MVAPICH
MPI, PGAS and Hybrid MPI+PGAS Library

ICICLE
DEMOCRATIZING AI

HARVEST Inference: Characterizing Digital Agriculture Workloads across Compute Continuum

Tian Chen Quentin Anthony Dhabaleswar K. Panda

{ chen.9891, anthony.301, panda.2 }@osu.edu

The Ohio State University

Outline

- Introduction
- Background
- Challenges
- Experiments
- Conclusion

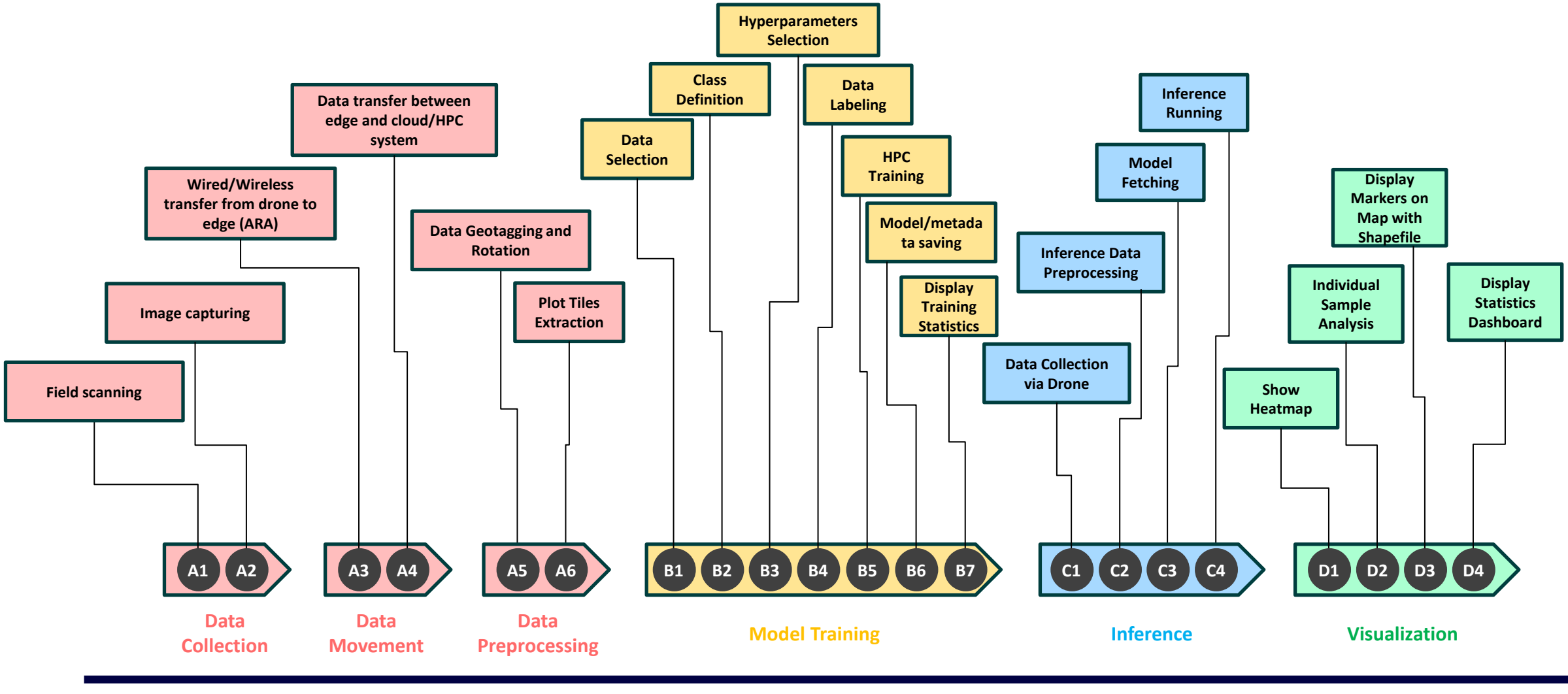
Introduction

- Challenges in Agriculture
- HARVEST framework (Recap)

