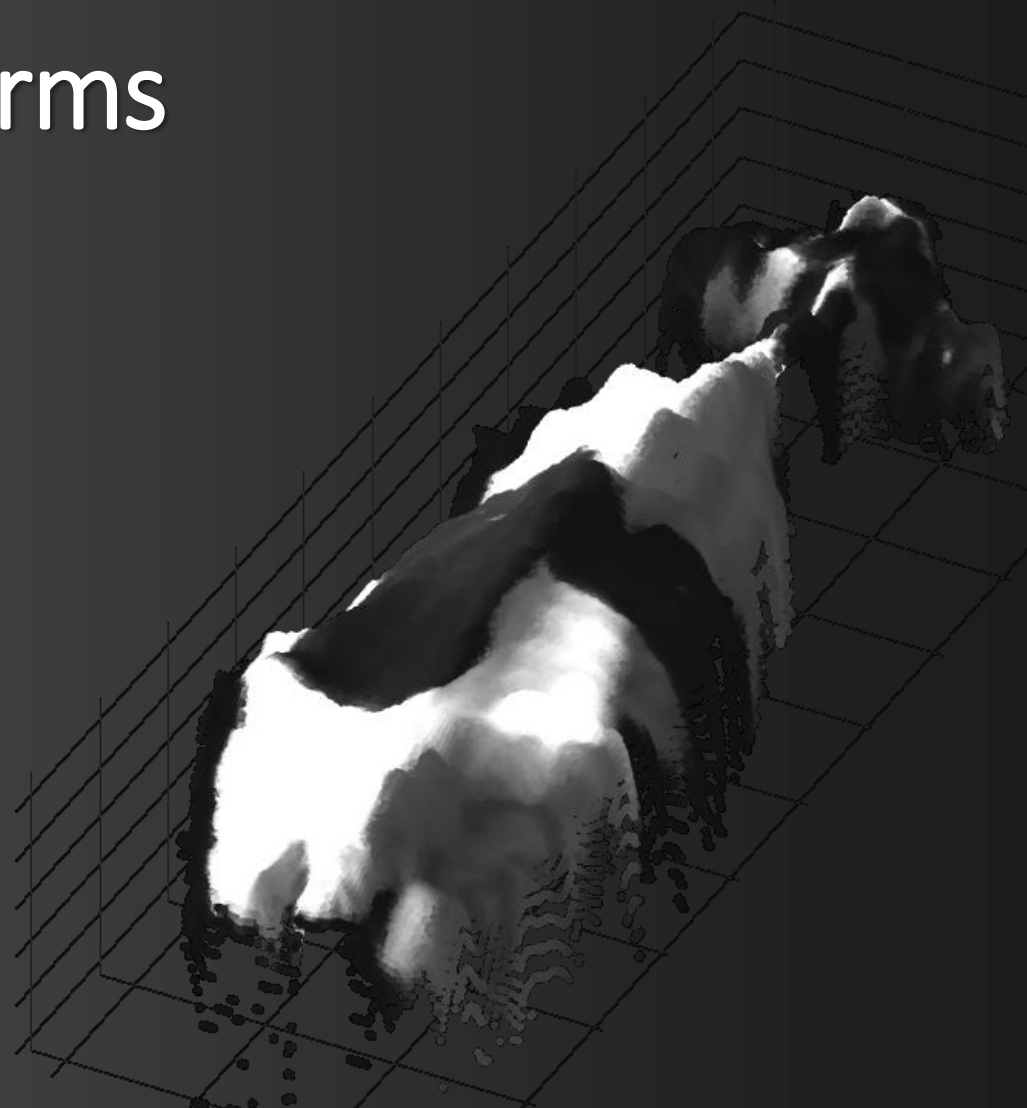
Using computer vision for individual animal monitoring in dairy farms

Rafael E. P. Ferreira

PhD Candidate

Department of Animal and Dairy Sciences

WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON



Outline

1. My research
2. Why CHTC?
3. My experience with CHTC
   a. Learning resources
   b. Use cases
   c. Pain points / nice-to-haves
4. Personal and professional benefits

