SERVING MACHINE LEARNING INFERENCE USING HETEROGENEOUS HARDWARE

Baolin Li*, Vijay Gadepally†, Siddharth Samsi†,
Mark Veillette†, Devesh Tiwari*

*Northeastern University, †MIT Lincoln Laboratory

Northeastern University

MIT LINCOLN LABORATORY

Wide Range of Applications using Machine Learning

Vision

Recommendation

amazon

NETFLIX

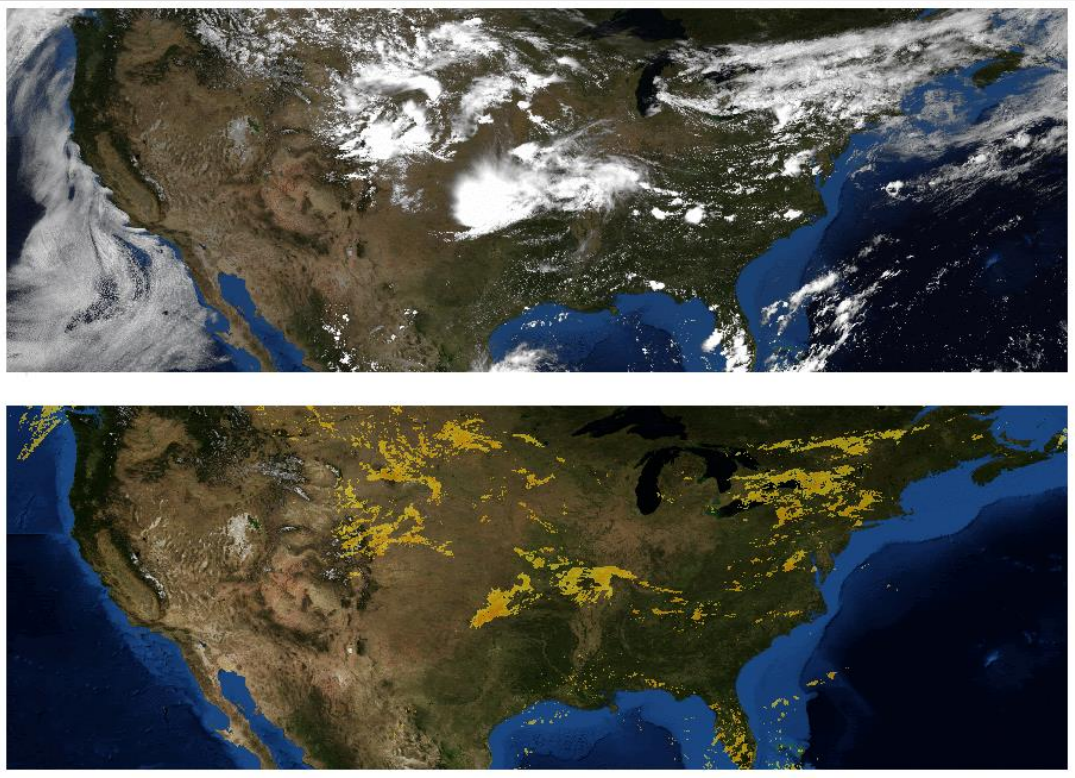
Language

Google Translate

Siri

https://www.dynam.ai/what-is-computer-vision-technology/

Weather Nowcasting

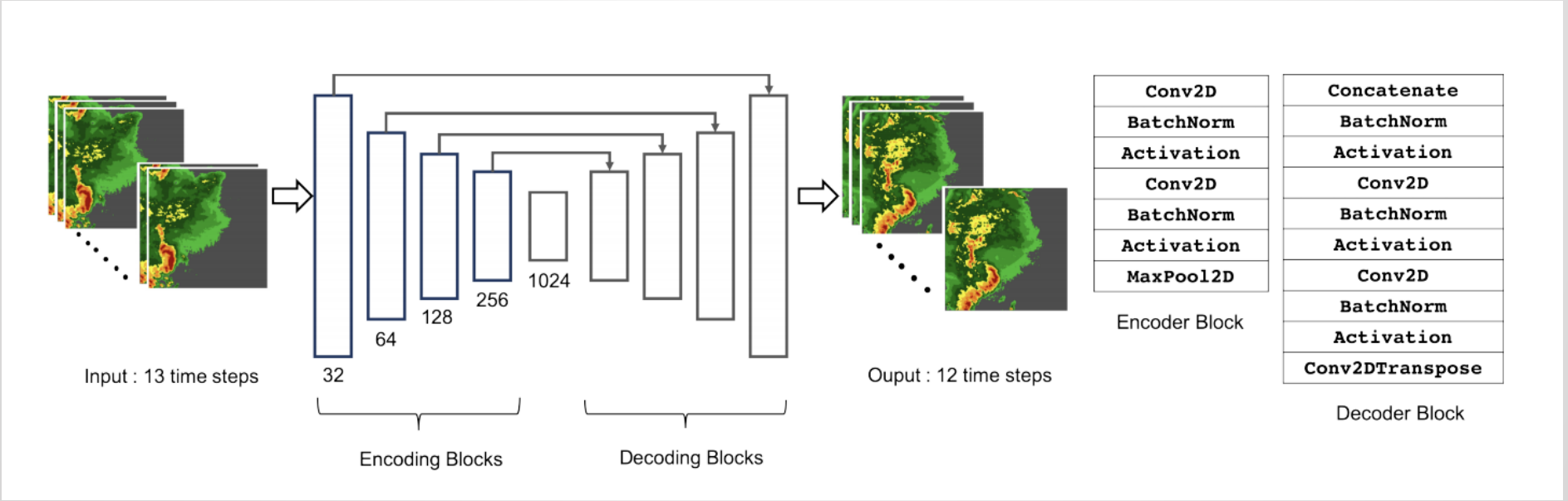


