Where is the bear?

Implementing Machine Learning-Based Image Recognition for Animal Detection

Department of Computer Science

Andy Rosales Elias
Mentor: Nevena Golubovic
Advisors: Chandra Krintz, Rich Wolski
Lab: RACELab

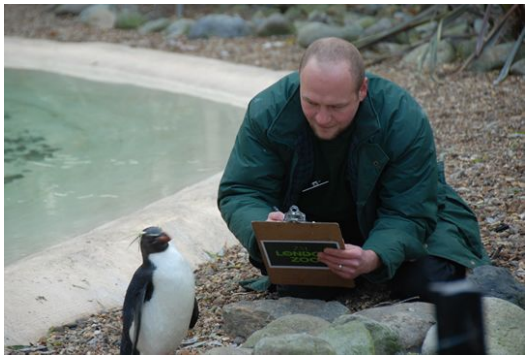
UCSB

CSEP

Carters New Agency
http://www.catersnews.com/

Carters New Agency
http://www.catersnews.com/

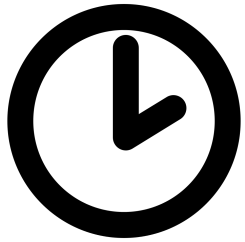
The Problem of Accuracy, Cost, And Efficiency



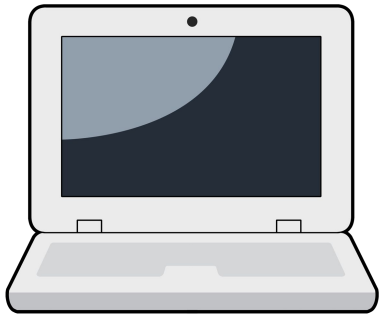
=

+

+



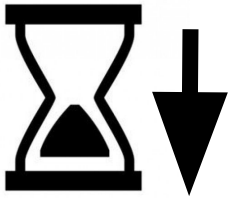
The Problem of Accuracy, Cost, And Efficiency



=

+ Data Reliability

+

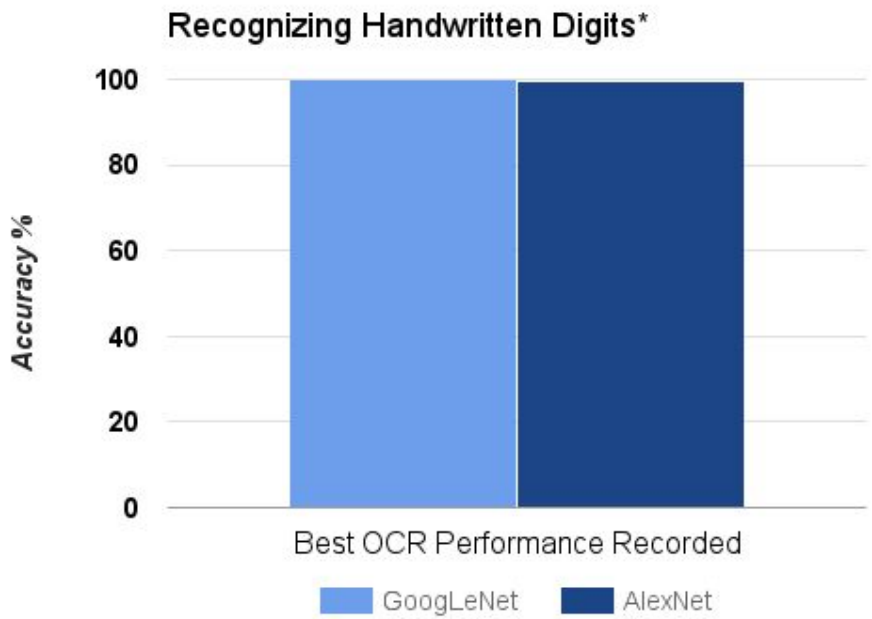


Animal Recognition Using Different Machine Learning Frameworks

Output: “Bear”

Output: “None”

Using Convolutional Neural Networks (CNN) to Recognize Animals

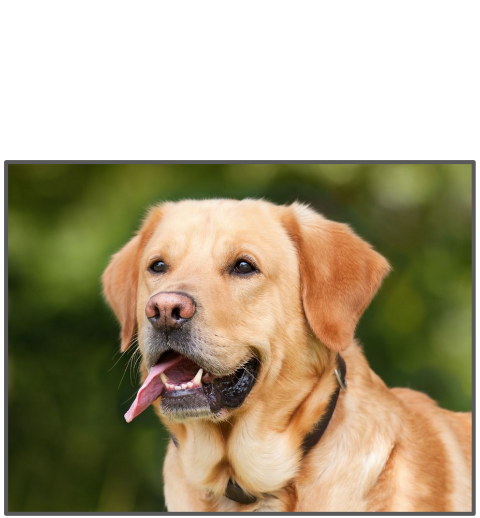


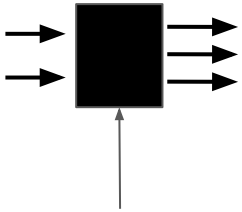
- Almost every highly ranked team used CNN as their basic framework

- Foundation of GoogLeNet and AlexNet

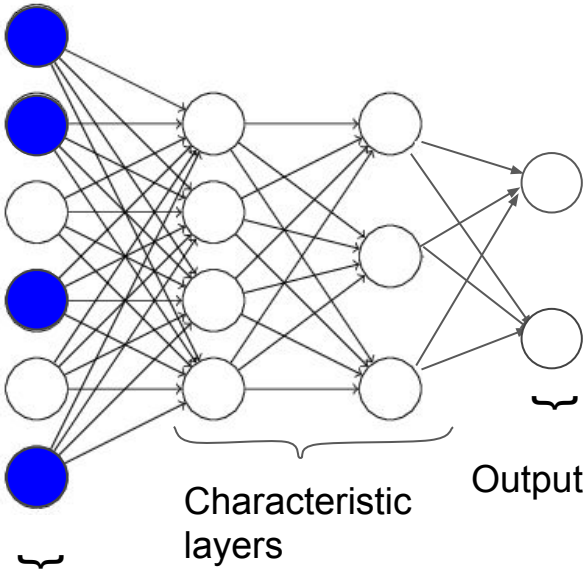
-Zhong, Zhuoyao et al. “High Performance Offline Handwritten Chinese Character Recognition Using GoogLeNet and Directional Feature Maps.”
-"Multi-column deep neural networks for image classification". *2012 IEEE Conference on Computer Vision and Pattern -Recognitio*n
-Ciresan, Dan; Ueli Meier; Jonathan Masci; Luca M. Gambardella; Jurgen Schmidhuber. "Flexible, High Performance Convolutional Neural Networks for Image Classification"