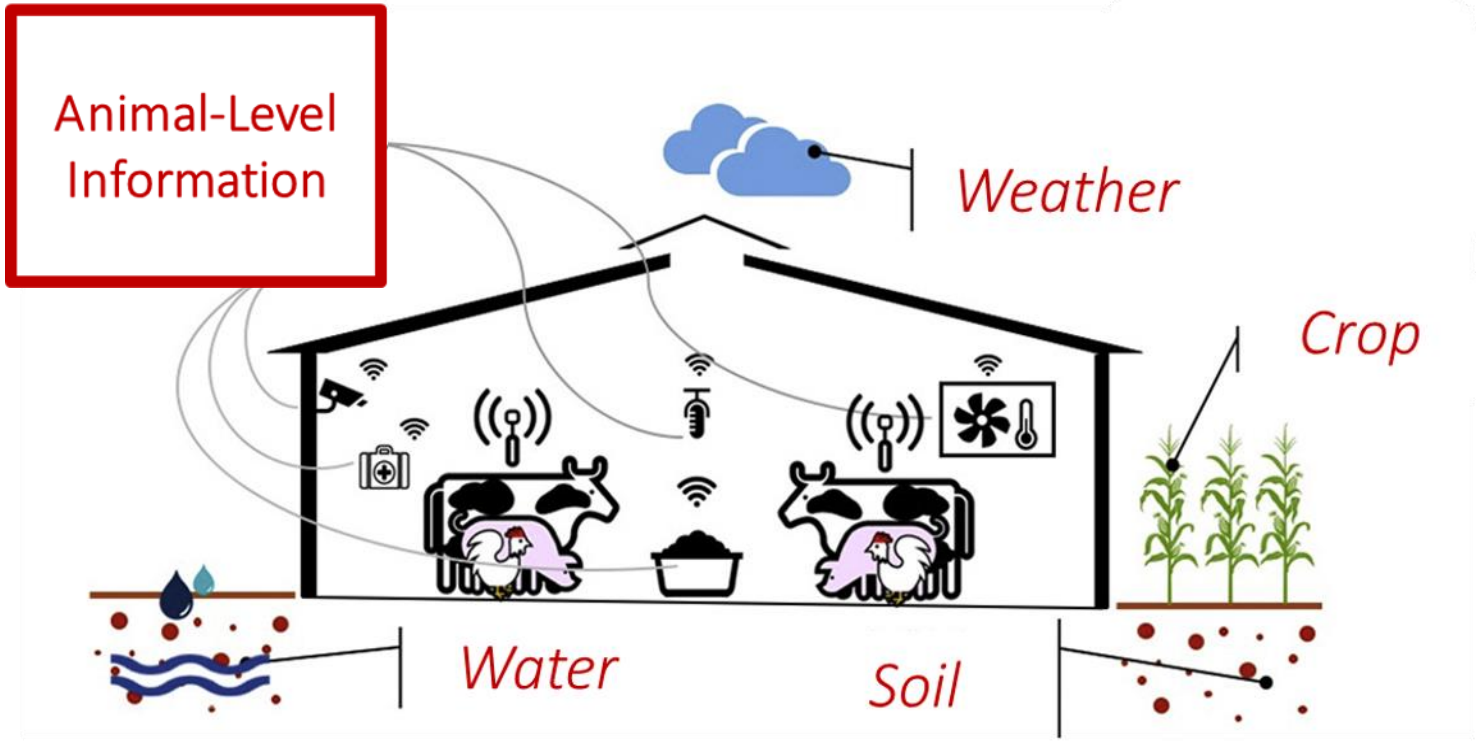
1. My research

Dr. Dorea's lab

- Research applications of machine learning and computer vision for precision dairy farming
- Motivation: Farm management and genetic selection

Sensors:
Wearable
Cameras
IR Spec.
RFID
Sound
Housing

Animal Identification
Animal Behavior
Body Weight
BCS/Composition
Milk Components
Milk Yield
Estrus Event
Feed Intake
Feed Efficiency
Disease Risks



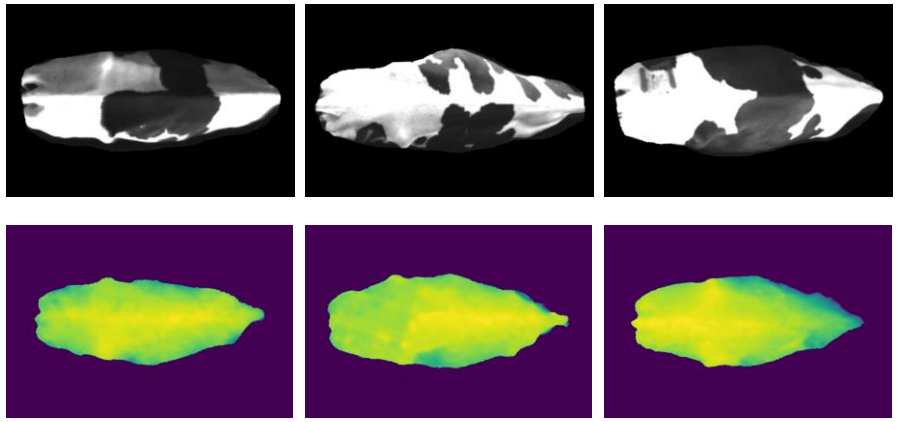
Tullo et al., 2019

Why Computer Vision?

- Cameras are relatively inexpensive and easy to install
- Monitor multiple animals at a time
- Images can be very informative:
  - Weather/season
  - Quality of the pasture
  - Location of the animals
  - Social interaction
  - Many more…

Individual animal monitoring

- First step: Animal identification
  - Coat color pattern (dairy cows)
  - 3D surface (other species or animals that look similar to each other)



