Team B1: PhotoRobo

Justin Kiefel, Sidhant Motwani, Fernando Paulino

Putting it all together...

Use Case Requirements

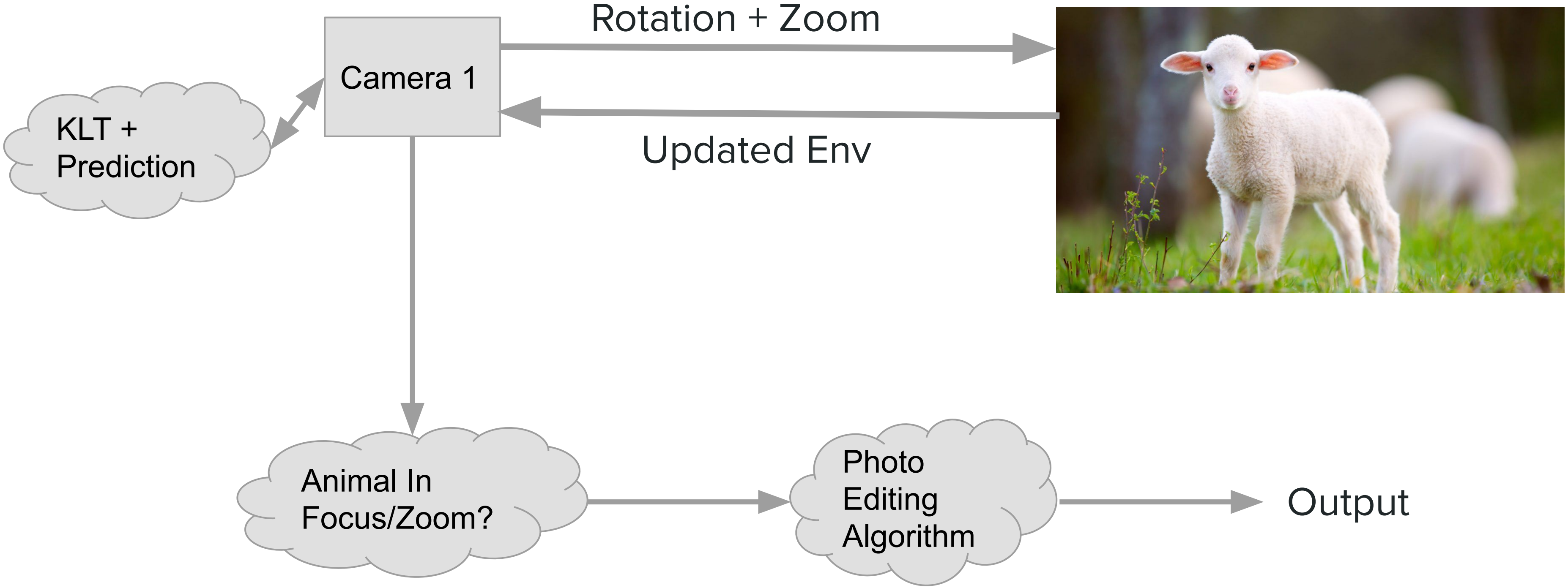
- Problem: Animal photography is time consuming and mundane. Remote controlled photography robots do little to fix this problem. Photo editing is necessary, but similarly human-capital intensive.
- Goal: Produce a stationary nature photography robot, which can locate, track and photograph animals. The system should also perform automated image editing.
- ECE Areas:
  - Signals and Systems - Image Processing and Control Systems
  - Software Systems - Embedded System Programming

Quantitative Requirements

- Detection Ability: The system must detect animals within 25 meters with a recall of 75%
- Detection Speed: The system must detect animals within 15 seconds.
- Tracking Ability: The system must be able to follow and photograph an animal moving 2 m/s.
- Photo Quality: The photo should be 12MP and have quality indistinguishable from a human shot and edited photograph. Human testers should not do better than guessing (50% accuracy) when labeling photos as robot or human pictures.

Solutions Approach

- 3 Phase Software Approach
  - Animal Search and Detection
  - Tracking and Photography
  - Photo Editing



Camera 1

KLT + Prediction

Rotation + Zoom

Updated Env

Animal In Focus/Zoom?