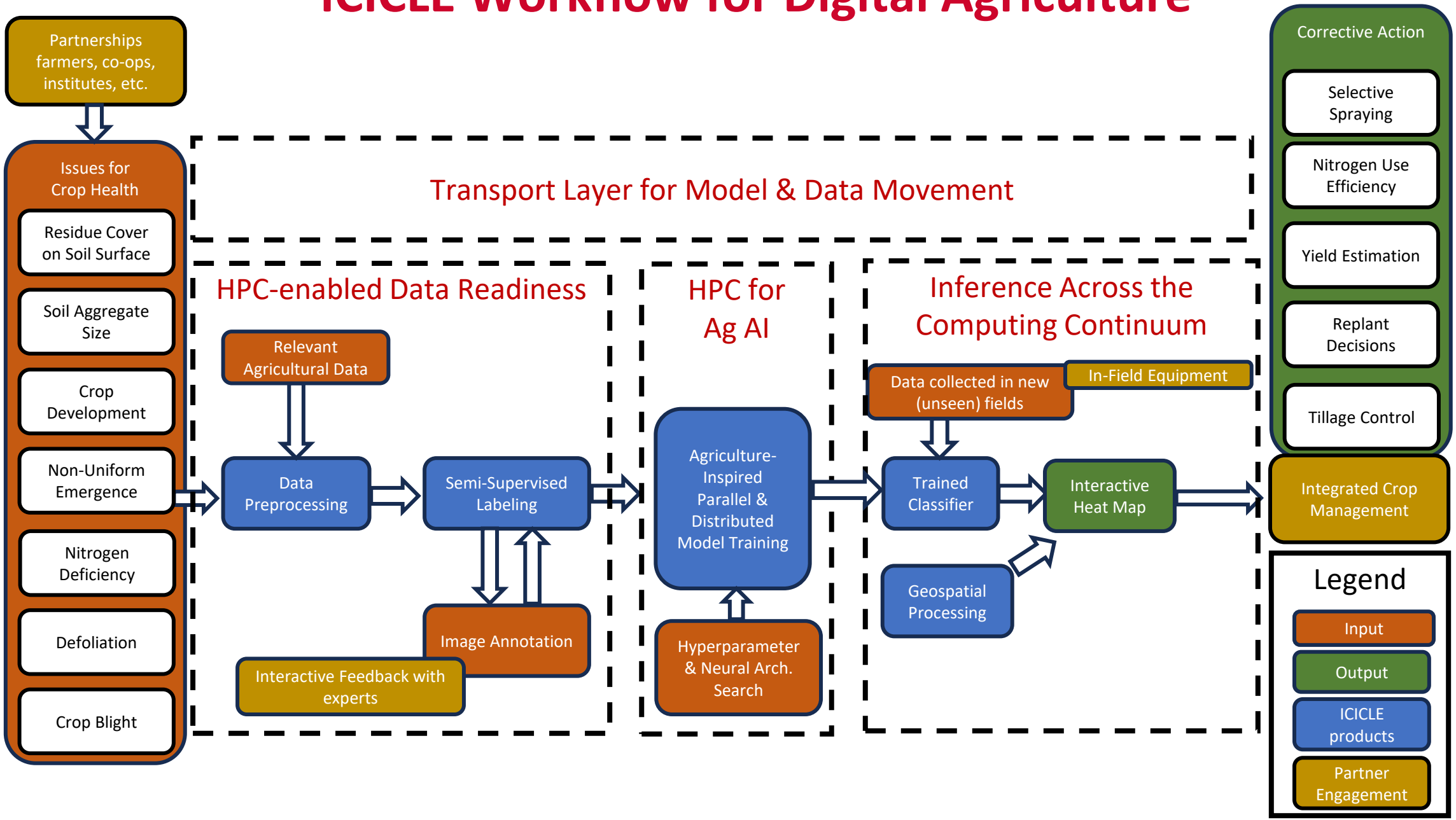
Challenges in Agriculture

- **Food security/sustainability in 2050**
  - 9.8B people, climate; 0.5x arable land per cap vs 1985
  - Wide gains in crop management needed (typical yields fall 3X below best practice)
- **Sustainable agricultural workforce**
  - The next generation of agriculture professionals will include engineers, computer scientists, data scientists
- **Democratization of digital agriculture capabilities**
  - Autonomous unmanned aerial vehicles, self-driving tractors and sprayers, fertilizer and seed recommendations
  - Big and small farms, staple and specialty crops, underrepresented communities
  - Privacy and ethical considerations