https://ai.googleblog.com/2020/01/using-machine-learning-to-nowcast.html

NWP	ML
Numerical	Statistical Inference
Hours of simulation	Runs in Seconds

Weather Nowcasting

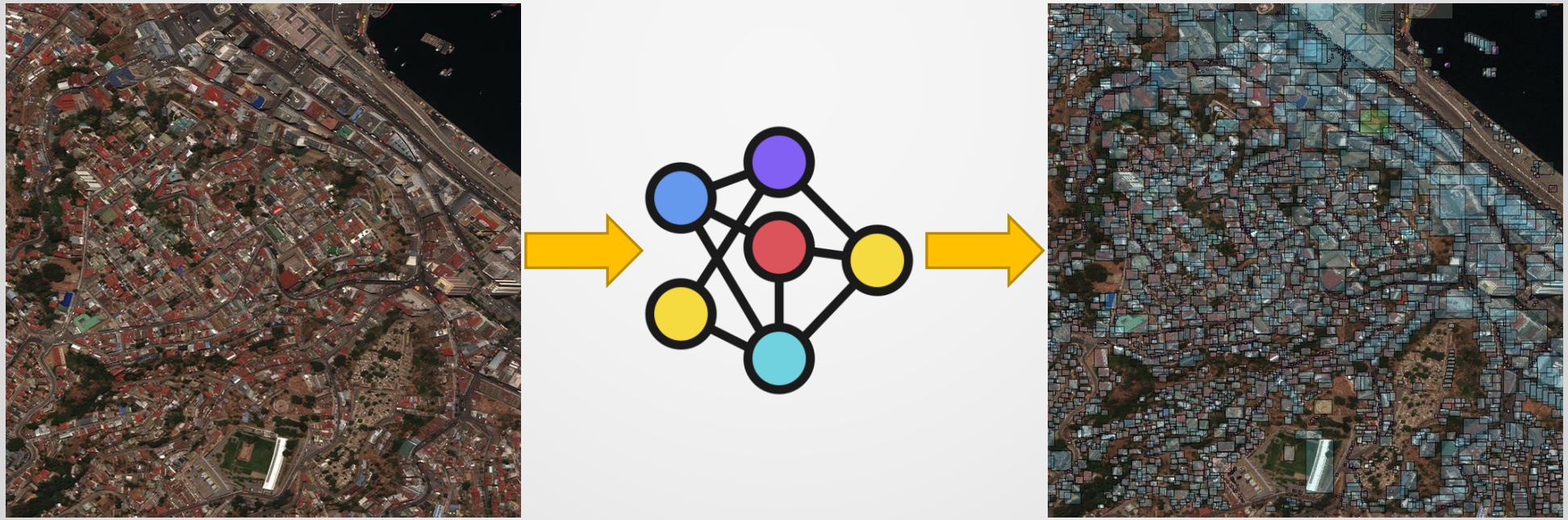
- U-Net weather nowcasting on SEVIR (Storm Event Imagery Dataset).
- Inference takes < 200ms on an NVIDIA T4 GPU.



https://github.com/MIT-AI-Accelerator/sevir_challenges

Satellite Imagery Object Detection

- xView dataset: http://xviewdataset.org/. Covers 1400 km^2 of earth surface.
- YOLOv3 model for real-time detection with low end-to-end latency.