How Convolutional Neural Networks Classify Images





Feature extractor
black box



Deactivated neuron

Activated neuron

Dog

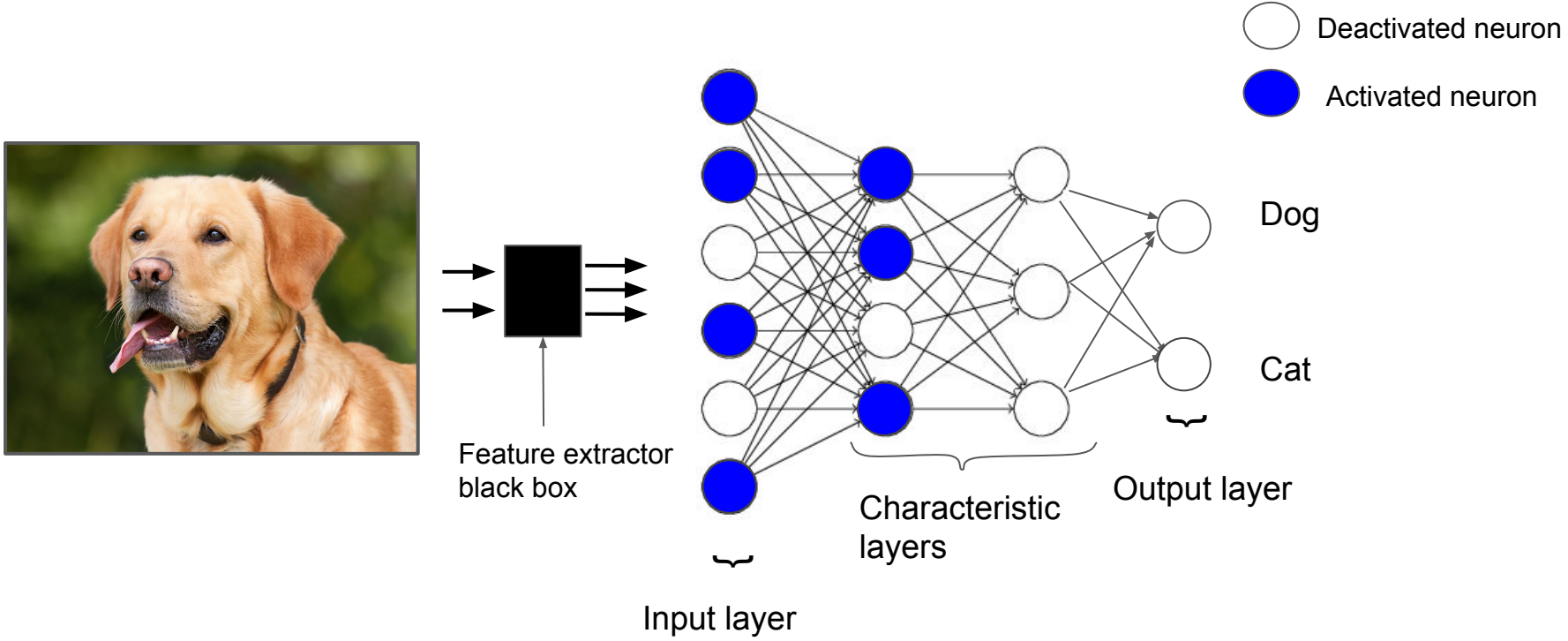
Cat

Input layer

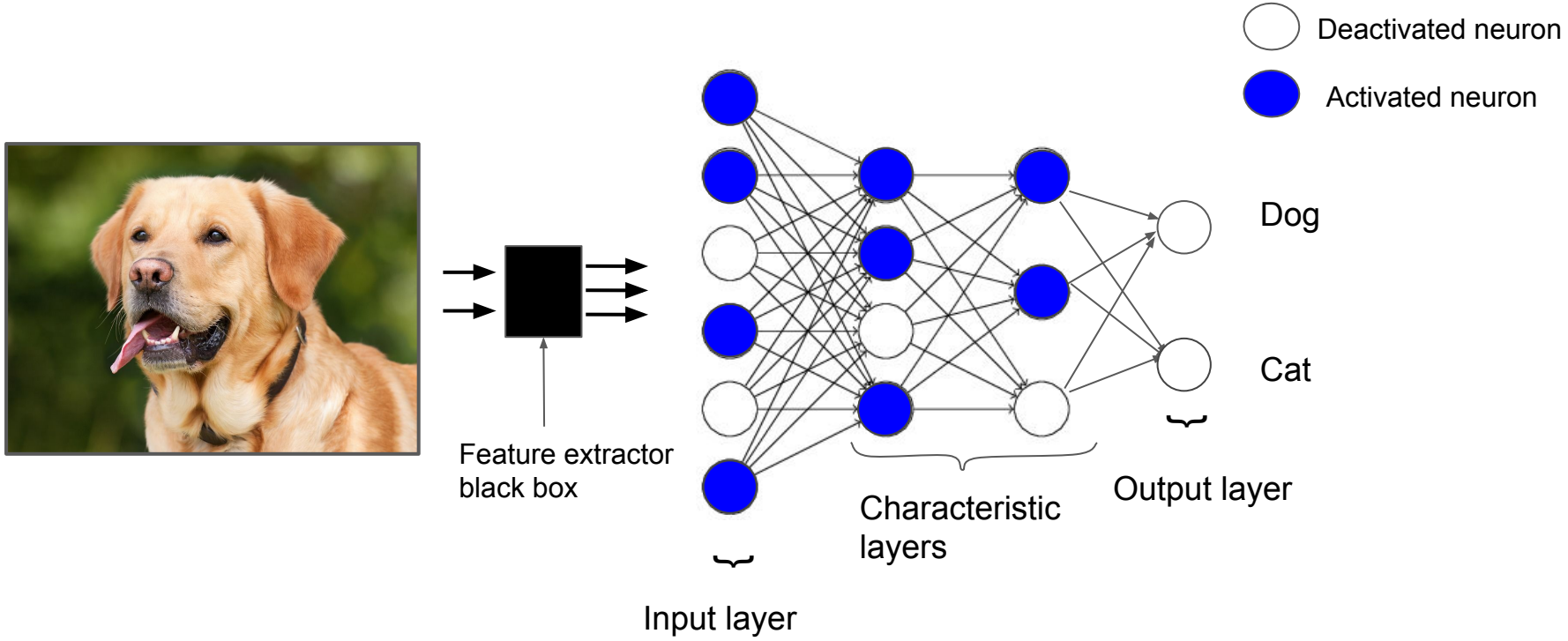
Characteristic
layers

Output layer