The ICICLE Digital Agriculture Pipeline

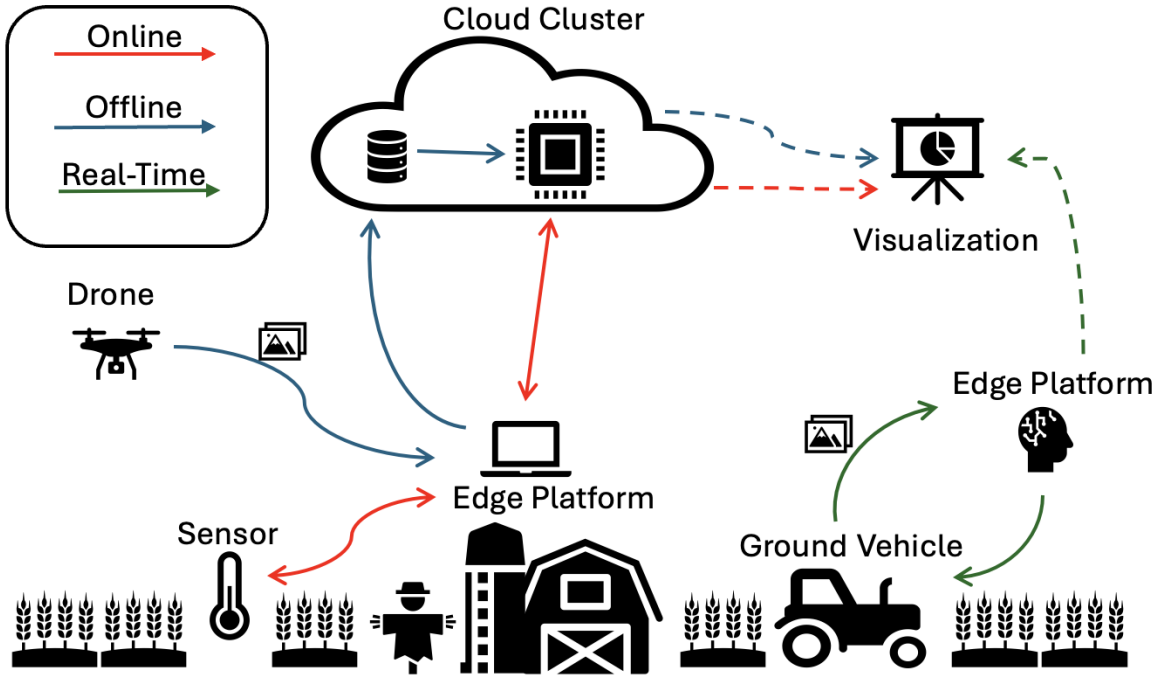


ICICLE Workflow for Digital Agriculture

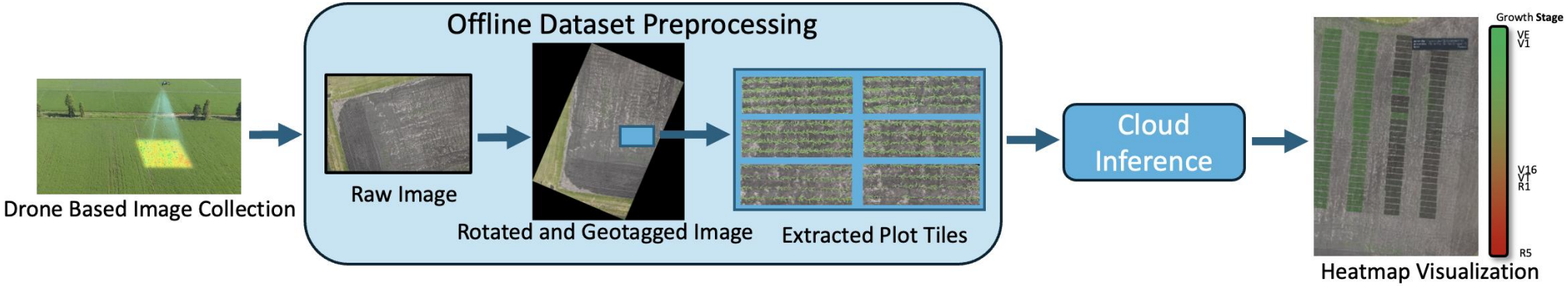


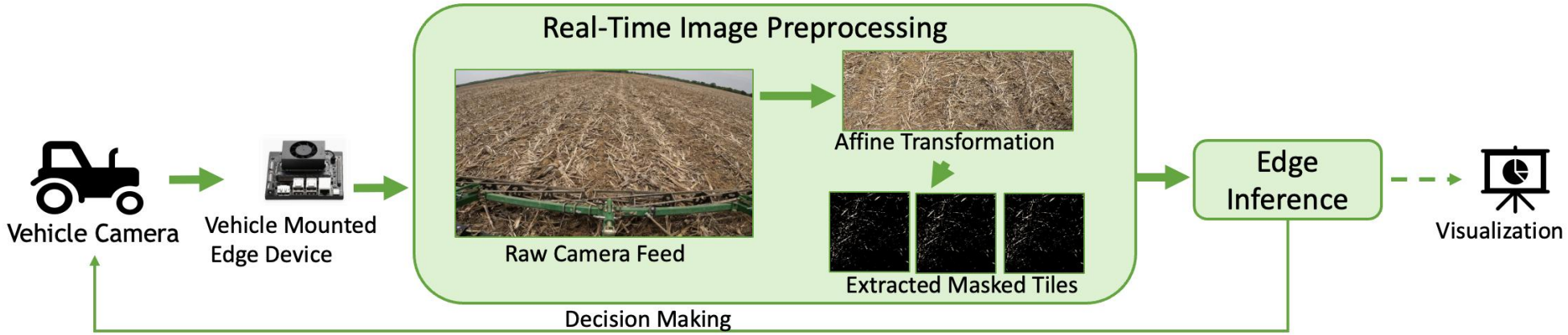
Deployment Scenarios

Scenario	Inference Platform	Transmission Required?	Input	Output	Latency Requirement	Throughput
Offline	Cloud	✅	Entire dataset	Visualization	Low	High
Online	Cloud	✅	A batch of images	Live Visualization	Medium	Medium