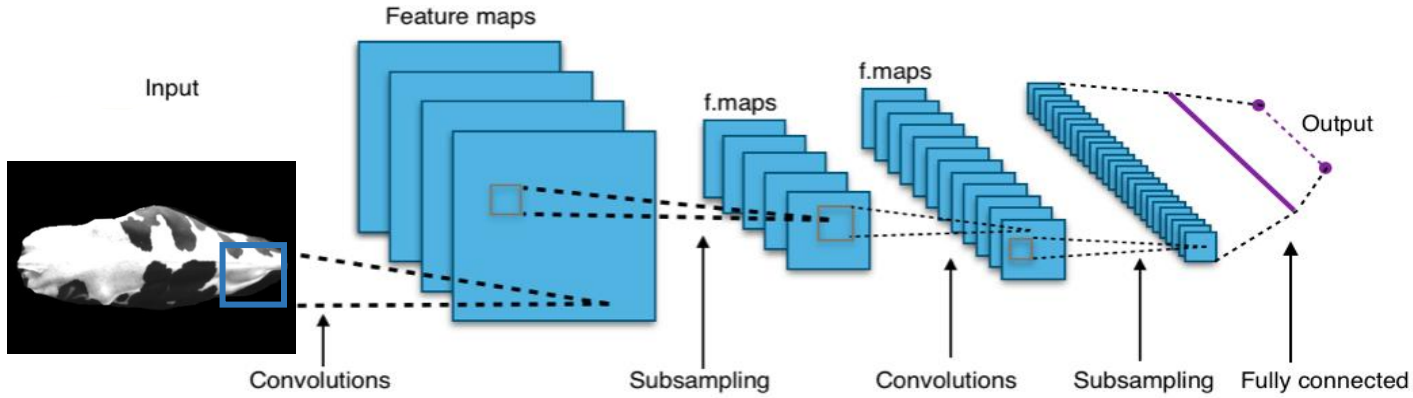
- State-of-the-art: Deep Learning

Animal Identification
Animal Behavior
Body Weight
BCS/Composition
Milk Components
Milk Yield
Estrus Event
Feed Intake
Feed Efficiency
Disease Risks

2. Why CHTC?

Deep Learning

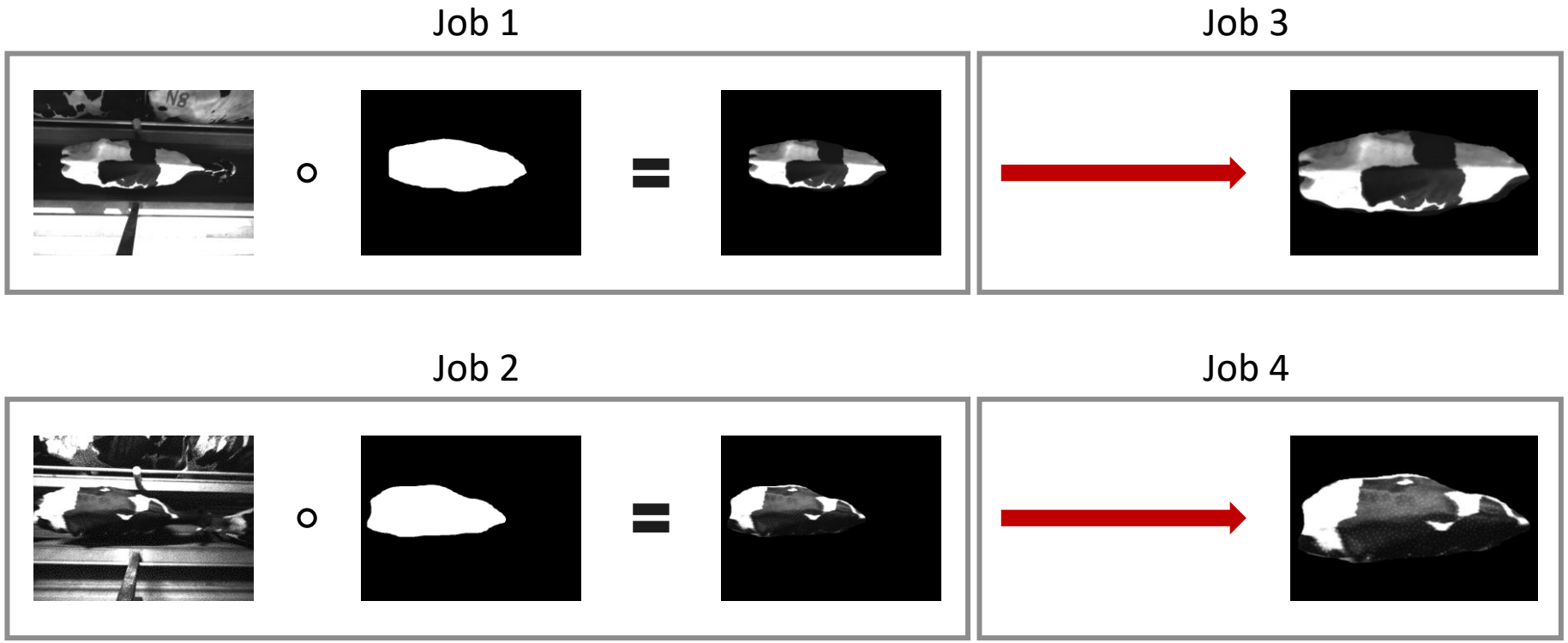
- Complex algorithms



- Requires large amounts of data (usually thousands per class)
- Image data is large (400x600 = 240,000 pixels)
- Often requires image preprocessing
- Very computationally demanding to train