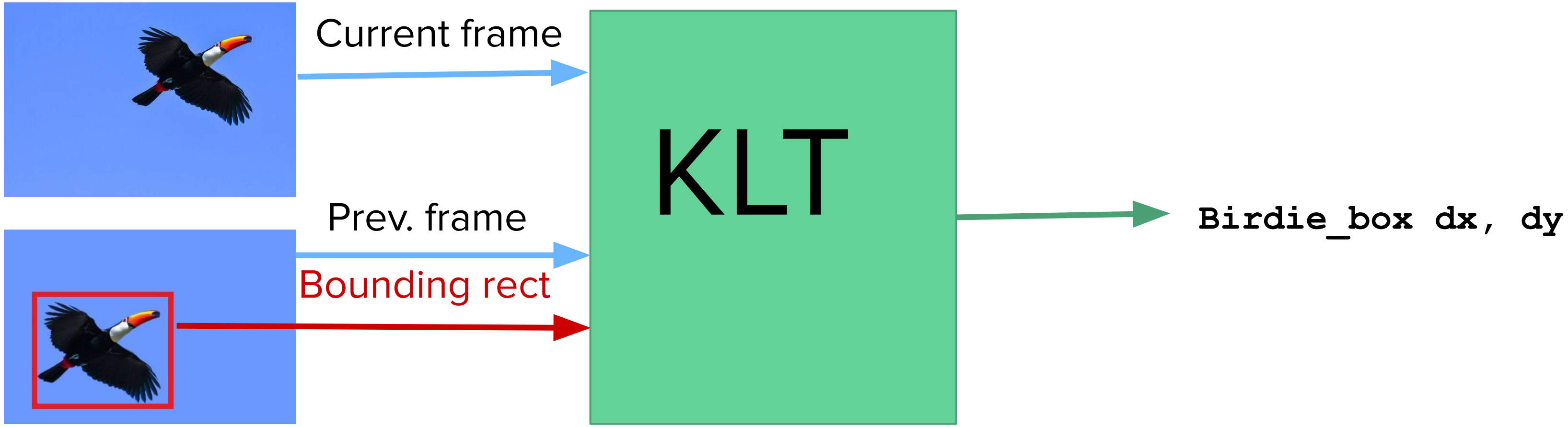
Photo Editing Algorithm

Output

Complete Solution - Tracking

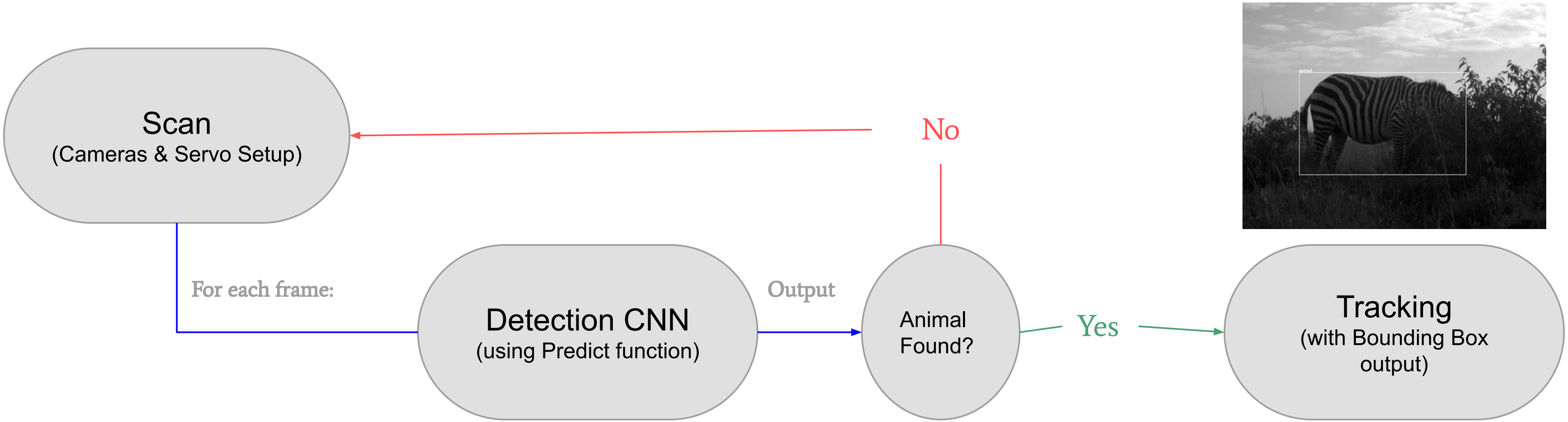
Lucas-Kanade Tracker

- Numpy based KLT
- GSTREAMER pipes images from Arducam to the app
- ~~Feature matching algorithm to align the two cameras~~



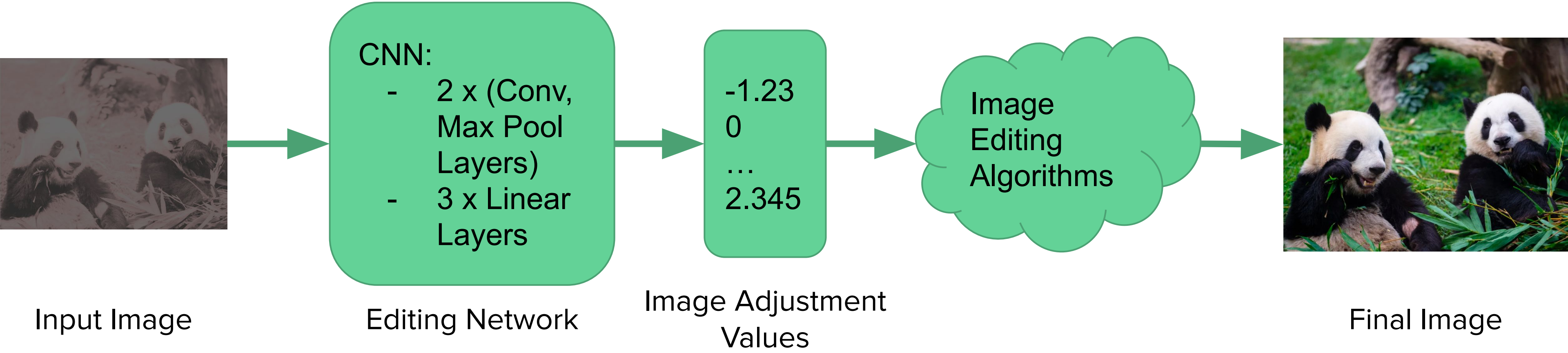
Complete Solution - Search and Detection

- Constant speed scan using Camera setup with different Zoom layers
- Frame-by-frame processing
- The trained convolutional neural network outputs a bounding box for any animal found
- Transfers control to ‘Tracking’, by providing it the location of the bounding box



Complete Solution - Photo Editing

- 7 Image Editing Algorithms (Sharpening, Contrast, Tint, Temperature, Saturation, Vibrance, Exposure)
- CNN Trained to apply various amounts of algorithms
- Dataset created by applying random modifications to Kaggle Animal Classification Dataset
    - Inverse functions of editing algorithms implemented to get ground truth editing amounts



CNN:

- 2 x (Conv, Max Pool Layers)
- 3 x Linear Layers

-1.23
0
…
2.345

Image Editing Algorithms

Input Image

Editing Network

Image Adjustment Values

Final Image

Testing, Verification, and Validation- Detection and Tracking

- Subject: Printed cutouts of animal pictures
- Search & Detection
  - Placed at different distances
  - Placed in different lighting conditions
  - Different Orientations
- Fast + accurate camera

  >=

  human reaction speed?

Detection Neural Network Training Results		
Epochs: 30	Recall: ~ 93%	Accuracy: ~ 92.8%

Testing, Verification, and Validation - Photo Editing

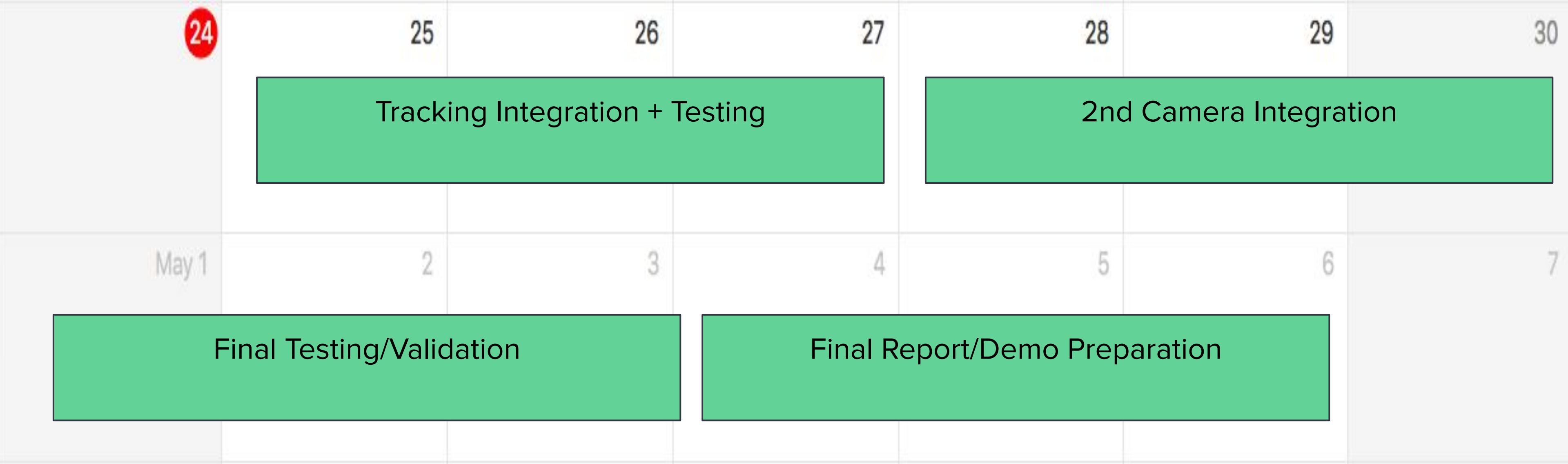
- 2 Testers (Not members of project) shown 50 image pairs for each image editing approach and asked to identify original image
- One image is original image, and the other is an edit of a randomly augmented image
- Ideal image editing algorithms will lead to a 50% correct rate

Photo Editing Testing Results (Percent Correct Guesses)		
No Modification	Heuristic Editing Rules (Original Approach)	Neural Network Approach
91%	85%	89%

Future Work

- **Detection**: Large and Precise Dataset
- **Tracking**: Two Camera Approach for Focussing
- **Editing**: Different editing CNN architectures
- **Hardware**:
    - Update to two cameras
    - Better Processing Unit (>Jetson Nano)

Project Management



24 25 26 27 28 29 30

Tracking Integration + Testing

2nd Camera Integration

May 1 2 3 4 5 6 7

Final Testing/Validation

Final Report/Demo Preparation