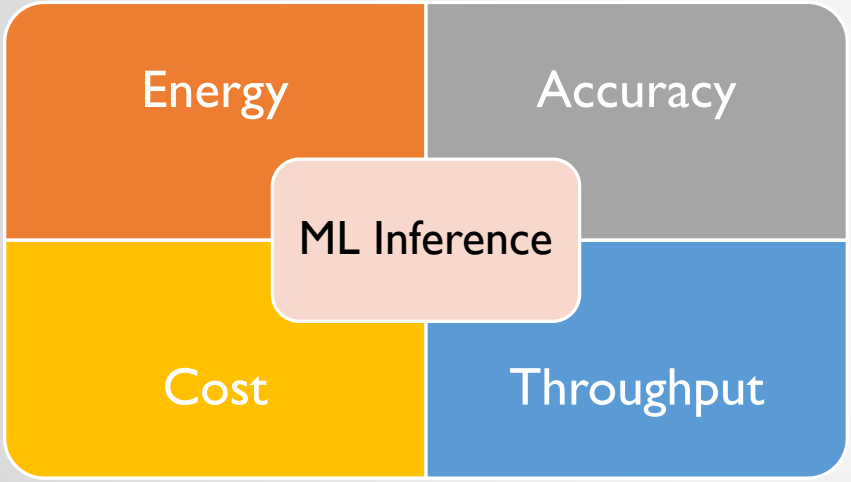
Heterogeneity in HPC Systems

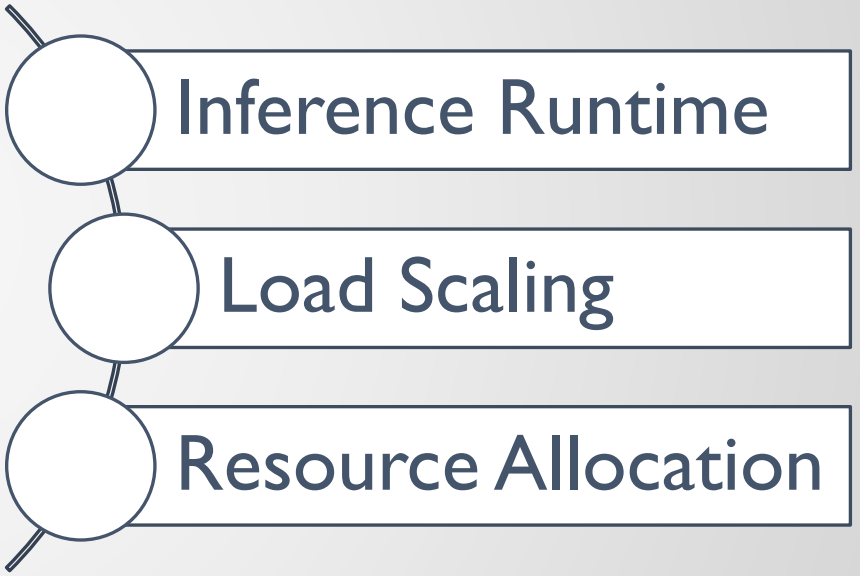
- HPC systems tend to be heterogeneous.

TACC – TEXAS ADVANCED COMPUTING CENTER	MGHPCC – MASSACHUSETTS GREEN HIGH PERFORMANCE COMPUTING CENTER	SDSC – SAN DIEGO SUPERCOMPUTER CENTER
NVIDIA GTX 1080	NVIDIA K80	NVIDIA K80
NVIDIA Tesla P100	NVIDIA P100	NVIDIA P100
NVIDIA Tesla V100	NVIDIA V100	
	NVIDIA T4	

Previous Work in Inference Serving

- Previous work have explored various areas of ML inference serving.