Real-Time	Edge	❌	Single image	Control Decision	High	Low



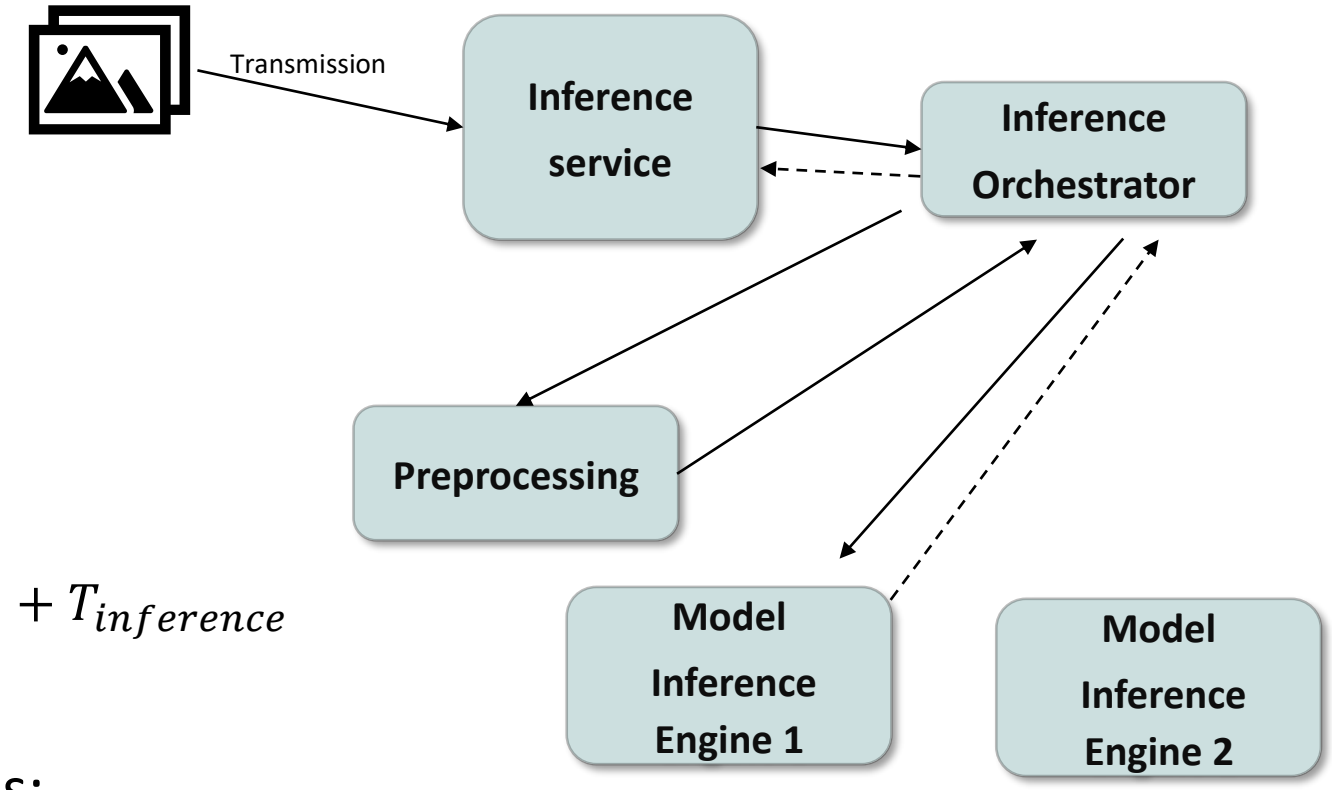
Deployment Scenarios - Example





Inference pipeline

- Steps involved:
  - Inference server send request
  - Inference Orchestrator
  - Application Specific Preprocessing
  - Model inference engine
- The latency consist of:
  - $T_{latency} = T_{scheduling} + T_{preprocessing} + T_{inference}$



