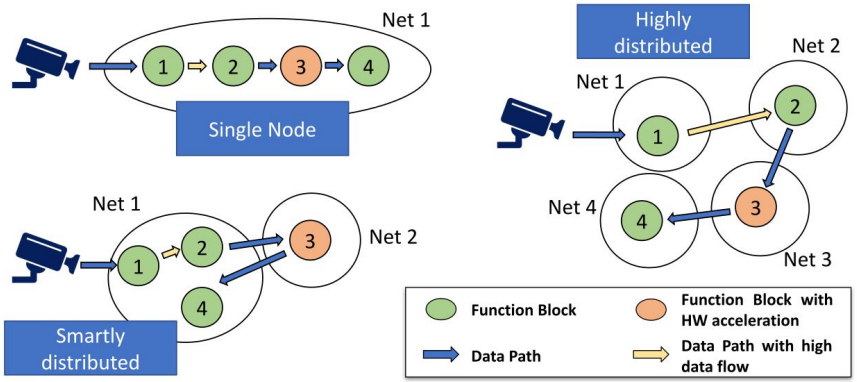
- Parallel utilization
- Sharing of common functions
- Conditional processing of functions



Adaptability

- Every task on one node

- Distributing all the tasks across different nodes in different networks



Our Solution -
Partly distributing the tasks among different networks

Experimental Setup

- NVIDIA Jetson Nano
  - Quad-core CPU
  - 128-core GPU
  - 4 GB shared RAM
- Functions
  - HTTP based microservices
  - Containers
    - CPU utilization and binding
  - Future - kubernetes like
- Traffic Control (TC)
  - Network Emulation (netem)
  - LAN - 1ms, 100 Mbps
  - WAN - 40 ms, 50 Mbps

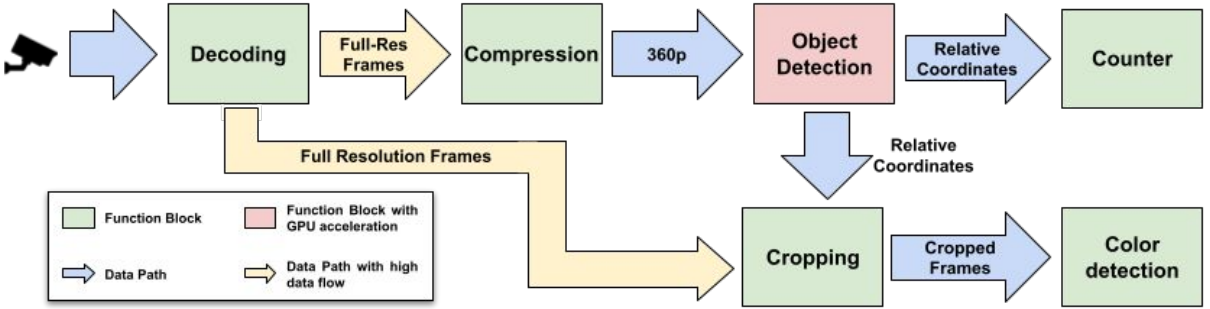
Experimental Setup

Applications Implemented

- Vehicle Counting
- Vehicle Color Recognition

Blocks implemented

- Decoding
- Compression/resize
- Object Detection
- Vehicle Counter
- Cropping
- Recognition



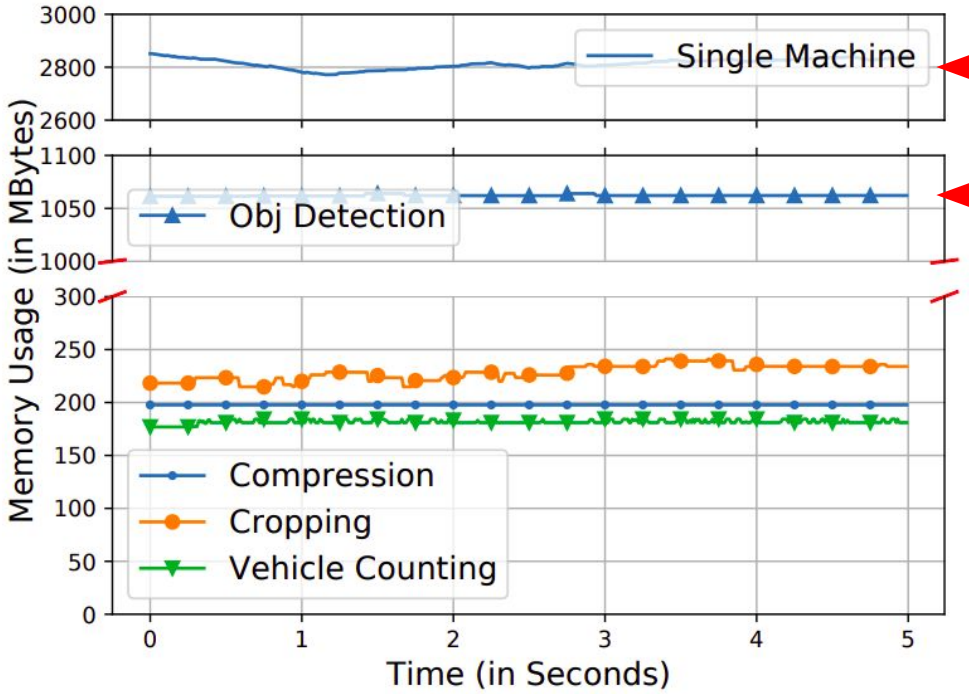
Evaluation

1. How is the resource utilization?
    - Memory, CPU, GPU utilization
2. Does distribution of blocks affect the performance?