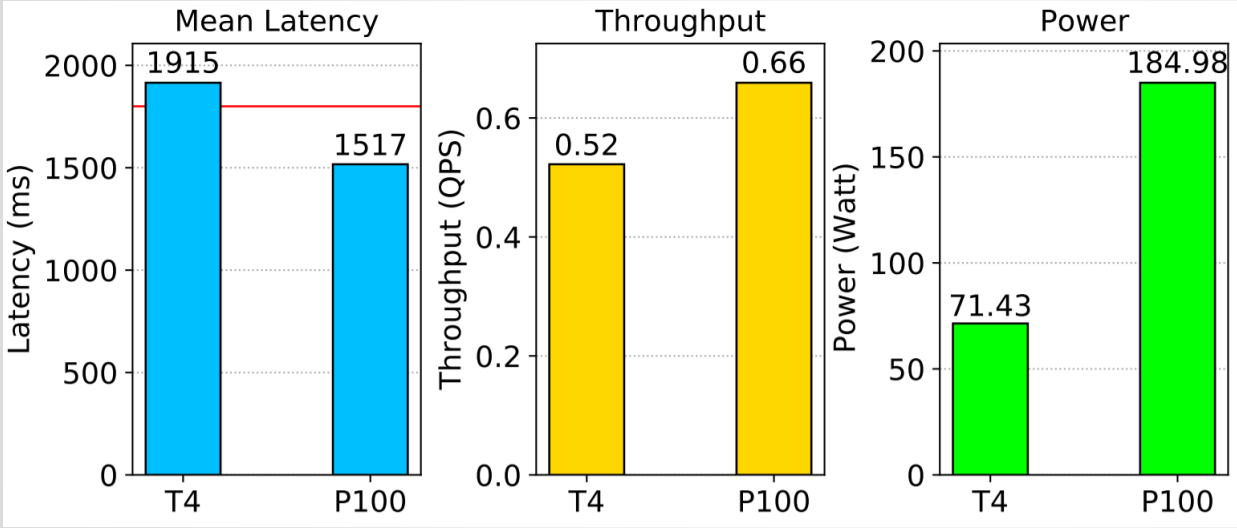




- What is missing: an inference solution that exploits heterogeneity in HPC systems.

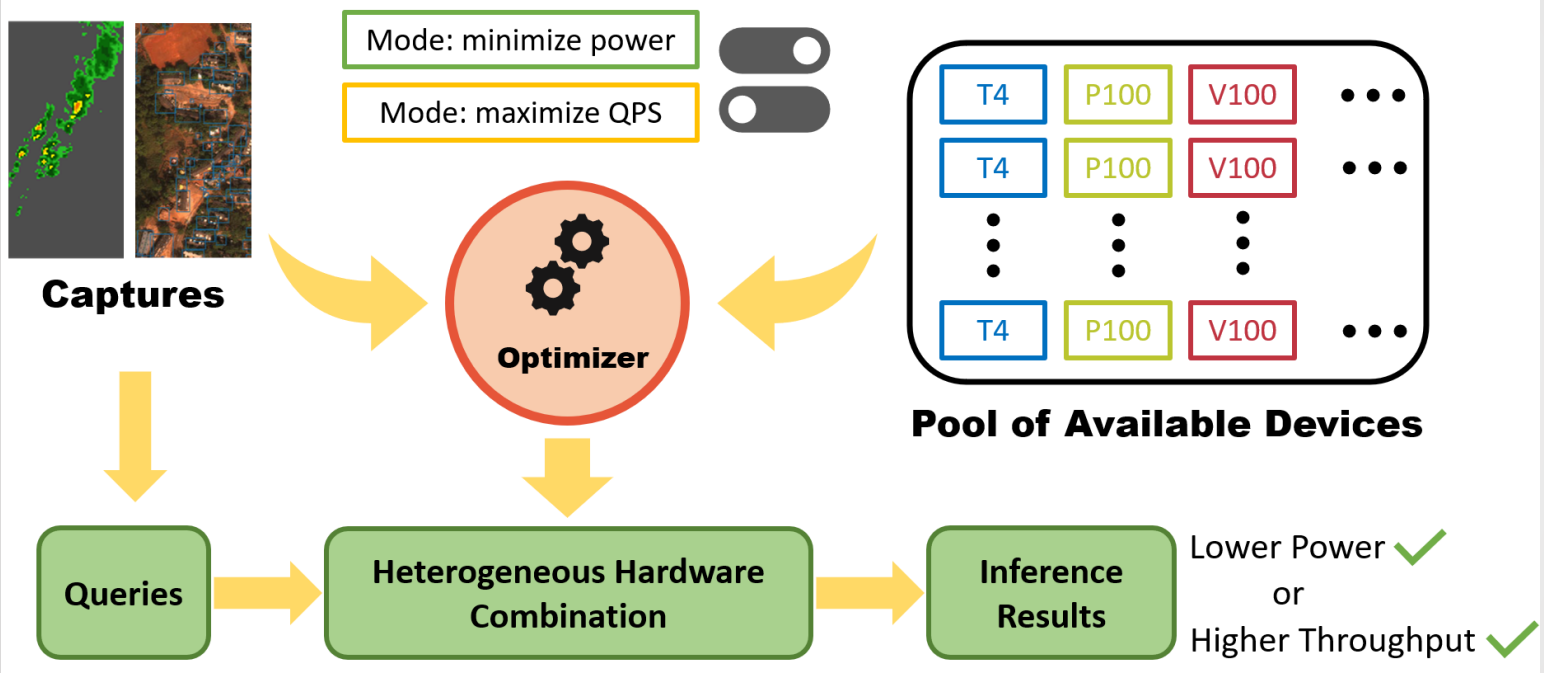
Latency, Throughput and Power Trade-offs