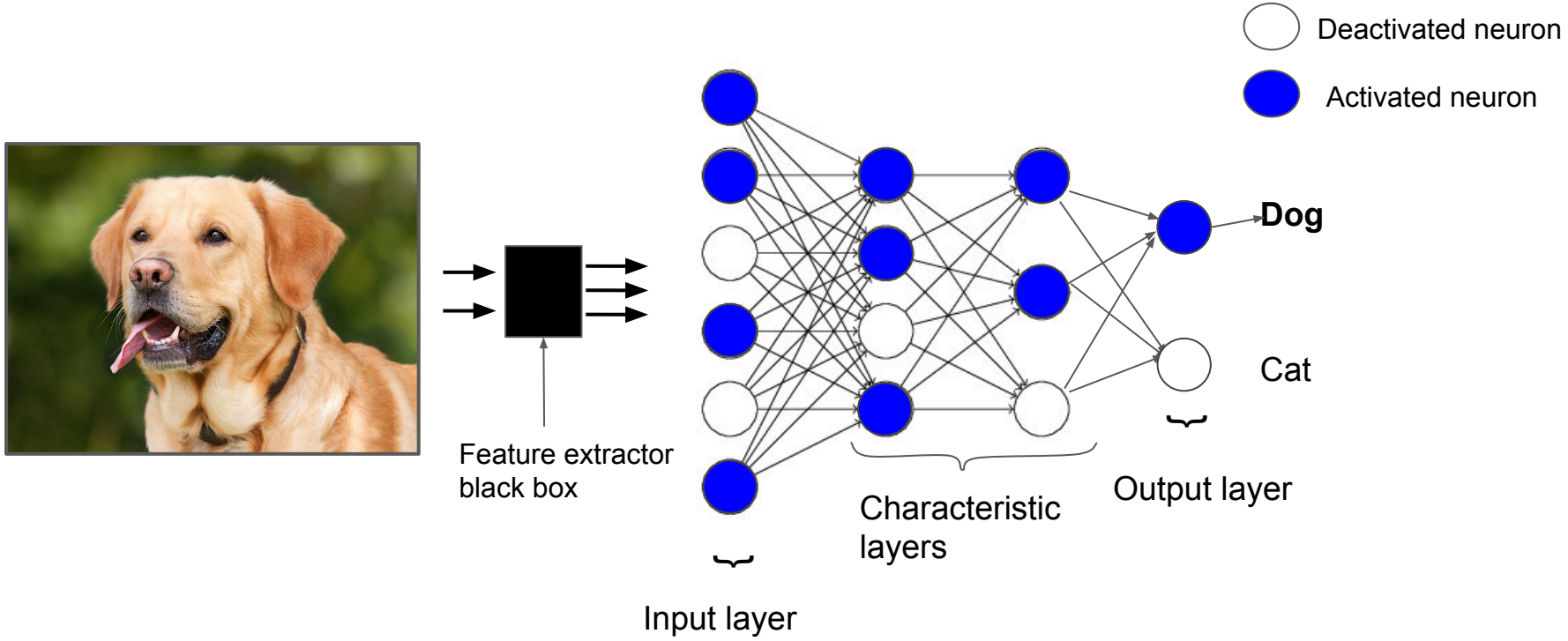
How Convolutional Neural Networks Classify Images



How Convolutional Neural Networks Classify Images

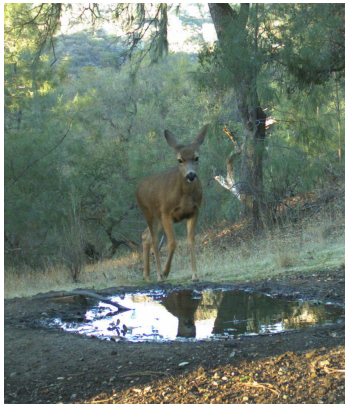


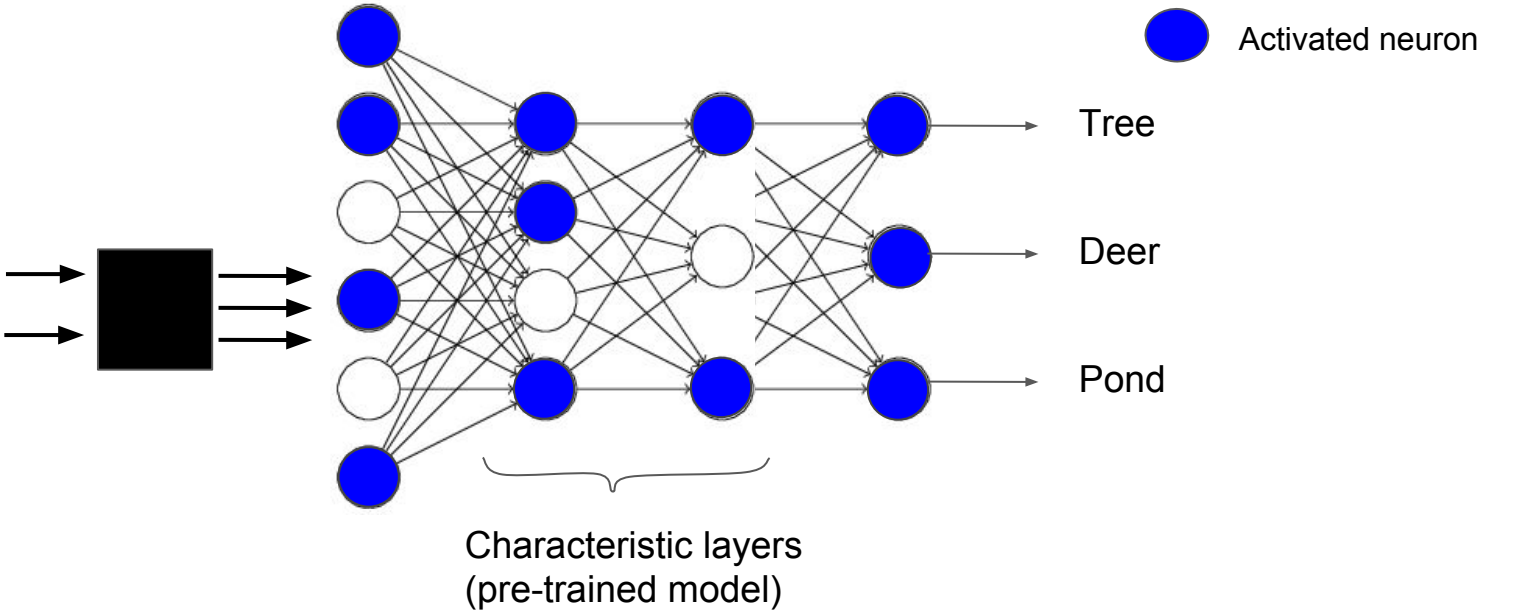
How Convolutional Neural Networks Classify Images