Image processing

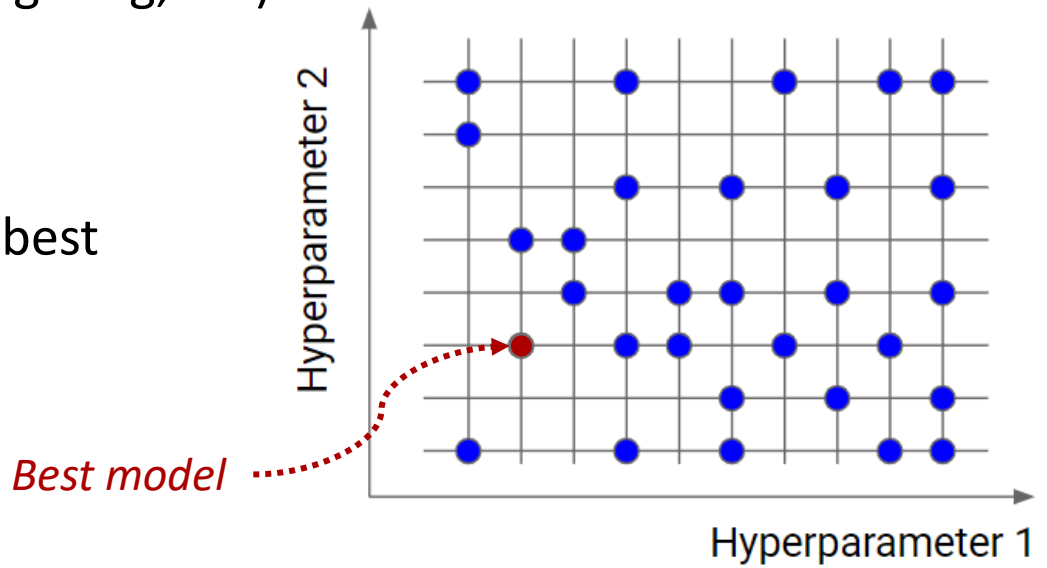
- Divide full dataset into multiple smaller sets
- Divide preprocessing stage into multiple smaller tasks



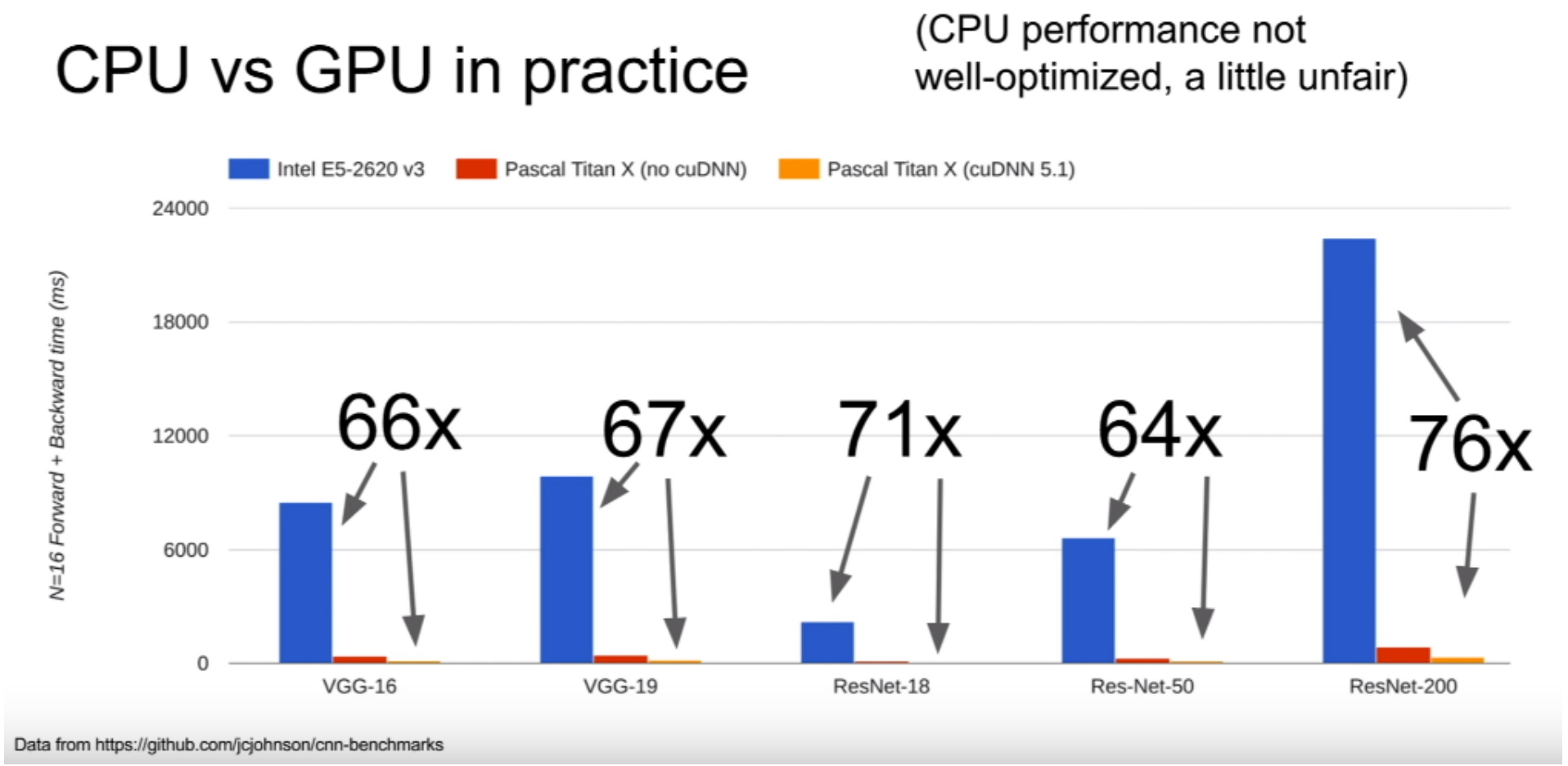
Neural network training

- Multiple neural networks to train independently:
- Different datasets

  - Evaluate which one is the best for training (preprocessing, data collection strategies, etc)
  - Perform multiple experiments (effect of day, lighting, etc)
- Hyperparameter tuning
  - Train using multiple combinations to find the best



Neural network training: GPUs



Stanford cs231n.