- xView object detection inference using T4 and P100 GPUs.



- Can we combine different GPU types to serve the queries such that:
  - Latency is within a target
  - Throughput or power are optimized

Inference Serving System using Heterogeneous Hardware



- Which hardware type to choose?
- How many devices of each type to use?

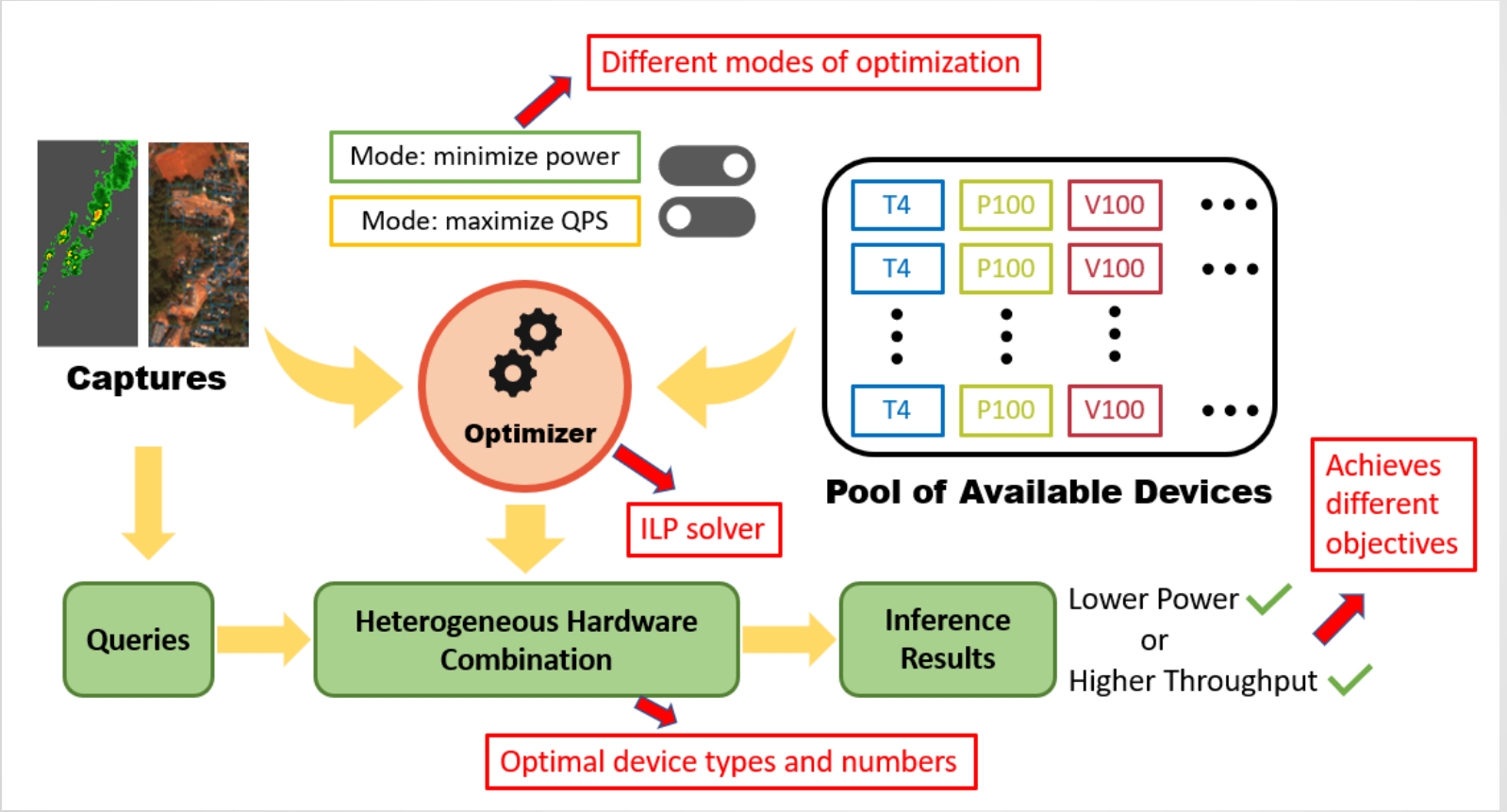
Optimization Goals and Constraints

- Two optimization modes

Power	Throughput
Latency limit	Latency limit
Throughput target	Power limit
Minimize power	Maximize throughput

- Inputs: inference latency, throughput and power of each hardware type
- Variables: integer number of devices for each type
- All optimization constraints and objectives are linear functions to the variable
  - Integer linear programming (ILP) problem

Inference Serving System



Northeastern University

MIT LINCOLN LABORATORY

Evaluation - Power Saving

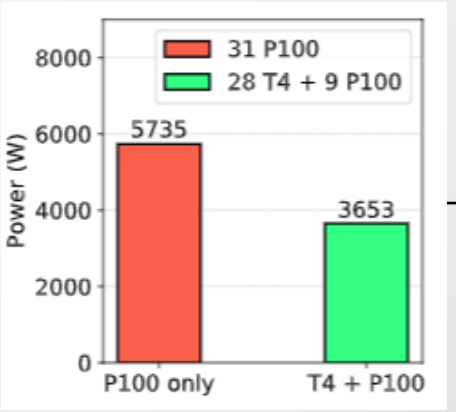