Re-training an Existing CNN Model

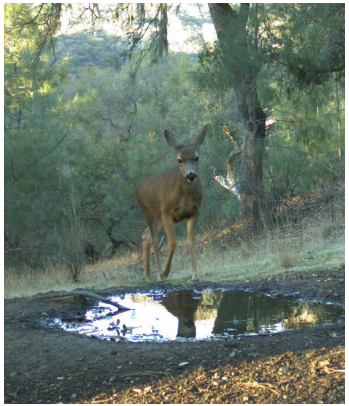
- Use a pre-trained CNN model

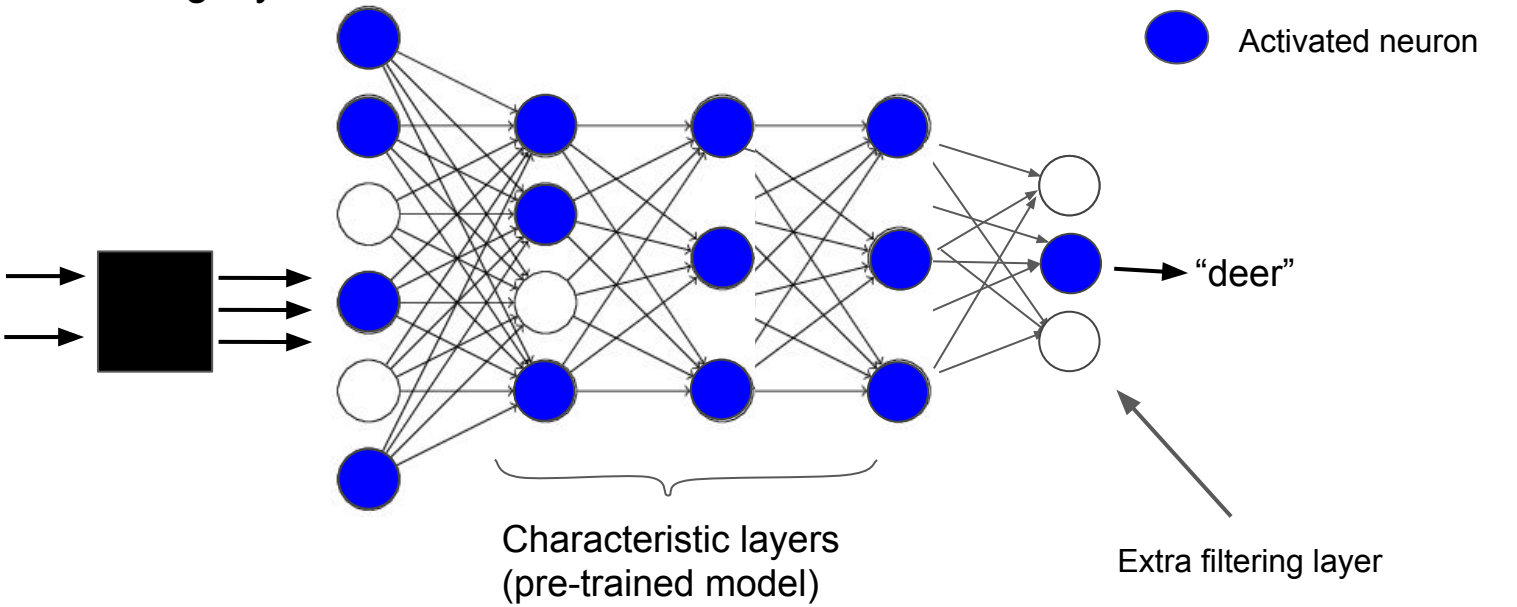




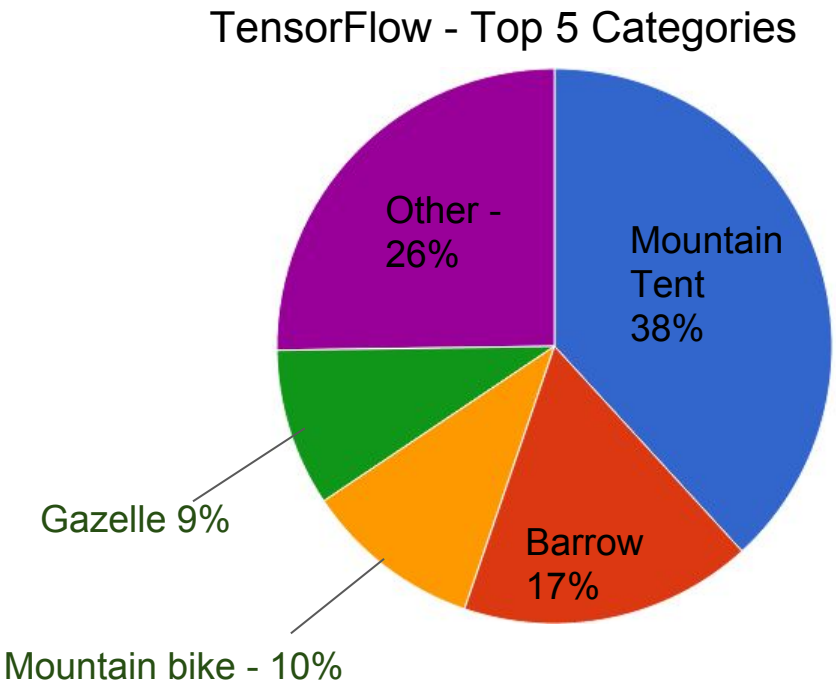
Re-training an existing CNN Model

- Use a pre-trained CNN model
- Add an extra filtering layer

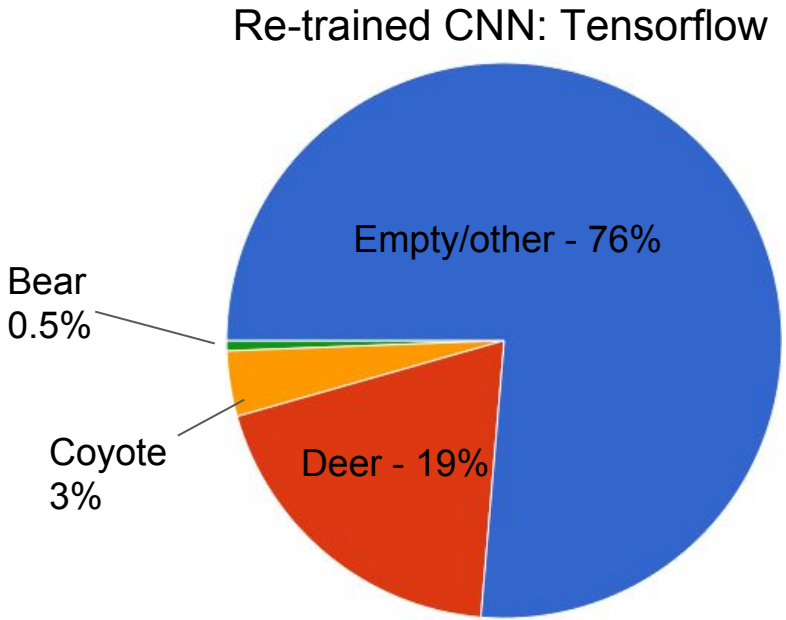




Results After Classifying 12210 Images Using Pre-trained CNN Model



Results After Classifying 12210 Images Using TensorFlow Re-trained In Four Categories

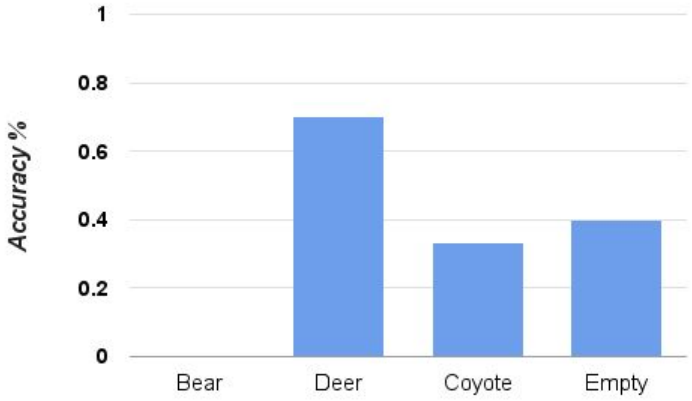


Re-training Dataset:

- Bear: 170 images
- Deer: 489 images
- Coyote: 259 images
- Empty/other: 21 images

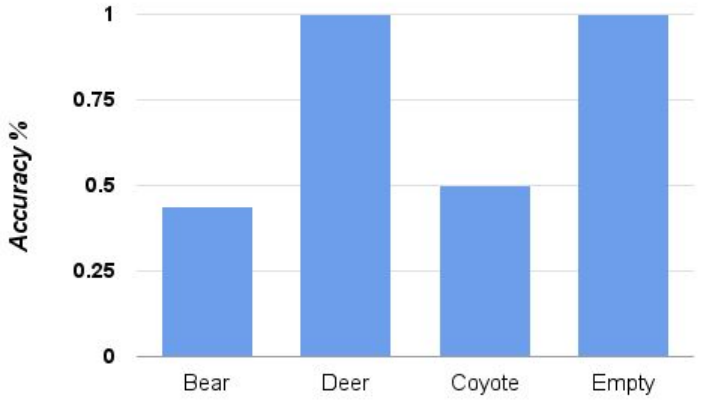
Accuracy of TensorFlow: Before And After Re-Training

No Re-training



Average accuracy: 35%