- The throughput can be presented as:
- $Throughput = (1 + \text{overlapping}) * BatchSize * 1/Max(T_{scheduling} + T_{preprocessing} + T_{inference})$

Challenges

- The large number of possible application/model/hardware/scenario combination
  - The mystery mutual effect of different components.
  - Each case has different emphasize/criteria.
  - Different dynamic for different platforms.
    - Inference on Cloud vs Edge
    - Port same model to different platform
  - Hard to predict model performance in advance.

- Goal:
  - Application specific optimization
  - Predict performance in advance of model training/deployment.

Experiment Setup - Hardware

Table 1: Evaluated Cloud and Edge Platforms

Platform	OSC Pitzer Cluster (V100)	MRI Cluster (A100)	NVIDIA Jetson Orin Nano Super
CPU	40 cores	128 cores	6 cores
GPU	NVIDIA V100 16GB×2	NVIDIA A100 40GB×2	Ampere architecture based 1024 CUDA cores 32 tensor cores
Memory	384GB	256GB	8GB
Scenario	Online, Offline	Online, Offline	Real-Time
Theory TFLOPS	112 @FP16	312 @BF16	17 @FP16
Practical TFLOPS	92.6	236.3	11.4 @BF16

Note: V100 and A100 experiments used only one of the two available GPUs. Jetson platforms feature CPU and GPU sharing 8GB unified memory and operate in 25W power mode.

Experiment Setup – Application/Dataset

Table 2: Agriculture Datasets Used in The Evaluation

Dataset	Classes	Samples	Image Size	Use Case
Plant Village [8]	39	43430	256x256	Plant disease classification
Weed Detection in Soybean [10]	4	10635	Show in Fig. 4a	Weed detection in soybeans
Sugar Cane-Spittle Bug [27]	2	10100	Show in Fig. 4b	Pest bugs detection
Fruits-360 [18]	81	40998	100x100	Fruits classification
Corn Growth Stage [28]	23	52198	224x224	Corn Growth Stage Classification, UAS Based
CRSA	-	992	3840x2160	Crop Residue Soil Aggregate, Ground Vehicle based