SEVIR inference with U-Net			
Types	NVIDIA T4	NVIDIA P100	Target
Mean latency (ms)	150	137	145
Throughput (QPS)	6.7	7.3	200
Power (W)	92	133	Minimize

Find optimal configuration →

Results in → 16% power saving

xView inference with YOLOv3			
Types	NVIDIA T4	NVIDIA P100	Target
Mean latency (ms)	1915	1517	1800
Throughput (QPS)	0.52	0.66	20
Power (W)	71	185	Minimize

Find optimal configuration →



Results in → 36% power saving

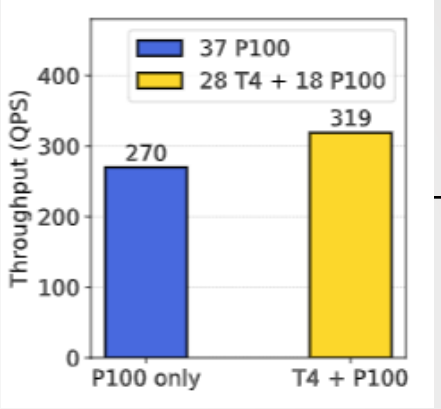
Northeastern University

MIT LINCOLN LABORATORY

Evaluation - Throughput Improvement

SEVIR inference with U-Net			
Types	NVIDIA T4	NVIDIA P100	Target
Mean latency (ms)	150	137	145
Throughput (QPS)	6.7	7.3	Maximize
Power (W)	92	133	5000

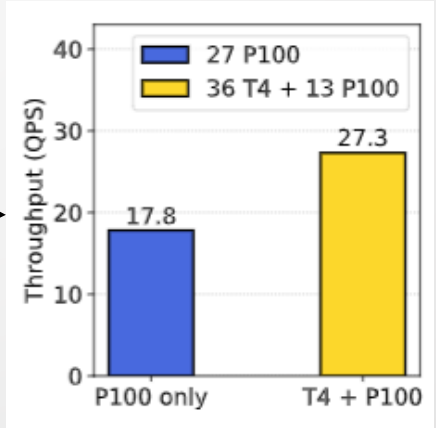
Find optimal configuration →



Results in → 18% throughput gain

xView inference with YOLOv3			
Types	NVIDIA T4	NVIDIA P100	Target
Mean latency (ms)	1915	1517	1800
Throughput (QPS)	0.52	0.66	Maximize
Power (W)	71	185	5000