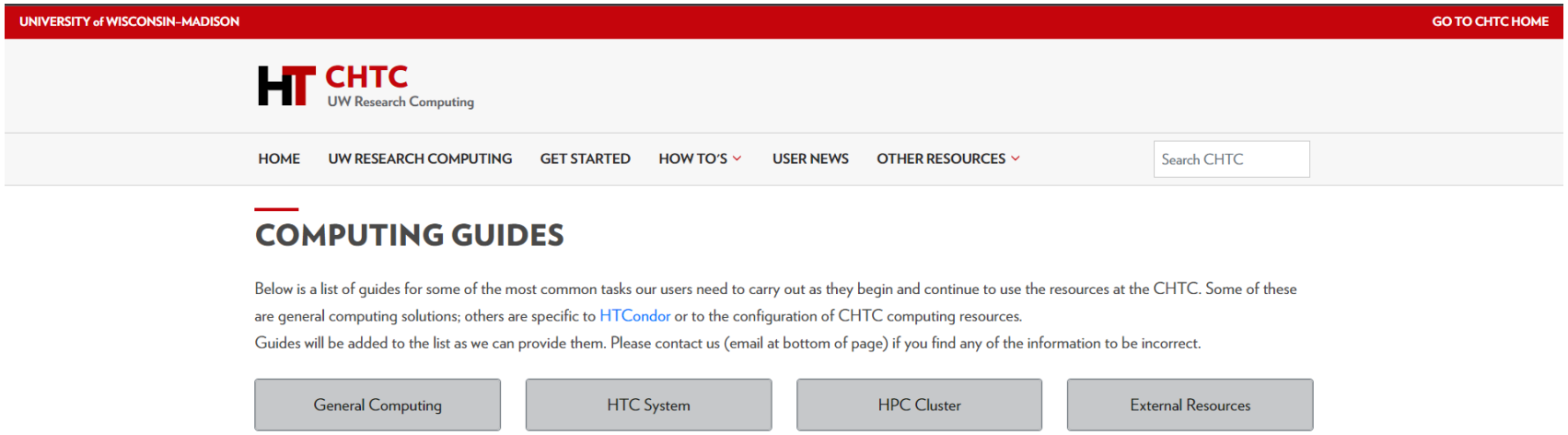
3. My experience with CHTC

Usage

- Using since Fall 2019
- > 8,000 computing hours last 15 months
- 15-20 computer vision projects
- Larger projects require dozens of trained neural networks
- Each project contains around 10,000 to 50,000 thousand images for training, and 100,000s or even millions for inference (images captured 24/7)

Learning Resources

- CHTC Computing Guides



- CHTC facilitators
  - GPU/TensorFlow errors
  - Explore new functionalities (remotely submit jobs using Python)
  - General optimizations