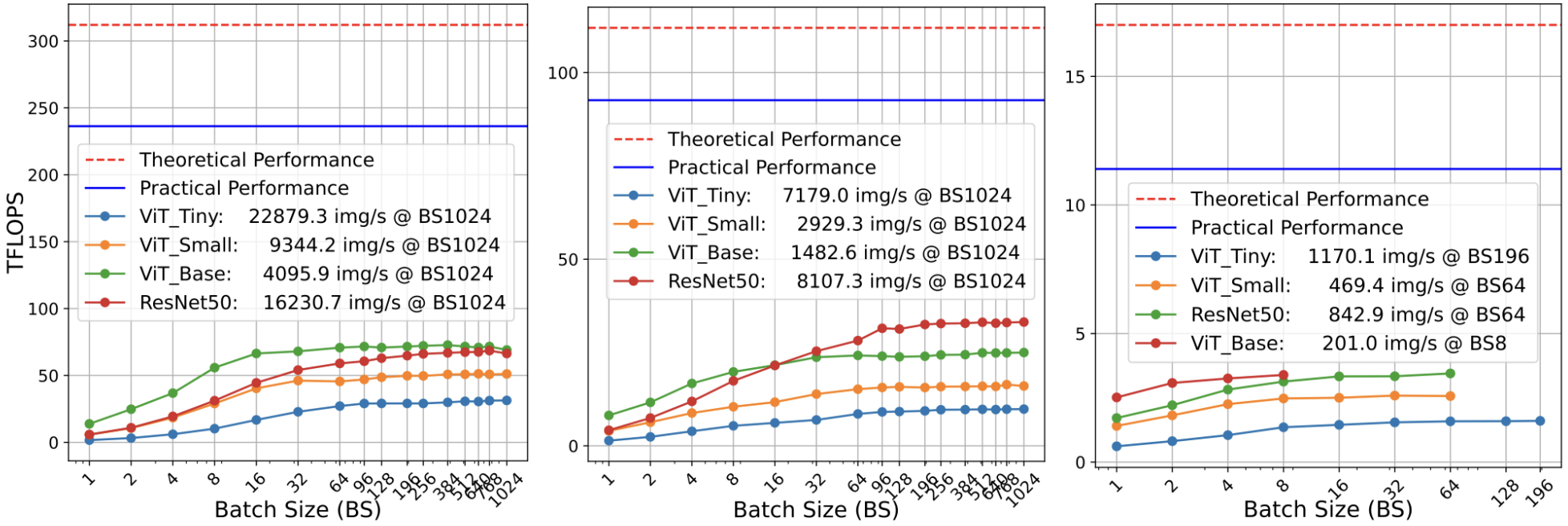
Experiment Setup – Model

Table 3: Model Evaluated and Computational Intensity

Model		ViT Tiny	ViT Small	ViT Base	ResNet50
Parameter		5.39M	21.40M	85.80M	25.56M
Archetecture		Transformer Based			CNN Based
GFLOPs/Image		1.37	5.47	16.86	4.09
Input Size		32×32	32×32	224×224	224×224
Throughput	A100	172,508	43,214	14,013	57,775
UpperBound	V100	67,602	16,935	5,491	22,641
images/sec	Jetson	8,322	2,085	676	2,787

Experiments – Compute Intensity vs Batch Size

- When we consider model inference solely:
  - Large BatchSize gives better performance. But will saturate gradually.
  - Gap between theoretic performance and practice number (Model Flops Utilization (MFU))



(a) A100 (b) V100 (c) Jetson

Figure 5: Scaling Behavior Of Compute Intensity With Varying Batch Sizes On Different Hardware Platforms. The Dashed Lines Represent The Theoretical FLOPS, While The Solid Lines Indicate The Actual FLOPS Achieved By Each Model.

Experiments – Request Latency vs Batch Size

- When we consider request latency and latency constrains:
  - BatchSize trade off between throughput and latency.
  - Gap between theoretic performance and practice number (Model Flops Utilization (MFU))

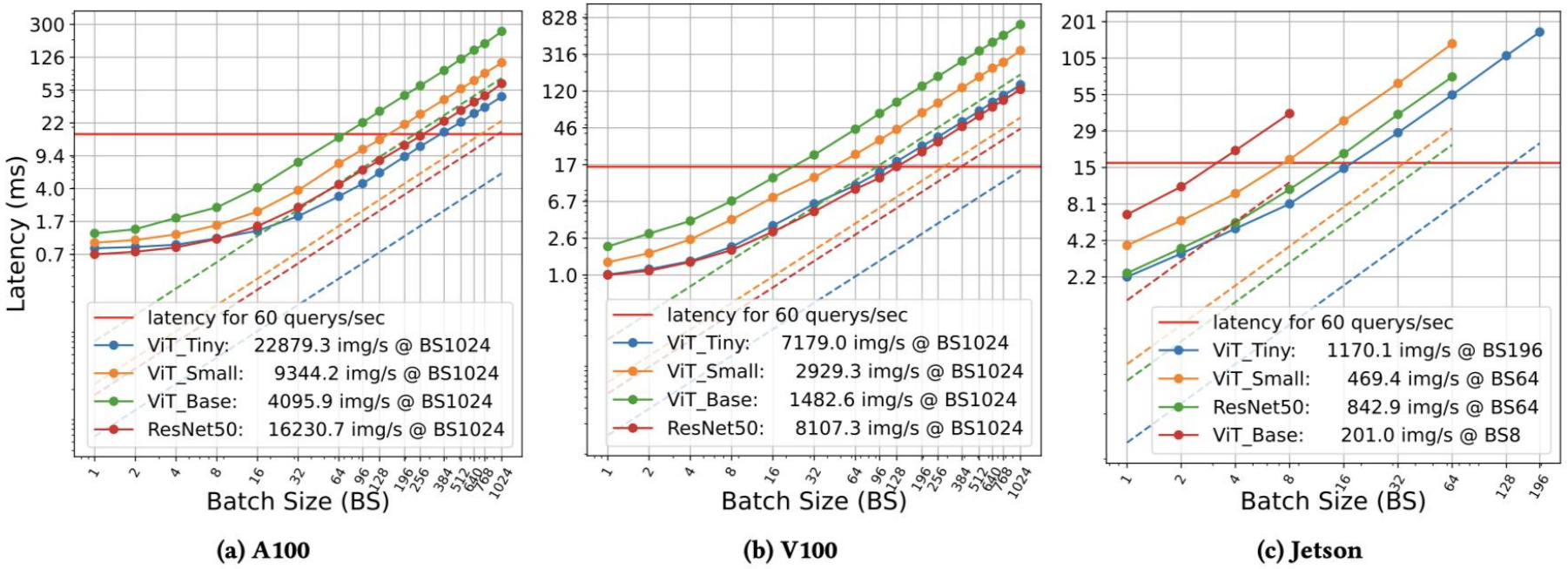


Figure 6: Request Latency Vs. Batch Size Across Hardware Platforms For Multiple Models. The Dashed Lines Represent The Theoretical Latency, While The Solid Lines Indicate The Actual Latency Achieved By Each Model.

Experiments – Preprocessing

- When we consider preprocessing performance:
  - GPU provide huge speedup (pitfall: will it cause contention with model inference?)
  - Not all application support GPU preprocessing.
  - Image size matter in some cases.

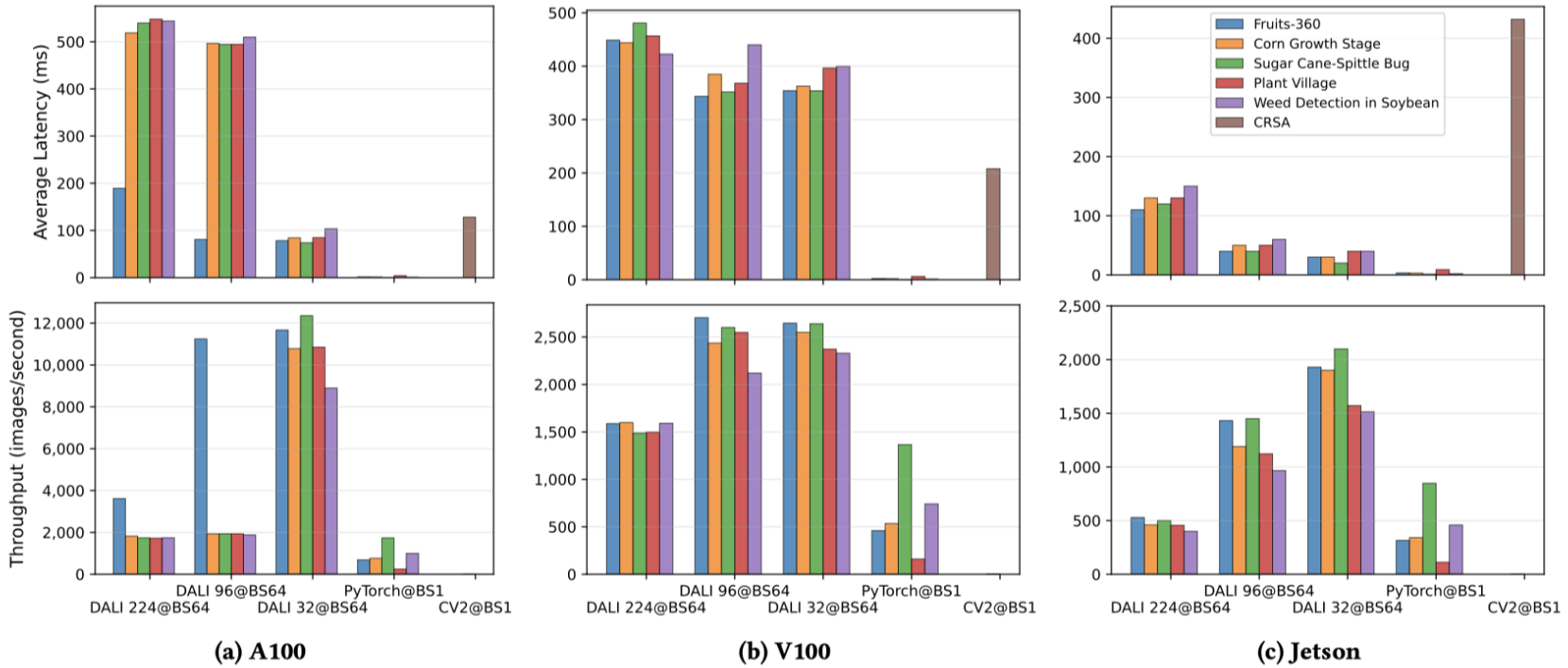


Figure 7: Preprocessing Throughput And Latency For Different Datasets Across Platforms. Upper figure show request latency, lower show throughput across different preprocessing Methods. Batch Size varies by method.

Experiments – End to End evaluation

- Put things together:
  - Some preprocessing bound, some inference bound. (cloud/edge)
  - Best batch size in previous section may out of memory.
  - -> it's a more complex mutual effect.