Retrained in four categories



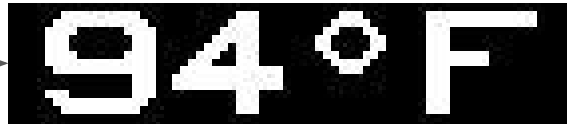
Average accuracy: 73%

Metadata Extraction & Fusion

- Learn more about how animals are affected by the environment.
- Analyze this data to learn more about certain species.

Extracting metadata from images

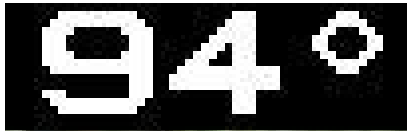
- Time and date can be extracted from the EXIF data
- Temperature needs to be recognized using OCR



Camera:	Reconyx HC500 HYPERFIRE
Exposure:	Auto exposure, 1/30 sec, ISO 800
Flash:	Auto, Fired
Date:	**January 1, 2016** 6:30:55AM (timezone (6 months, 25 days, 17 hours, 52 minutes, 17 s Pacific)
File:	**1,920 × 1,080** JPEG (**2.1** megapixels) 248,746 bytes (243 kilobytes)

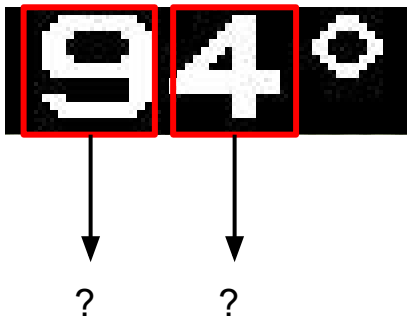
Recognizing temperature digits using OCR