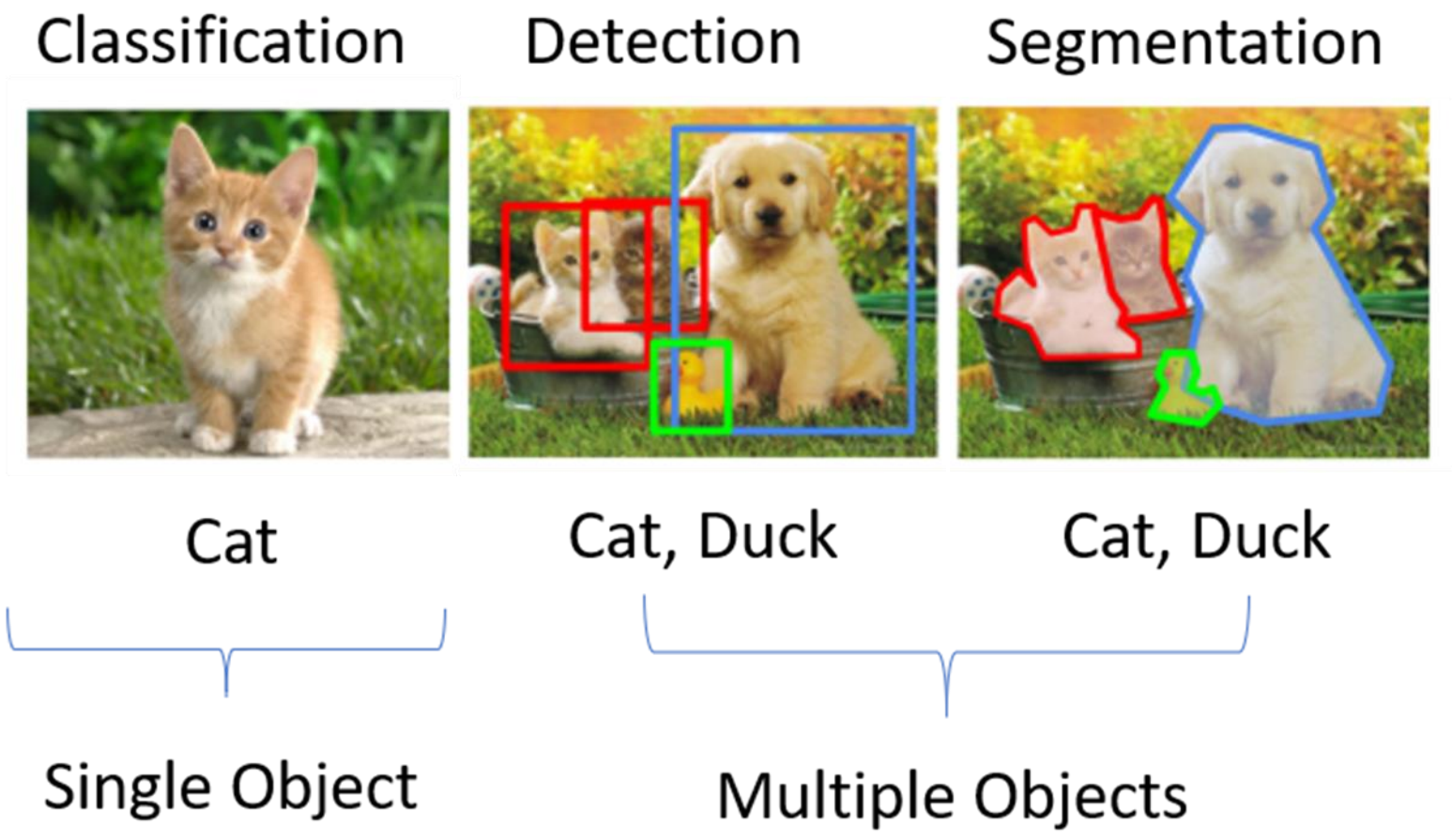
Features and optimizations

- Datasets stored on Staging
- Python environments using Miniconda
  - Compressed environment packs stored on SQUID
- Queue jobs using txt files

```
knn_iteration0_60_120_2048_000,xception,30,60,8,59
knn_iteration0_60_120_2048_050,xception,30,60,8,59
knn_iteration0_60_120_2048_075,xception,30,60,8,59
knn_iteration0_60_120_2048_100,xception,30,60,8,59
xgboost_iteration0_60_120_2048_000,xception,30,60,8,59
xgboost_iteration0_60_120_2048_050,xception,30,60,8,59
xgboost_iteration0_60_120_2048_075,xception,30,60,8,59
xgboost_iteration0_60_120_2048_090,xception,30,60,8,59
```

- Template folder for each category of project (containing .sub, .sh, python files, etc)
  - Each template expects datasets following a certain format and outputs files/folders following a certain format

Use cases


Classification
Detection
Segmentation
Cat
Cat, Duck
Cat, Duck
Single Object
Multiple Objects