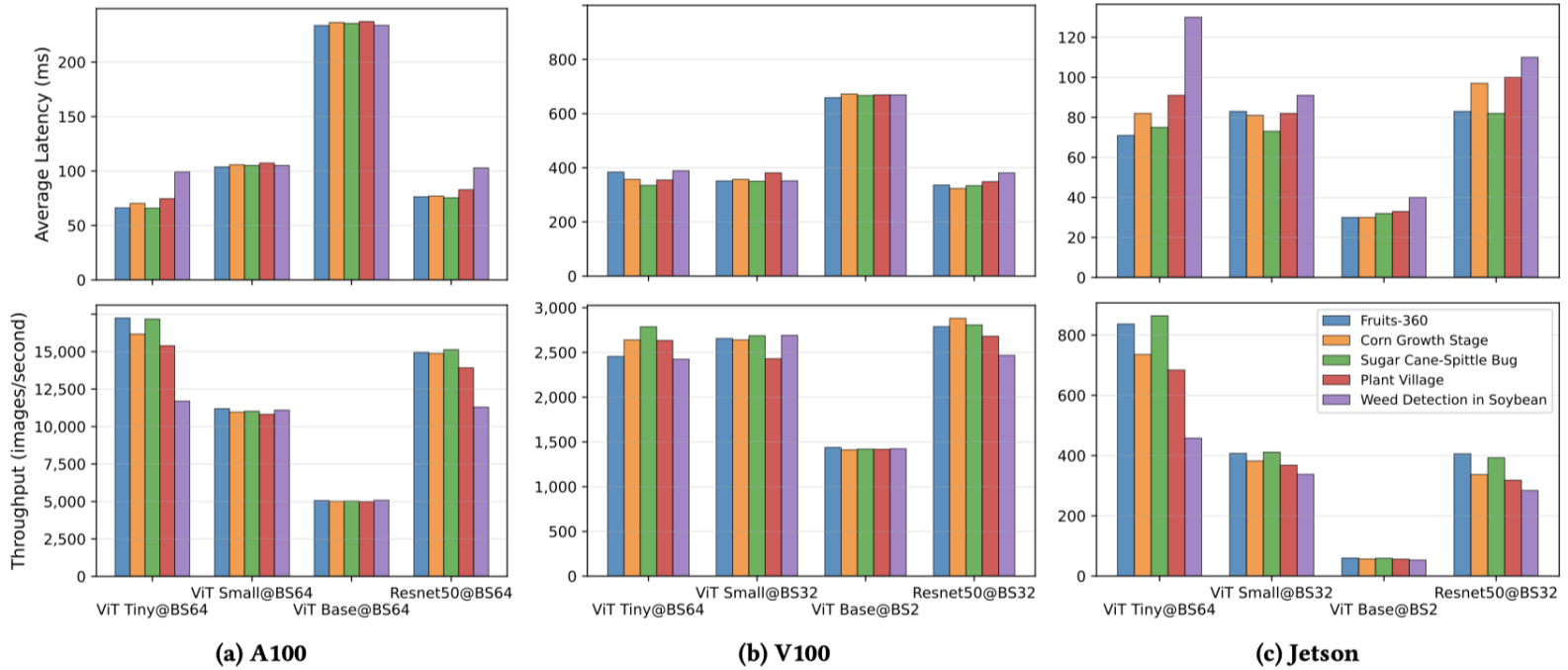


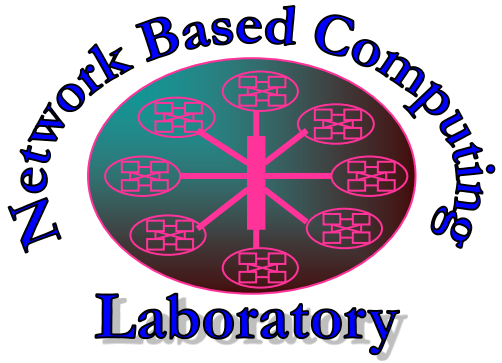
Figure 8: End-To-End Pipeline Inference Latency And Throughput For Different Datasets Across Platforms. Upper figure show request latency, lower show throughput across different model. The largest Batch Size before Out-of-memory (OOM) was used.

Conclusion

- Harvest inference pipeline supports wide range of digital agricultures applications with different scenarios. Boosting the democratized AI in DA.
- The large combination of application, scenarios, hardware, model combination, and complex mutual effect caused challenge to optimizing inference.
- This work disclosure the characteristic of different inference component in a qualitative way.
- Future work:
  - Give a formulized guidance for end user predict model performance in advance to model training/deployment. A toolkit for it.

Thank You!

shineman.5@osu.edu, michalowicz.2@osu.edu



Network-Based Computing Laboratory
http://nowlab.cse.ohio-state.edu/

The High-Performance MPI/PGAS Project
http://mvapich.cse.ohio-state.edu/

The High-Performance Deep Learning Project
http://hidl.cse.ohio-state.edu/