Baseline -

Both application pipelines on a single machine

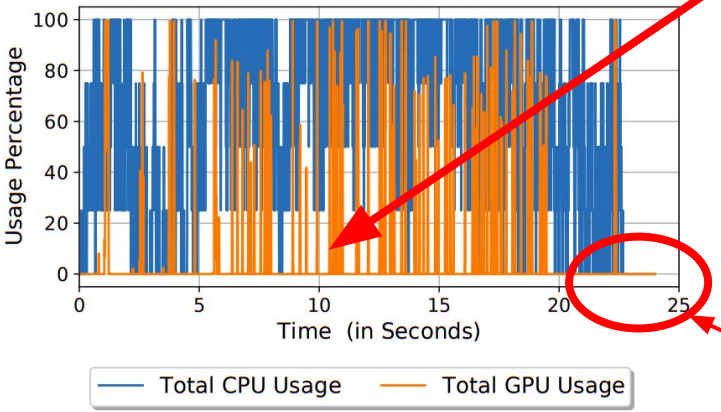
Evaluation - Memory Utilization



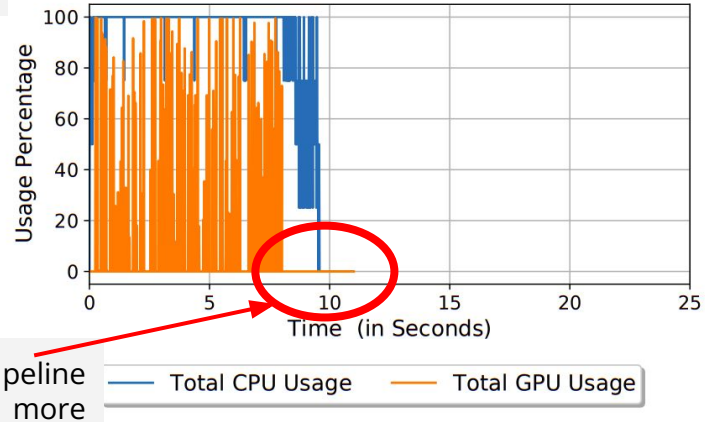
Baseline consumes more memory

Obj detection needs more memory

Evaluation - CPU and GPU Utilization



GPU Util is sparse

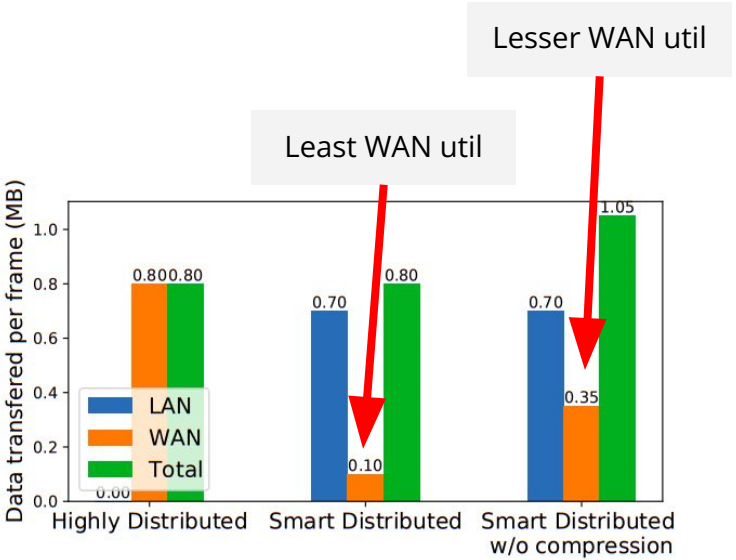


Distributed pipeline processes more frames per sec

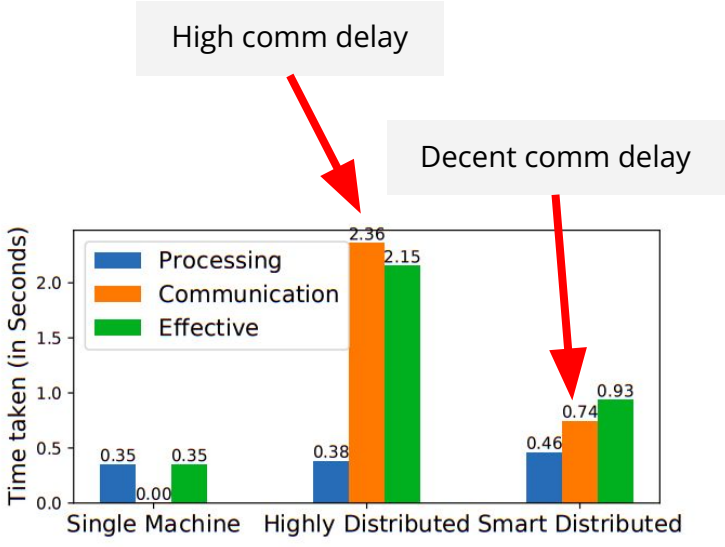
Change in processor utilization in single node pipeline over time

Change in processor utilization in distributed pipeline over time

Evaluation - Impact of distribution of blocks



Amount of data transferred over LAN and WAN in different distributions



Time taken per frame in different settings

Conclusion

- Modular decentralized architecture for video analytics at edge
- Functions splitting and distribution
- Feasibility study - more utilization and throughput

Future work-

- Easy programming construct - new blocks, pipelines
- Automated pipeline deployment
- Block deployment strategies

Thanks for your attention

Any Questions?

Summary-

- Modular decentralized architecture for video analytics at edge
- Functions splitting and distribution
- Feasibility study - more utilization and throughput

Future work-

- Automated pipeline deployment
- Block deployment strategies
- Easy programming construct