Have temperature digits recognized automatically



Recognizing temperature digits using OCR

Have temperature digits recognized automatically

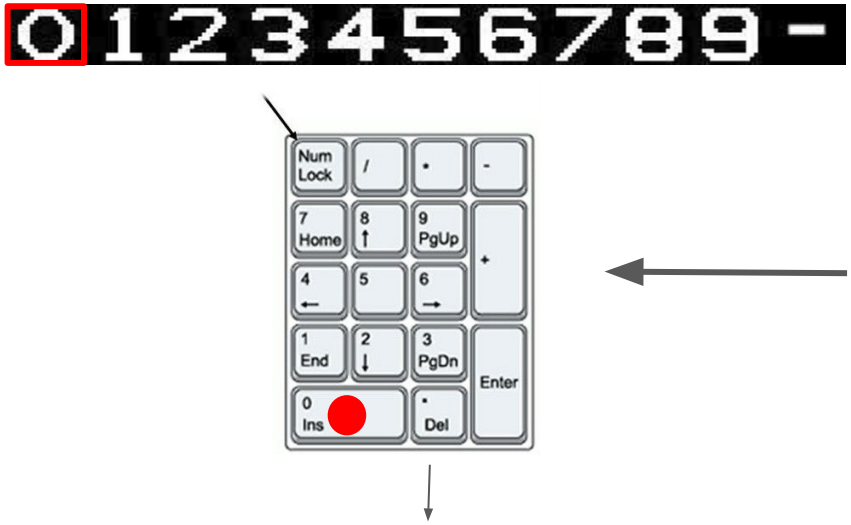


Training phase - labeling digits

Label each digit in the font family in order to train the model

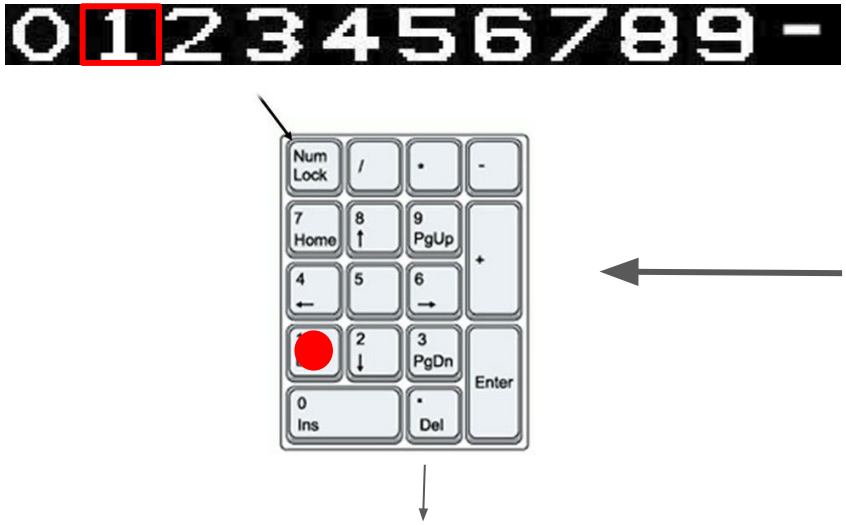
Training phase - labeling digits

Label each digit in the font family in order to train the model



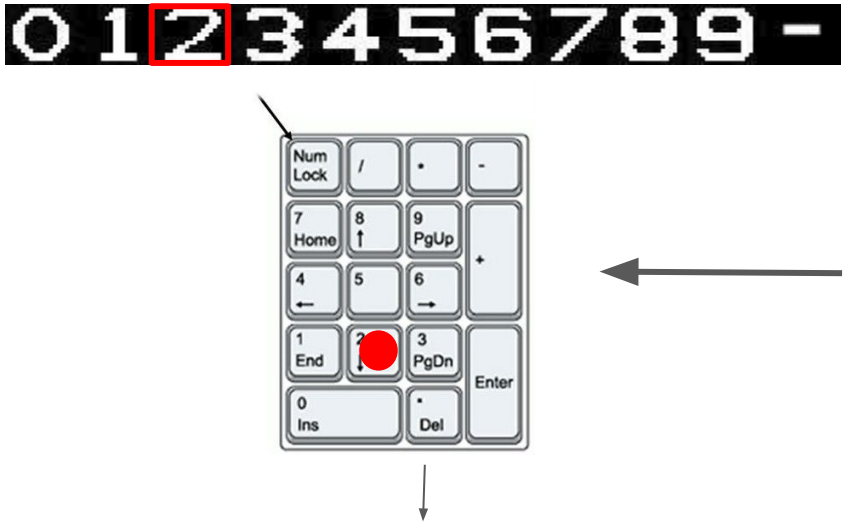
Training phase - labeling digits

Label each digit in the font family in order to train the model



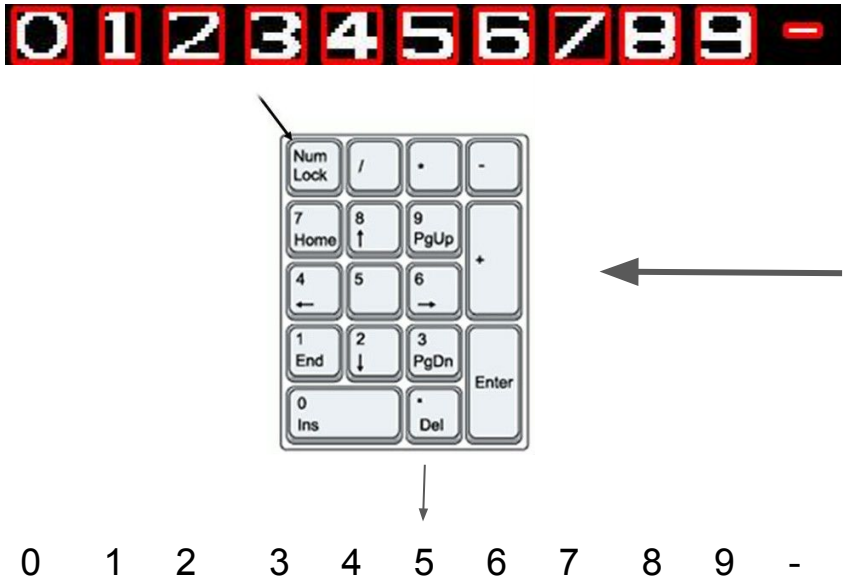
Training phase - labeling digits

Label each digit in the font family in order to train the model



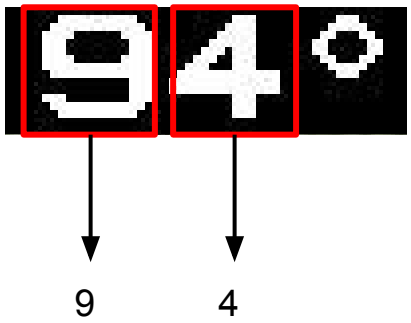
Training phase - labeling digits

Label each digit in the font family in order to train the model



Recognizing temperature digits using OCR

- The trained model will recognize any unlabeled character
- Which number is the “nearest” in similarity with the labeled data

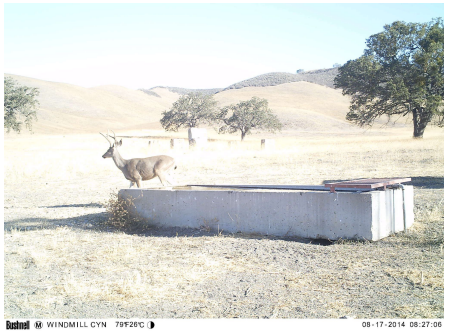


Collecting metadata with hidden, motion-triggered cameras at the Sedgwick Reserve

Camera 1

Camera 2

Camera 3

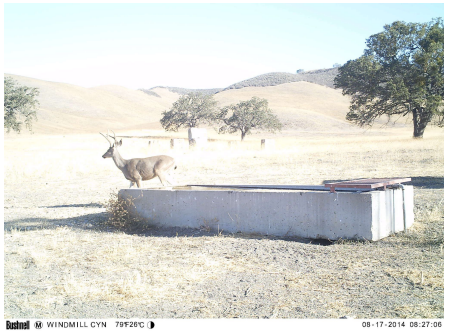


Collecting metadata with hidden, motion-triggered cameras at the Sedgwick Reserve

Camera 1

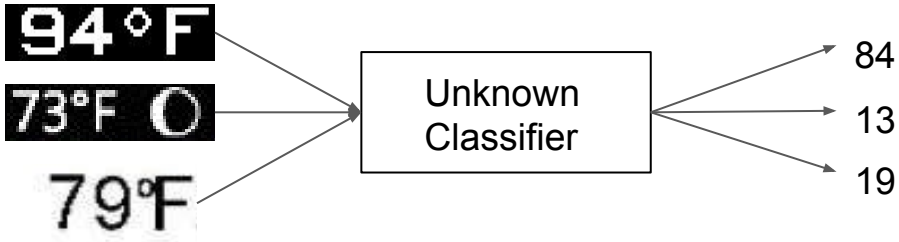
Camera 2

Camera 3



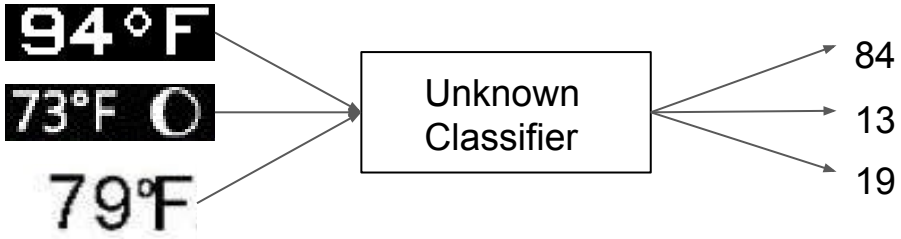
How each OCR algorithm processes temperature images