Use cases: Cow identification

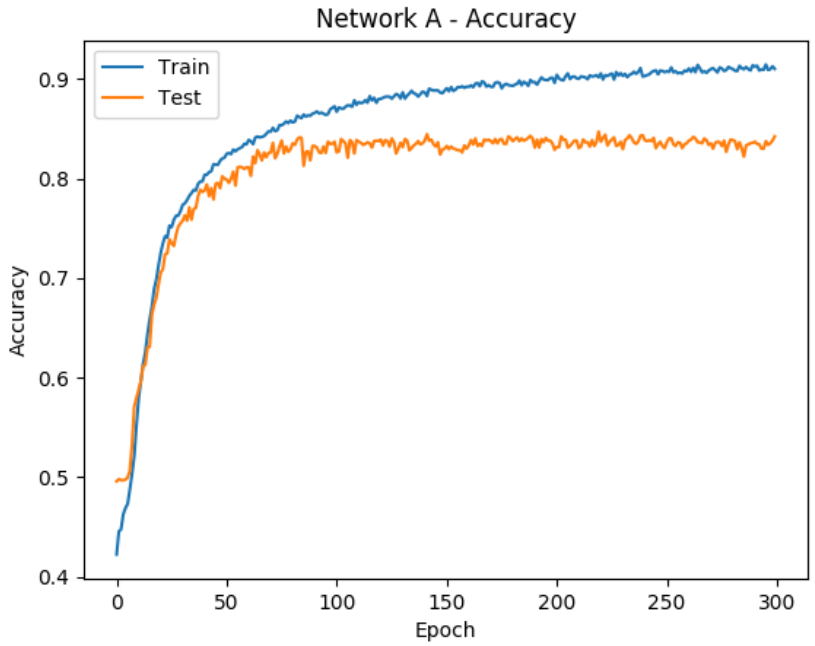
Classification

Cat

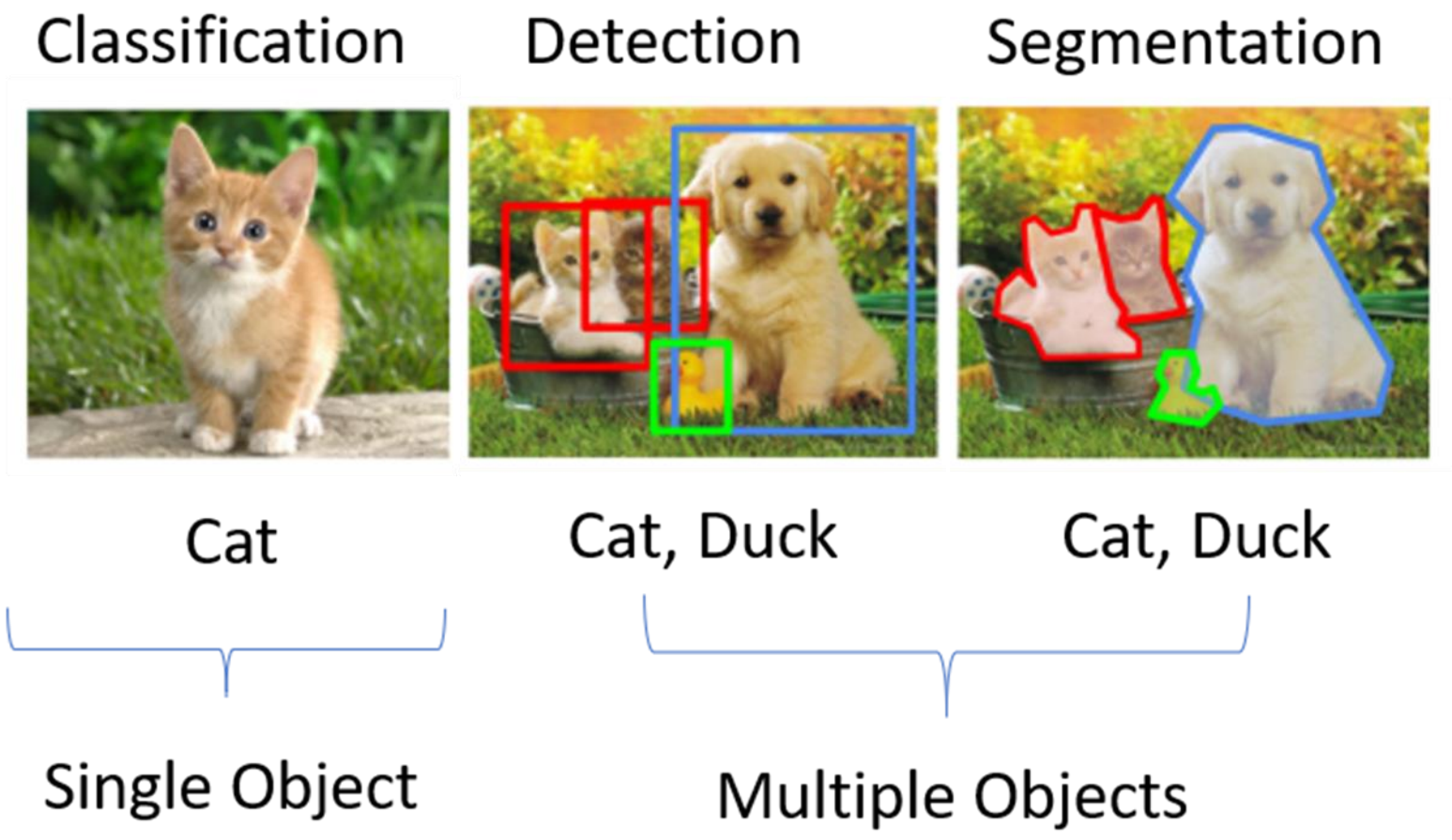
- Input: Dataset containing images separated by class folder

Dataset/
- *8199/*
  - *Img8199_01.png*
  - *Img8199_02.png*
  - *Img8199_03.png*
  - ⋮
- *9273/*
  - *Img9273_01.png*
  - *Img9273_02.png*
  - *Img9273_03.png*
  - ⋮
- ⋮

- Output: Test predictions and performance

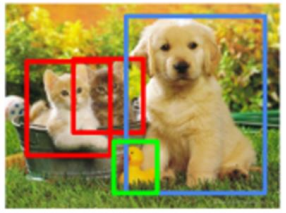


Use cases


Classification
Detection
Segmentation
Cat
Cat, Duck
Cat, Duck
Single Object
Multiple Objects