Find optimal configuration →



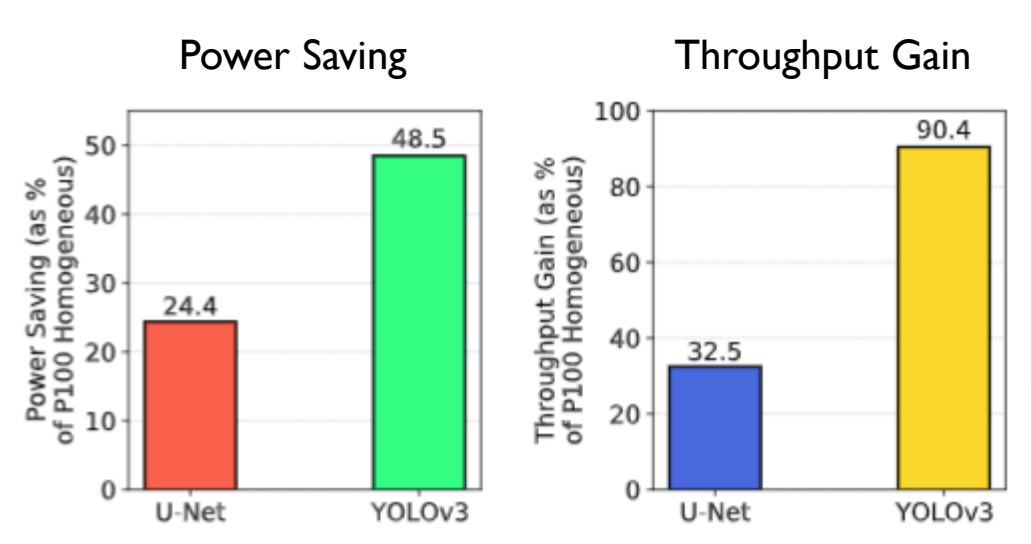
Results in → 53% throughput gain

Northeastern University

MIT LINCOLN LABORATORY

Evaluation – More Device Types

- Suppose a wide variety of device types are available
  - Intel Xeon Silver 4114 CPU
  - NVIDIA K80
  - NVIDIA M60
  - NVIDIA P100
  - NVIDIA V100
  - NVIDIA T4

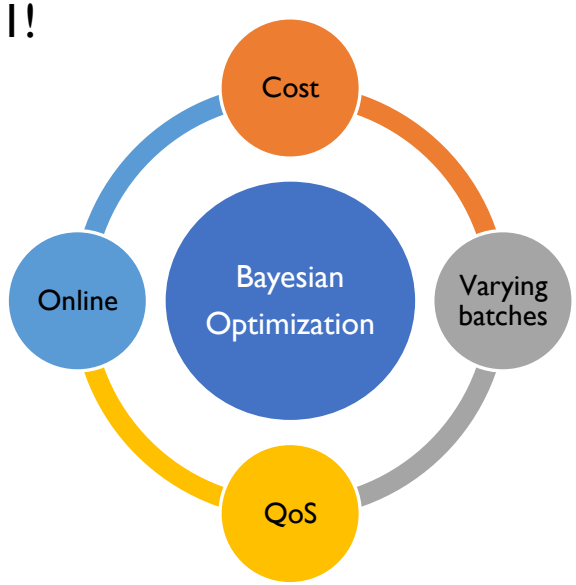


- The optimizer finds the optimal device types (V100 and T4) and configures hardware combination

Takeaways and Limitations

- Main takeaways
  - HPC systems tend to be heterogeneous
  - Our framework exploits this heterogeneity for power and throughput optimizations
- Limitations of this work
  - We assumed queries have fixed batch size
  - Requires prior profiling of the model served by each hardware type
  - Tail latency as quality-of-service (QoS) cannot be analytically derived

Check out our upcoming presentation "Ribbon" (Request Inference Based on Bayesian Optimization) at Supercomputing in Nov. 2021!



Research was sponsored by the United States Air Force Research Laboratory and the United States Air Force Artificial Intelligence Accelerator and was accomplished under Cooperative Agreement Number FA8750-19-2-1000. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the United States Air Force or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

Northeastern University

MIT LINCOLN LABORATORY

Questions

For further questions please email me at li.baol@northeastern.edu