Previous OCR

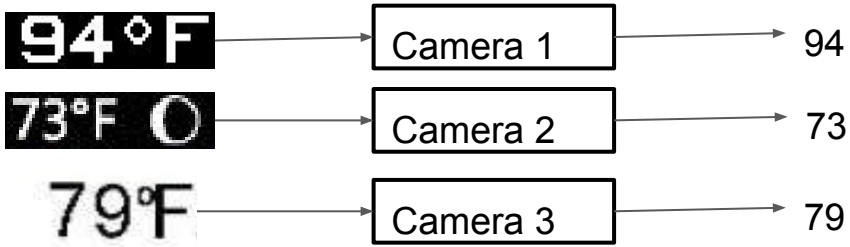


How each OCR algorithm processes temperature images

Previous OCR

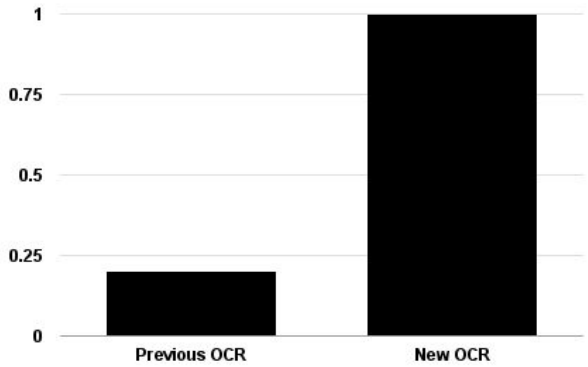


New OCR

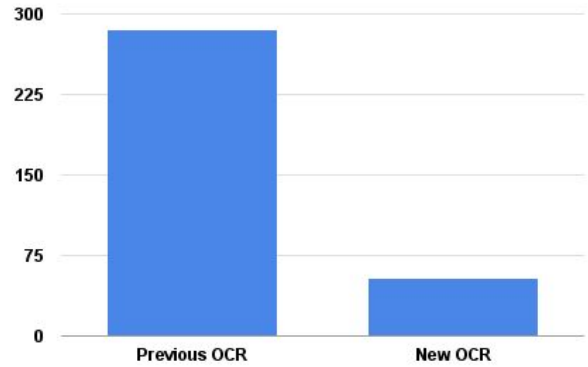


Analyzing the accuracy and processing time of both OCR algorithms

Accuracy %

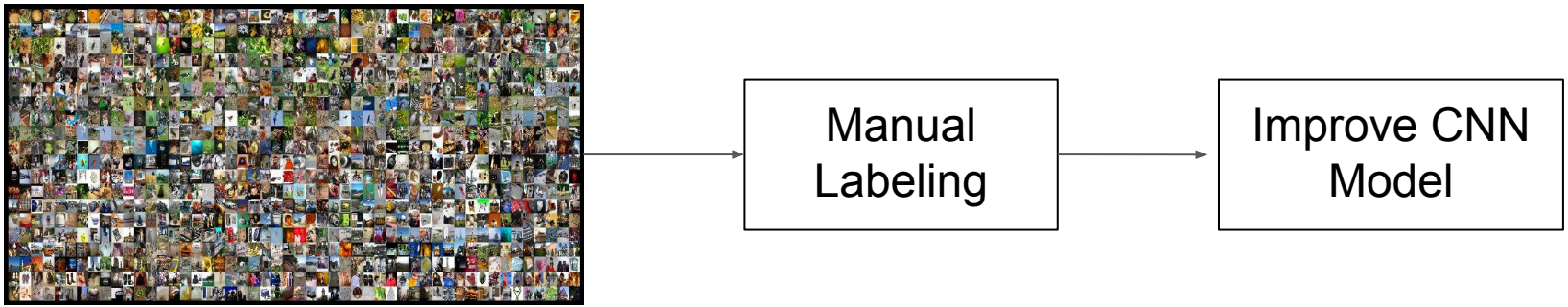


Time in seconds



The Road Ahead

- Re-train TensorFlow with labeled images from Sedgwick Reserve
- Collect more data to build a bigger dataset
- Move everything to the cloud

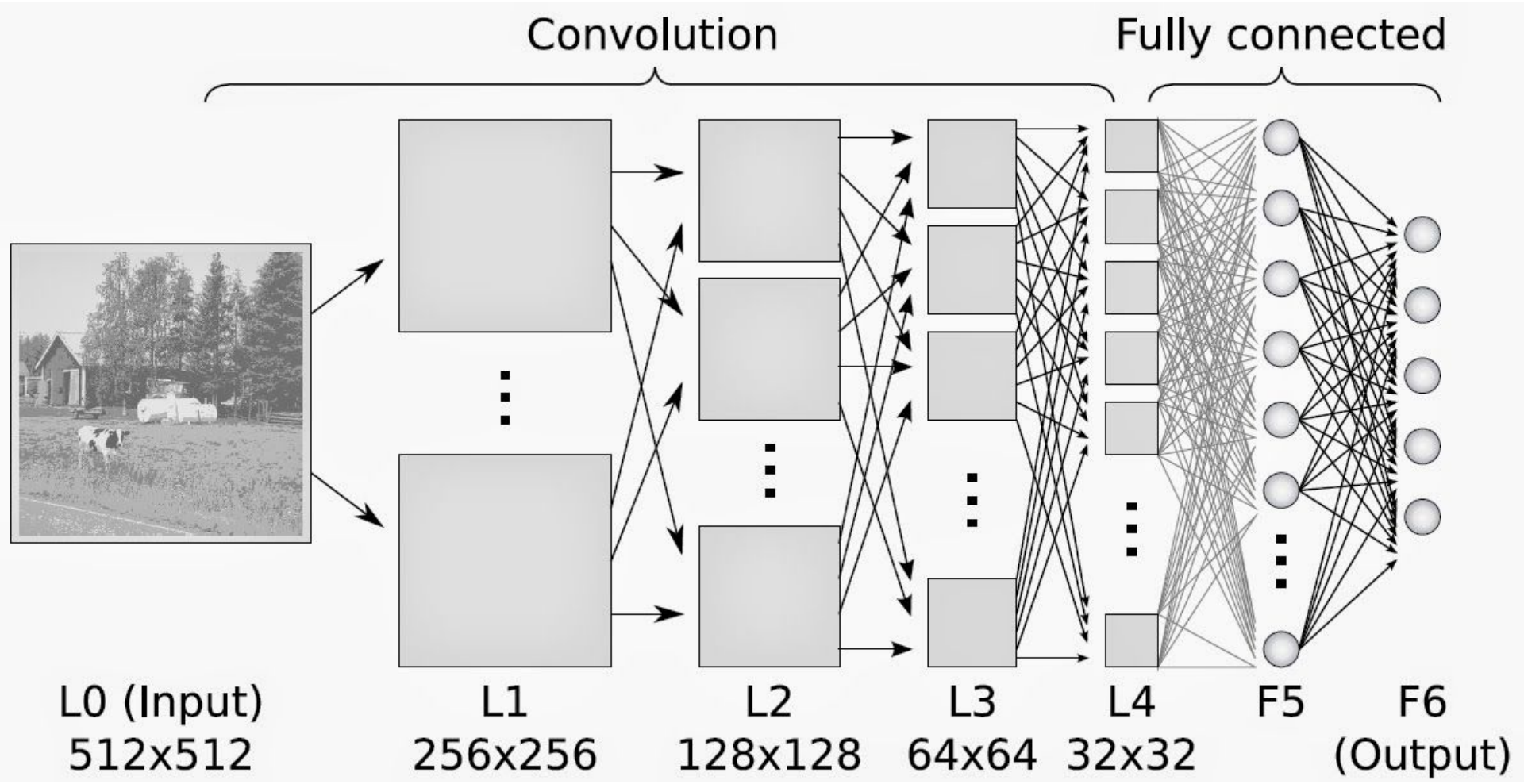


Acknowledgments

Special Thanks To:
Mentor: Nevena Golubovic
Advisors: Chandra Krintz and Rich Wolski
Program Coordinator: Stephanie Mendes

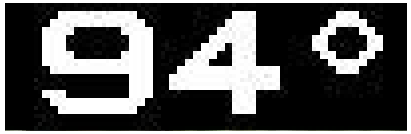
UCSB

CSEP


Convolution
Fully connected
L0 (Input)
512x512
L1
256x256
L2
128x128
L3
64x64
L4
32x32
F5
F6
(Output)

Recognizing temperature digits using OCR

- We need to automatically have these numbers recognized



Recognizing temperature digits using OCR

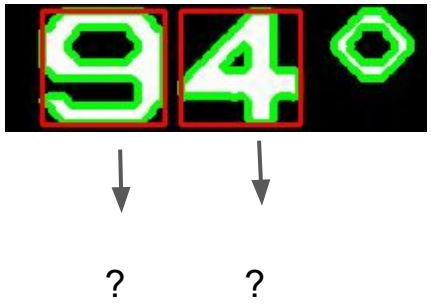
- We need to automatically have these numbers recognized
- Find the edges on the image

Recognizing temperature digits using OCR

- We need to automatically have these numbers recognized
- Find the edges on the image
- With the edges, we know the dimensions of each digit/symbol

Recognizing temperature digits using OCR

- We need to automatically have these numbers recognized
- Find the edges on the image
- With the dimensions, we ignore characters with small area (degree symbol)



- However, numbers are still unlabeled