Use cases: Cow detection

Detection



Cat, Duck

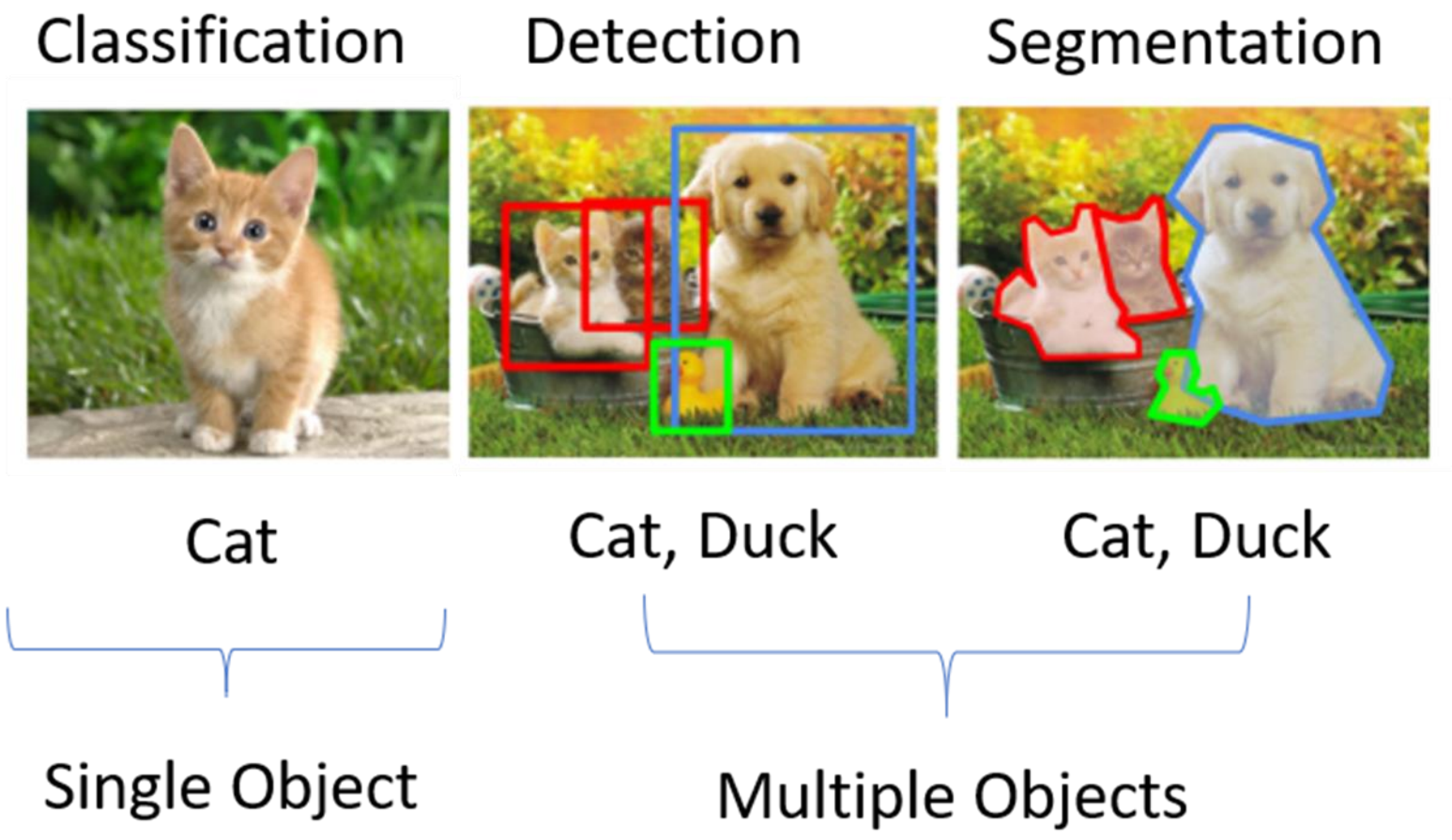
- Input: Dataset containing images and bounding boxes

```
Dataset/
    Img01.png
    Img01_bbox.txt
    Img02.png
    Img02_bbox.txt
    Img03.png
    Img03_bbox.txt
    Img04.png
    Img04_bbox.txt
    ⋮
```


```
19 0.895052 0.334259 0.0578125 0.0944444
19 0.922396 0.343056 0.065625 0.117593
19 0.963542 0.391667 0.0708333 0.181481
19 0.693229 0.828241 0.144792 0.337963
19 0.616146 0.595833 0.273958 0.308333
19 0.54349 0.3625 0.0651042 0.284259
19 0.451042 0.375 0.0739583 0.259259
19 0.58125 0.797222 0.230208 0.362963
```

- Output: Bounding box predictions on test set

Use cases


Classification
Detection
Segmentation
Cat
Cat, Duck
Cat, Duck
Single Object
Multiple Objects

Use cases: Cow body segmentation

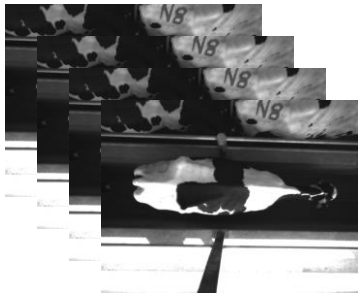
Segmentation

Cat, Duck

- Input: Dataset containing images and masks

Dataset/
- *Img01.png*
- *Img01_mask.png*
- *Img02.png*
- *Img02_mask.png*
- *Img03.png*
- *Img03_mask.png*
- *Img04.png*
- *Img04_mask.png*

⋮



- Output: Mask predictions on test set

Pain points / nice-to-haves

- Large datasets can take too long to transfer, especially when working from home using VPN
- Not being able to see log (printed to .err and .out files) when a job is held

- Being able to access data directly from our own servers
- Checking logs to have an idea of how far into the job (which epoch, for example) the 12/24/72 hr limits were reached

Pain points / nice-to-haves

- Having to frequently check if jobs are done running

- Having e-mails sent (or other notification system) when jobs are done running
- Being able to read .err and .out while the job is running (maybe there is a way that I'm not yet aware of)

Personal and professional benefits

Personal and professional benefits

- Perform dozens of experiments in parallel
- Experience accessing a remote Linux server using SSH
- I might come across other high-throughput computing systems in the future
- Think about how to break computer tasks into smaller bits
- Consider data flow and automation within remote server environments

✓ Made our research possible!

Thank you!

referreira@wisc.edu

WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON